

Unmanned Autonomous Systems in Complex Environments

Lead Guest Editor: Chenguang Yang

Guest Editors: Guang Li, Ning Wang, Xin Dong, and Rongxin Cui





Unmanned Autonomous Systems in Complex Environments

Complexity

Unmanned Autonomous Systems in Complex Environments

Lead Guest Editor: Chenguang Yang


Guest Editors: Guang Li, Ning Wang, Xin Dong,
and Rongxin Cui



Copyright © 2021 Hindawi Limited. All rights reserved.

This is a special issue published in "Complexity." All articles are open access articles distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Chief Editor

Hiroki Sayama , USA

Associate Editors

Albert Diaz-Guilera , Spain
Carlos Gershenson , Mexico
Sergio Gómez , Spain
Sing Kiong Nguang , New Zealand
Yongping Pan , Singapore
Dimitrios Stamovlasis , Greece
Christos Volos , Greece
Yong Xu , China
Xinggang Yan , United Kingdom

Academic Editors

Andrew Adamatzky, United Kingdom
Marcus Aguiar , Brazil
Tarek Ahmed-Ali, France
Maia Angelova , Australia
David Arroyo, Spain
Tomaso Aste , United Kingdom
Shonak Bansal , India
George Bassel, United Kingdom
Mohamed Boutayeb, France
Dirk Brockmann, Germany
Seth Bullock, United Kingdom
Diyi Chen , China
Alan Dorin , Australia
Guilherme Ferraz de Arruda , Italy
Harish Garg , India
Sarangapani Jagannathan , USA
Mahdi Jalili, Australia
Jeffrey H. Johnson, United Kingdom
Jurgen Kurths, Germany
C. H. Lai , Singapore
Fredrik Liljeros, Sweden
Naoki Masuda, USA
Jose F. Mendes , Portugal
Christopher P. Monterola, Philippines
Marcin Mrugalski , Poland
Vincenzo Nicosia, United Kingdom
Nicola Perra , United Kingdom
Andrea Rapisarda, Italy
Céline Rozenblat, Switzerland
M. San Miguel, Spain
Enzo Pasquale Scilingo , Italy
Ana Teixeira de Melo, Portugal

Shahadat Uddin , Australia
Jose C. Valverde , Spain
Massimiliano Zanin , Spain

Contents

High-Accuracy Real-Time Fish Detection Based on Self-Build Dataset and RIRD-YOLOv3

Wenkai Wang, Bingwei He , and Liwei Zhang 




Research Article (8 pages), Article ID 4761670, Volume 2021 (2021)

Robust Control for the Suspension Cable System of the Unmanned Helicopter with Sensor Fault under Complex Environment

Rong Mei 


Research Article (9 pages), Article ID 8869292, Volume 2021 (2021)

Multiscale Receptive Fields Graph Attention Network for Point Cloud Classification

Xi-An Li , Li-Yan Wang , and Jian Lu 






Research Article (9 pages), Article ID 8832081, Volume 2021 (2021)

Research on the Precession Characteristics of Hemispherical Resonator Gyro

Li-Jun Song , Rui Yang, Wang-Liang Zhao, Xing He, Shaoliang Li, and You-Jun Ding



Research Article (9 pages), Article ID 8825017, Volume 2021 (2021)

Effect of Bird Yaw/Pitch Angles on Soft Impact Damage of a Fan Assembly

Junjie Li , Yunfeng Lou , Gaoyuan Yu , Tong Li , and Xianlong Jin 

Research Article (13 pages), Article ID 8879874, Volume 2021 (2021)

Stochastic Parameter Identification Method for Driving Trajectory Simulation Processes Based on Mobile Edge Computing and Self-Organizing Feature Mapping

Jingfeng Yang, Zhiyong Luo , Nanfeng Zhang, Jinchao Xiao, Honggang Wang, Shengpei Zhou, Xiaosong Liu, and Ming Li 


Research Article (8 pages), Article ID 8884390, Volume 2021 (2021)

A Gradient-Based Recurrent Neural Network for Visual Servoing of Robot Manipulators with Acceleration Command

Zhiguan Huang , Zhengtai Xie , Long Jin , and Yuhe Li 

Research Article (11 pages), Article ID 2305459, Volume 2020 (2020)

Prediction of the RFID Identification Rate Based on the Neighborhood Rough Set and Random Forest for Robot Application Scenarios

Hong-Gang Wang, Shan-Shan Wang , Ruo-Yu Pan, Sheng-Li Pang, Xiao-Song Liu, Zhi-Yong Luo, and Sheng-Pei Zhou






Research Article (15 pages), Article ID 8831963, Volume 2020 (2020)

Distributed Integrated Sliding Mode-Based Nonlinear Vehicle Platoon Control with Quadratic Spacing Policy

Lei Zuo , Ye Zhang, Maode Yan, and Wenrui Ma

Research Article (9 pages), Article ID 4949520, Volume 2020 (2020)

Extended Kalman Filter-Based Approach for Autonomous Synchronization and Ranging in GPS-Denied Environments

Xiaobo Gu , Weiqiang Tan , Di Zhang , Yudong Lu , and Ruidian Zhan 

Research Article (7 pages), Article ID 8836584, Volume 2020 (2020)

Novel Robust Stability Criteria of Uncertain Systems with Interval Time-Varying Delay Based on Time-Delay Segmentation Method and Multiple Integrals Functional

Xing He , Li-Jun Song , Yu-Bin Wu, and Zi-Yu Zhou

Research Article (13 pages), Article ID 8841137, Volume 2020 (2020)

Locomotion Prediction for Lower Limb Prostheses in Complex Environments via sEMG and Inertial Sensors

Fang Peng , Cheng Zhang , Bugong Xu , Jiehao Li , Zhen Wang , and Hang Su 



Research Article (12 pages), Article ID 8810663, Volume 2020 (2020)

Adaptive Visually Servoed Tracking Control for Wheeled Mobile Robot with Uncertain Model Parameters in Complex Environment

Fujie Wang , Yi Qin , Fang Guo, Bin Ren, and John T. W. Yeow


Research Article (13 pages), Article ID 8836468, Volume 2020 (2020)

Disturbance-Observer-Based Fuzzy Control for a Robot Manipulator Using an EMG-Driven Neuromusculoskeletal Model

Longbin Zhang , Wen Qi, Yingbai Hu , and Yue Chen




Research Article (10 pages), Article ID 8814460, Volume 2020 (2020)

Load Parameter Identification for Parallel Robot Manipulator Based on Extended Kalman Filter

Shijie Song, Xiaolin Dai, Zhangchao Huang, and Dawei Gong 


Research Article (12 pages), Article ID 8816374, Volume 2020 (2020)

Adaptive Robust Dynamic Surface Integral Sliding Mode Control for Quadrotor UAVs under Parametric Uncertainties and External Disturbances

Ye Zhang , Ning Xu, Guoqiang Zhu , Lingfang Sun, Shengxian Cao , and Xiuyu Zhang



Research Article (20 pages), Article ID 8879364, Volume 2020 (2020)

Acceleration Level Control of Redundant Manipulators with Physical Constraints Compliance and Disturbance Rejection under Complex Environment

Jinglun Liang, Yisheng Rong, Guoliang Ye, Xiaoxiao Li , Jianwen Guo, and Zhenzhen He

Research Article (14 pages), Article ID 8844209, Volume 2020 (2020)




Bottom Detection Method of Side-Scan Sonar Image for AUV Missions

Huapeng Yu , Ziyuan Li, Dailin Li, and Tongsheng Shen 



Research Article (9 pages), Article ID 8890410, Volume 2020 (2020)

Contents

Analysis on the Impact of Data Augmentation on Target Recognition for UAV-Based Transmission Line Inspection

Chunhe Song , Wenxiang Xu, Zhongfeng Wang, Shimao Yu, Peng Zeng , and Zhaojie Ju 
Research Article (11 pages), Article ID 3107450, Volume 2020 (2020)

Multimodal Multiobject Tracking by Fusing Deep Appearance Features and Motion Information

Liwei Zhang , Jiahong Lai, Zenghui Zhang, Zhen Deng , Bingwei He, and Yucheng He
Research Article (10 pages), Article ID 8810340, Volume 2020 (2020)

Research Article

High-Accuracy Real-Time Fish Detection Based on Self-Build Dataset and RIRD-YOLOv3

Wenkai Wang, Bingwei He , and Liwei Zhang 

School of Mechanical Engineering and Automation, Fuzhou University, Fuzhou 350000, China

Correspondence should be addressed to Bingwei He; mebwe@fzu.edu.cn

Received 19 July 2020; Revised 5 January 2021; Accepted 4 March 2021; Published 8 April 2021

Academic Editor: Xin Dong

Copyright © 2021 Wenkai Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

To better detect fish in an aquaculture environment, a high-accuracy real-time detection model is proposed. An experimental dataset was collected for fish detection in laboratory aquaculture environments using remotely operated vehicles. To overcome the inaccuracy of the You Only Look Once v3 (YOLOv3) algorithm in underwater farming environment, a suitable set of hyperparameters was obtained through multiple sets of experiments. Then, a real-time image recovery algorithm is applied before YOLOv3 to reduce the effects of both noise and light on images whilst keeping the real-time capability, leading to a mean average precision of 0.85 and frame rate of 17.6 fps, respectively. Finally, compared with the base detection model using only the YOLOv3 algorithm, the enhanced detection model presented results in a reduction of miss detection rate from 23% to only 9% across different environments and with the detection accuracy of the target in different environments being improved from 8% to 37%.

1. Introduction

Recently, ocean engineering and research have increasingly relied on underwater images captured by autonomous underwater vehicles (AUVs) and remotely operated vehicles (ROVs) [1]. However, since the collection of underwater datasets is more difficult than that for onshore datasets, there are few generally accessible datasets for underwater creatures, and public datasets for freshwater creatures are even rarer. In addition, underwater images usually suffer from various types of degeneration, such as low contrast, color casts, and noise, due to wavelength-dependent light absorption and scattering as well as the effects of low-end optical imaging devices [2]. To obtain much higher quality underwater images, a number of advanced methods have been designed and used. For example, Gray World [3] and White Patch [4] are used in color correction. Fang et al. proposed a single image enhancement approach based on image fusion strategy to enhance the underwater image [5]. Li et al. presented a systematic underwater image enhancement method including underwater image dehazing algorithms and a contrast enhancement algorithm for high-quality underwater images [6], and Hitam et al. utilized the

contrast limit adaptive histogram equalization (CLAHE) to enhance the contrast [7]. Recently, Peng and Cosman proposed a depth and background light estimation method for underwater scenes based on image blurriness and light absorption, which can be used to restore and enhance underwater images [8]. Besides, many studies try to address the issue from the physical level. Typically, Schechner and Karpel employed a polarizer in front of their camera [9]. These methods work well for underwater image processing, but few of them took the degeneration model into account or the proposed models are too complex to work in real-time. Moreover, most existing algorithms are lacking in the capability of self-adaption and self-adjustment, which are important for a robot working in a changing and complex underwater environment.

Instead of traditional target detection, artificial neural networks (ANNs) can be used to detect fish in images, and some methods have shown promise for real-time performance, such as Faster R-CNN [10], R-FCN [11], SSD [12], and YOLO series [13–15], amongst others. Among them, YOLOv3 performs well in both real-time and in terms of mean average precision (mAP). However, YOLOv3 just performs well in clear waters. When in dim and turbid

waters, YOLOv3 loses almost all its original land-based advantages.

In this paper, an experimental dataset was collected, which solves the problem of the lack of datasets for fish in aquaculture environments. Furthermore, a set of suitable hyperparameters were obtained for the dataset through multiple sets of experiments, reducing training time and improving the detection accuracy. To improve the performance of YOLOv3, a high-accuracy real-time fish detection algorithm was proposed, named RIRD-YOLOv3. This paper is organized as follows: firstly, this paper introduces the laboratory acquisition of the dataset, the RIRD-YOLOv3 algorithm, and the matching of hyperparameters; secondly, we discuss the analytical results of the experiments; finally, we present the conclusion. In this paper, the proposed algorithm is tested, and it performs well.

2. Dataset Collection and RIRD-YOLOv3 Algorithm

2.1. Acquiring the Dataset. Deep learning [16] requires a large amount of training samples, and the amount of data used will directly affect the detection accuracy of fish for this application. However, the problem faced by the fish dataset is that its open source dataset is very scarce and does not meet the training needs of grass carp detection models.

To solve the problem of the lack of grass carp dataset in the breeding environment, in this paper, through a field investigation, a simulated grass carp breeding environment is established in the laboratory. Based on the growth environment of grass carp, the length, width, and height of the pool are set to 600 cm, 450 cm, and 250 cm. The pond can simulate the real grass carp breeding environment. The experiment site is shown in Figure 1.

The content of the sample of the dataset includes adult grass carp and robot fish. Among them, the robotic fish is a bionic robot purchased in the laboratory. It matches the shape and characteristics of the real fish. The purpose of placing it in the dataset is to verify whether the classification performance of the model obtained by the algorithm will be affected. Each of the sample image contains zero or more instances of fish. As a result, each image could contain from zero to multiple annotations. This method enriches the types of dataset samples and can be used to verify the classification characteristics of the detection model. The sample content of the dataset is shown in Figure 2.

To fully simulate the impact of light on grass carp in the breeding environment and to fully collect images of grass carp under different light environments, this paper accomplishes this by changing the lighting conditions at different time periods. The specific implementation is shown in Table 1.

According to Table 1, after using ROV to collect grass carp images, through sorting out, it is found that three types of images are useless, as shown in Figure 3. The three types of images shown in Figure 3 do not contain much in the dataset, but they still affect the accuracy of the subsequent detection model. Therefore, in order to achieve the purpose



FIGURE 1: The experiment site.

of improving the accuracy of the detection model, this paper removes these three types of images manually.

A standard dataset should include a training set and a testing set. The training set and the testing set are mutually exclusive. The training set is used to obtain an excellent detection model, and the testing set is used to test the performance of the model. After removing the above three types of useless samples, the dataset contains 3069 images. In order to ensure the performance of the detection model, this paper uses the ‘reserve method’ to randomly divide the images of the dataset into a training set and a testing set according to 7:3 ratio. After the division of “reserve method,” the training set contains 2148 images and the testing set 921 images. Besides, in this paper, the training set is classified according to the three types of lighting conditions shown in Table 1, and the number of the three types of samples is NL 443, NOL 1228, and 477 NORL. The dataset named the Grass Carp Dataset before Restoration (GCDBR) is composed of original images. The examples of GCDBR are shown in Figure 4.

Besides, to improve the detection accuracy, it is necessary to separate out the fish from the environment. Usually fish have a similar color to the environment to protect themselves. Therefore, a large number of optical images of the underwater environment were collected and labelled as ‘negative sample.’

2.2. Dataset Labelling. The dataset needs to be annotated to accomplish and validate the goal of classification and detection on the images. In this paper, labelling software is used to create annotations of respective classes for the images in the dataset based on the PASCAL VOC [17] standard labelling format. Each annotation is created by drawing a bounding box around the object of interest belonging to one of the classes and assigning the bounding box and the class label associated with it. For simplicity, in this paper, axis-aligned bounding boxes are used as described in the PASCAL VOC dataset paper [17]. The examples of annotated images are shown in Figure 5.

2.3. The RIRD-YOLOv3 Algorithm. The water quality of the grass carp farming environment is turbid, and due to the absorption of light by the water, the scattering effect, and

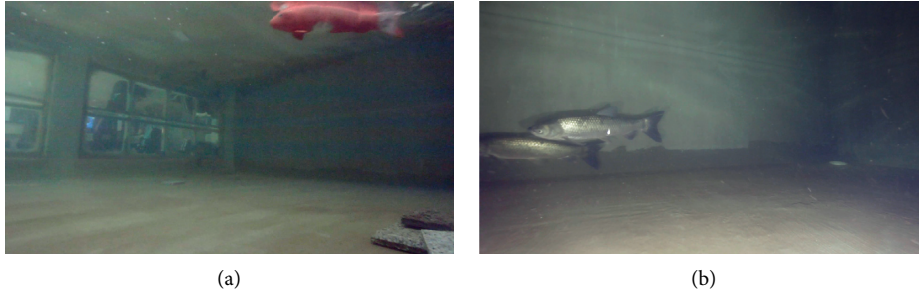


FIGURE 2: The sample content of the dataset: (a) robot fish; (b) grass carp.

TABLE 1: Different light conditions and implementation methods of grass carp growth.

Lighting conditions Reasons and method of implementation	Natural light (NL)	Natural and outdoor light (NOL)	Natural, outdoor, and ROV light (NORL)
Reason	Sufficient sunlight, ROV without ROV light can clearly collect grass carp images	With sunlight but weak, such as cloudy, rainy, and evening	Late at night or without any light
Method of implementation	Just having sunlight, without any additional auxiliary light source	Sunlight is the main light source; outdoor light source is the supplement	Outdoor light and ROV light are main light sources

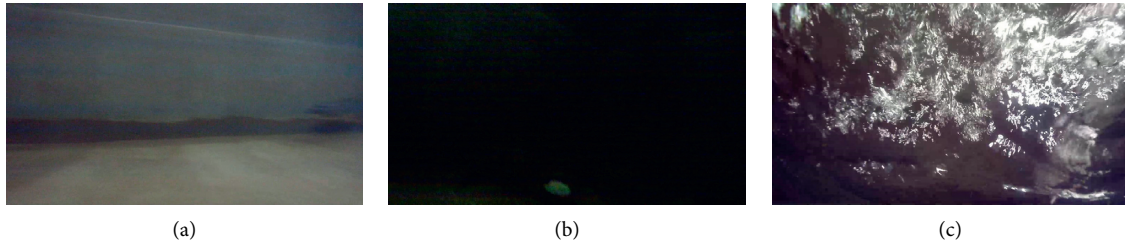


FIGURE 3: Cull images: (a) sports afterimage; (b) no target; (c) too many bubbles.

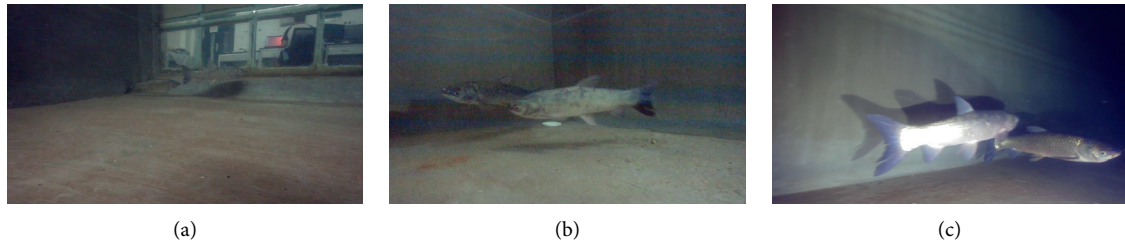


FIGURE 4: The examples of GCDBR: (a) NL; (b) NOL; (c) NORL.



FIGURE 5: Example images within the labelled dataset: (a) grass carp; (b) robot fish.

the uneven illumination of the ROV, the quality of the image will deteriorate and the grass carp cannot be distinguished by the naked eyes. The examples of images of grass carp sample of low quality are shown in Figure 6.

To overcome this problem, Chen et al. provided three parameters, related to underwater image degradation and color correction, by presearching in the first frame of image sequences using an artificial fish school algorithm [18]. The core of image restoration is a Wiener Filter in frequency domain as follows:

$$V_{\text{orig},C}(u, v) = \left[\frac{H(u, v)}{H(u, v)^2 + R} \right] V_{\text{deg},C}(u, v), \quad (1)$$

where $V_{\text{orig},C}$ represents one channel of the original image; $V_{\text{deg},C}$ represents one channel of the degraded image due to underwater scattering and abortion; R is the reciprocal of signal-to-noise ratio and was implemented to restrict scattering; $H(u, v)$ is originated as a general image degradation model in turbulent media [18] expressed by

$$H(u, v) = e^{-k(u^2+v^2)^{(5/6)}}, \quad (2)$$

where k is a crucial parameter related to the depth of water and the distance from the camera.

After Wiener Filter is applied, color correction is implemented on the image by gamma factor as follows:

$$I_{\text{corrected},C} = I^\gamma. \quad (3)$$

At this point, R , K , and γ have been introduced. To obtain a reliable combination of these three parameters, we employ a quality index of the restored image expressed as follows:

$$Q = \frac{\alpha\beta}{1 + \eta}, \quad (4)$$

where α is a haze indicator, describing the level of haze by gradient computed by the modified Tenegrad evaluation, given as follows:

$$\alpha = \frac{1}{W} \sum_{i=0}^M \sum_{j=0}^N \sum_{k=0}^7 |\text{Gradient}(V_g(i, j), k)|^2, \quad (5)$$

where $M \times N$ is the size of an input image; V_g is a grayscale map, and orientations of gradient are regulated as $k \times 45^\circ$. This indicator takes the textural feature and edge feature into consideration. Generally, a higher value of α reflects a clearer restored image.

β is a contrast indicator, which is calculated by histogram distribution in RGB channels, representing the image contrast as defined in the following equation:

$$\beta = \frac{1}{MN} \sum_{C \in \{R, G, B\}} \sqrt{\sum_{i=0}^{255} (h_C(i) \times i - \mu_C)^2}, \quad (6)$$

where $h_C(i)$ stands for the data of histogram curves at gray level i for channel C and μ_C shows the average of histogram

curves of channel C . Theoretically, objects can be distinguished more easily with a higher value of β .

η is an imbalance indicator, which denotes the level of color correction as follows:

$$\eta = |\mu_r - \mu_b| + |\mu_r - \mu_g| + |\mu_b - \mu_g|. \quad (7)$$

Clearly, η diminishes along with a better color correction.

A test result of a deep sea image is shown in Figure 7. Clearly, the method is effective in contrast and color correction, and it takes only 17.5 milliseconds for each frame on average. In addition, the amount of relevant information in the restored image has been retained to a large degree, such as color information, texture and edge information, and illumination information.

2.4. Matching of Hyperparameters. To train the ANN efficiently and well to predict the desired outcome, the hyperparameters of the network should be properly determined. For the various values of number of epochs, momentum, learning rate, and batch size, a grid search was performed to optimize the hyperparameters. All possible sets of values described in Table 2 were tested to train the network. Then, after training the network using each set of values, the sensitivity was assessed using 100 test images. Then, the values that maximize the quality of the network were adopted for the hyperparameters.

3. Results and Discussion

3.1. Experimental Platform and GCDAR. First of all, the experiment in this paper starts with image restoration. The specific implementation method is to restore the sample image of GCDAR to obtain the grass carp dataset after restoration (GCDAR). The comparison of sample images of grass carp dataset before and after restoration is shown in Figure 8. The appearance of the target in the image is different due to the influence of the light. In Figure 8(a), since the color of the target is similar to the background environment, it becomes difficult for the human eye to detect the target. In Figure 8(b), under the action of the outdoor auxiliary light, the underwater image halo is enhanced, and the target is very blurred due to the presence of water mist. In Figure 8(c), when natural light and ambient light do not work, the self-contained light source of the ROV is used, but the observation of the underwater target is still difficult due to the limited light strength. After the image is restored, the target in the restored image in Figure 8 becomes distinctly clear.

When training the model, our device is NVIDIA Tesla M40 with graphics card of 12 GB. When performing image recovery and detection, the actual ranges of K , R , and γ are $[10^{-7}, 1.5 \times 10^{-4}]$, $[0.01, 15]$, and $[0.4, 1]$ all of which are normalized as $[0.1, 150]$. The time-related data are obtained with $[640 \times 360]$ pixel-size images, and the processor was a Core i5-7300HQ CPU with the main frequency increased up to 2.5 GHz.

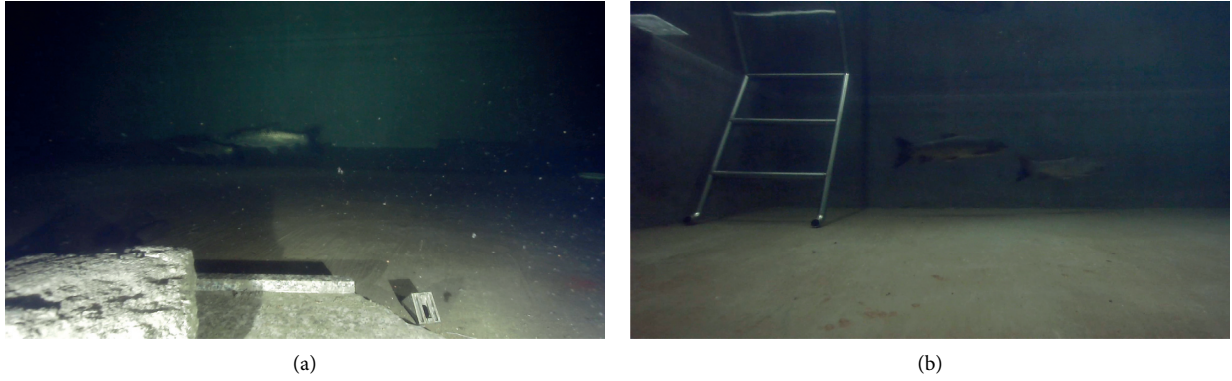


FIGURE 6: Examples of image of grass carp sample of low quality.



FIGURE 7: The performance of image restoration: (a) before restoration; (b) after restoration.

3.2. *Training Result.* The best hyperparameters for training are shown in Table 3. Figure 9 is a graph showing the relationship between batches and average loss. For each batch, 64 images are randomly selected and used to train the ANN. Since the number of samples is limited, each image is used multiple times. The graph shows that the average loss is almost reduced to 0 as batches progress. A total of 30200 epochs were run, and it took 48 hours to complete the training. Compared with the initial training parameters, the best hyperparameters reduce the training time by at least 48 hours.

To meet real-time requirements, through testing, the frame rate of GCDAR's model is shown in Table 4.

In addition, evaluation of the trained network is performed by taking our validation dataset consisting of 300 images and executing detection on it using the trained model.

The metrics used to evaluate the object detection are as follows:

- (a) *mAP*. This is the mean of the interpolated average precision across all the classes in the dataset used for object detection.
- (b) *IOU*. This is the ratio of the area of intersection to the area of union of the predicted bounding box and the corresponding maximally matched ground truth box as defined in the following equation:

$$IOU = \frac{\text{Area}(\text{PredictedBox} \cap \text{GroundtruthBox})}{\text{Area}(\text{PredictedBox} \cup \text{GroundtruthBox})}. \quad (8)$$

TABLE 2: Tested values in grid search of hyperparameters.

Number of epochs	1000, 2000, . . . , 40000
Momentum	0.6, 0.7, 0.8, 0.9
Learning rate	0.00001, 0.00002, 0.00003, 0.00004
Batch size	8, 16, 32, 64

3.3. *Comparison of Unrecovered and Recovered Images.* To verify the validity of RIRD-YOLOv3 in different environments, two new evaluation parameters were proposed, namely, missed detection rate (MDR) and target detection accuracy (TDA) in a single environment. In addition, to demonstrate the advantages of the method, a comparative experiment of image detection capability before and after restoration is proposed and conducted.

The MDR and the TDA were also tested (as defined in equations (9) and (10)). The results are shown in Table 5. As seen from Table 5, the restored image's MDR is reduced from 8.9% to 21.7% compared with the image before restoration. The restored image's TDA is increased from 7.8% to 36.8% compared with the image before restoration. The large reduction in the rate of missed detection indicates the effectiveness of the RIRD-YOLOv3 algorithm. The improvement of detection accuracy in different environments shows that the model has generally excellent performance. Figure 10 shows the IOU contrast between the prerecovery image and the restored image. In the original image, the target detection showed missed detection and false detection. However, in the restored image, both missed detection

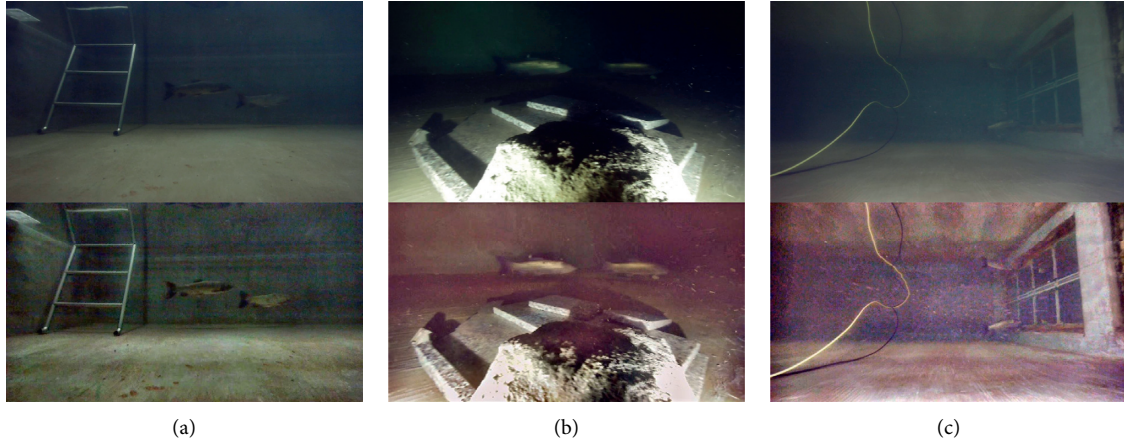


FIGURE 8: Example of the comparison of sample images of grass carp dataset before and after restoration: (a) NL; (b) NOL; (c) NORL.

TABLE 3: Hyperparameters to train.

Parameter	Number of epochs	Momentum	Batch size	Learning rate
Numerical value	30200	0.9	64	0.00001

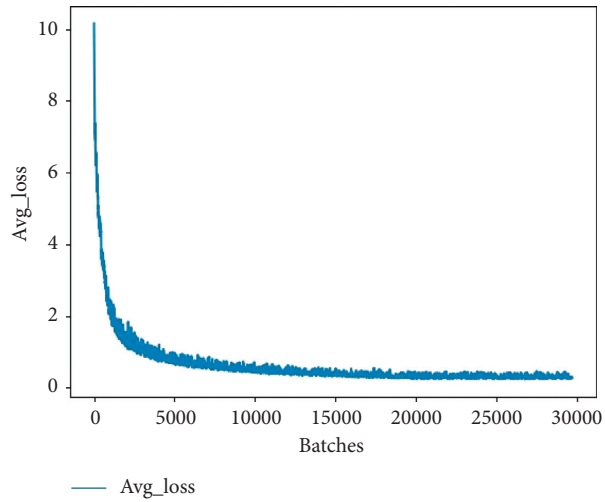


FIGURE 9: The average loss function of training.

TABLE 4: Performance of the test model based on the RIRD-YOLOv3 algorithm.

Metric	Frame rate	mAP	IOU
Numerical value	17.6	0.85	0.82

TABLE 5: Comparison table of missed detection rate and target detection accuracy in three environments.

	Miss detection rate		Detection accuracy	
	Before recovery (%)	After recovery (%)	Before recovery (%)	After recovery (%)
NL	12.1	3.2	89.3	97.1
NOL	26.4	12.5	62.9	99.71
NORL	33.3	11.6	73.1	86.7

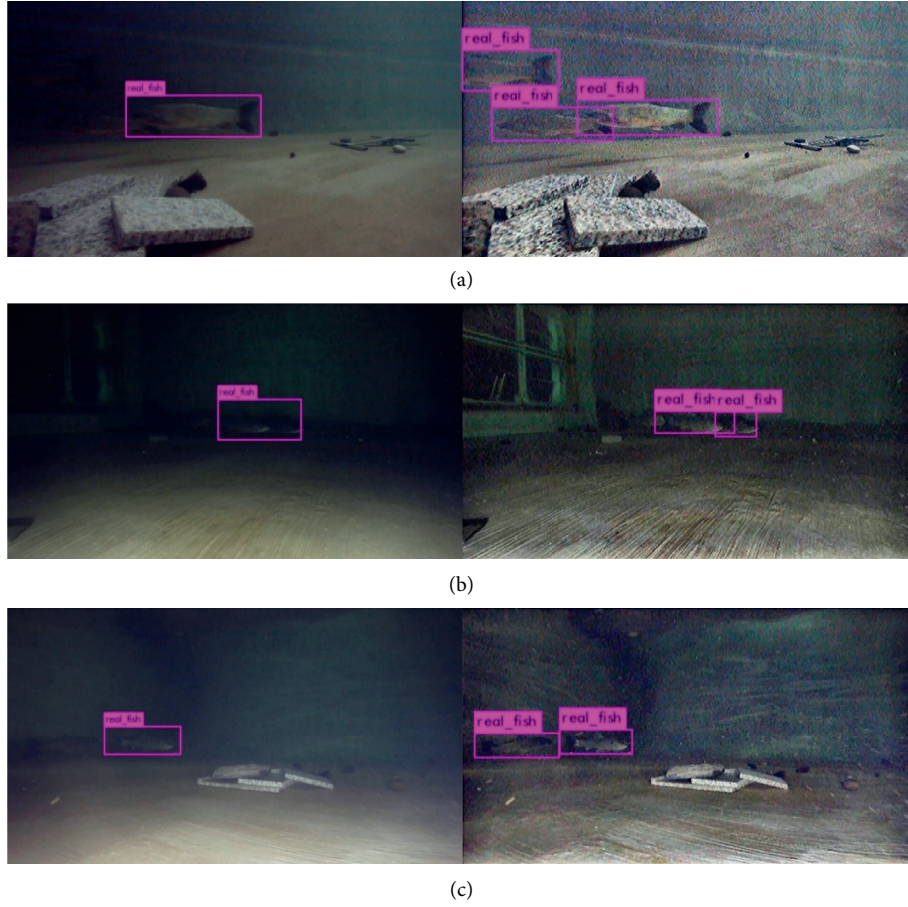


FIGURE 10: Comparison of fish detection between original image and restored image in three environments: (a) NL; (b) NOL; (c) NORL.

and false detection are reduced.

$$\text{MDR} = \frac{M}{M + N} \times 100\%, \quad (9)$$

where M is the missing detection image and N is the no missing detection image.

$$\text{TDA} = \frac{t_1 + t_2 + t_3 + \dots + t_n}{n} \times 100\%, \quad (10)$$

where t is the detection accuracy of a single target in test images and n is the number of targets.

4. Conclusions

In this paper, we presented a high-accuracy real-time fish detection algorithm, called RIRD-YOLOv3. It is able to solve the problem of image blur and noise caused by processing in an underwater environment. In addition, a set of suitable hyperparameters is provided for laboratory freshwater aquaculture environmental dataset. When using the hyperparameters is discovered, the experimental results show that the training time for the dataset is reduced by 48 hours. During testing, the frame rate of RIRD-YOLOv3 was 17.6 FPS and the model's mAP is 0.85. Prerecovery and postrecovery images were contrasted in three environments, and the miss detection rate and detection accuracy

are reduced from 23% to 9% and increased from 13% to 37%, respectively. Therefore, overall RIRD-YOLOv3 has demonstrated excellent performance for this type of environment.

The RIRD-YOLOv3 algorithm is of important significance for underwater target detection applications. It can be applied to underwater submersibles such as ROV and AUV. It has potential for further contribution to the exploration of underwater resources.

Data Availability

This paper proposes the laboratory acquisition of the dataset, which is published on CSDN, and data can be obtained from the following links: <https://download.csdn.net/download/qie123zi456/12328847>, <https://download.csdn.net/download/qie123zi456/12328855>, <https://download.csdn.net/download/qie123zi456/12328879>, <https://download.csdn.net/download/qie123zi456/12328894>, and <https://download.csdn.net/download/qie123zi456/12328901>, <https://download.csdn.net/download/qie123zi456/12328911>.

Conflicts of Interest

The authors declare that they have no conflicts of interest regarding the publication of this paper.

References

- [1] M. Bryson, M. Johnson-Roberson, O. Pizarro, and S. B. Williams, "True color correction of autonomous underwater vehicle imagery," *Journal of Field Robotics*, vol. 33, no. 6, pp. 853–874, 2016.
- [2] C. Li, J. Guo, and C. Guo, "Emerging from water: underwater image color correction based on weakly supervised color transfer," *IEEE Signal Processing Letters*, vol. 25, no. 3, pp. 323–327, 2018.
- [3] G. Buchsbaum, "A spatial processor model for object colour perception," *Journal of the Franklin Institute*, vol. 310, no. 1, pp. 1–26, 1980.
- [4] E. Provenzi, C. Gatta, M. Fierro, and A. Rizzi, "A spatially variant white-patch and gray-world method for color image enhancement driven by local contrast," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 10, pp. 1757–1770, 2008.
- [5] S. Fang, R. Deng, Y. Cao, and C. Fang, "Effective single underwater image enhancement by fusion," *Journal of Computers*, vol. 8, no. 4, pp. 904–911, 2013.
- [6] C.-Y. Li, J.-C. Guo, R.-M. Cong, Y.-W. Pang, and B. Wang, "Underwater image enhancement by dehazing with minimum information loss and histogram distribution prior," *IEEE Transactions on Image Processing*, vol. 25, no. 12, pp. 5664–5677, 2016.
- [7] M. S. Hitam, E. A. Awalludin, W. N. J. H. W. Yussof, and Z. Bachok, "Mixture Contrast Limited Adaptive Histogram Equalization for Underwater Image Enhancement," in *Proceedings of the 2013 International Conference on Computer Applications Technology (ICCAT)*, pp. 1–5, Sousse, Tunisia, January 2013.
- [8] Y.-T. Peng and P. C. Cosman, "Underwater image restoration based on image blurriness and light absorption," *IEEE Transactions on Image Processing*, vol. 26, no. 4, pp. 1579–1594, 2017.
- [9] Y. Y. Schechner and N. Karpel, "Clear underwater vision," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Washington, DC, USA, June 2004.
- [10] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *Advances in Neural Information Processing Systems*, vol. 39, pp. 91–99, 2015.
- [11] J. Dai, Y. Li, K. He, and J. Sun, "R-FCN: object detection via region-based fully convolutional networks," *Advances in Neural Information Processing Systems*, vol. 29, pp. 379–387, 2016.
- [12] W. Liu, D. Anguelov, D. Erhan et al., "SSD.: single shot MultiBox detector," in *Proceedings of the European Conference on Computer Vision*, pp. 21–37, Cham, Switzerland, October 2016.
- [13] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: unified, real-time object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 779–788, Las Vegas, NV, USA, June 2016.
- [14] J. Redmon and A. Farhadi, "YOLO9000: better, faster, stronger," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7263–7271, Honolulu, HI, USA, July 2017.
- [15] J. Redmon and A. Farhadi, "YOLOv3: an incremental improvement," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Salt Lake City, UT, USA, June 2018.
- [16] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.
- [17] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL visual object classes (VOC) challenge," *International Journal of Computer Vision*, vol. 88, no. 2, pp. 303–338, 2010.
- [18] X. Chen, Z. Wu, J. Yu, and L. Wen, "A real-time and unsupervised advancement scheme for underwater machine vision," in *Proceedings of the 7th Annual IEEE International Conference on Cyber Technology in Automation, Control, and Intelligent Systems (CYBER)*, pp. 271–276, Hawaii, USA, July 2017.

Research Article

Robust Control for the Suspension Cable System of the Unmanned Helicopter with Sensor Fault under Complex Environment

Rong Mei 

College of Criminal Investigation, Nanjing Forest Police College, Nanjing 210023, China

Correspondence should be addressed to Rong Mei; meirongnuuaa@163.com

Received 13 September 2020; Revised 11 November 2020; Accepted 26 February 2021; Published 11 March 2021

Academic Editor: Ning Wang

Copyright © 2021 Rong Mei. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Aiming at the suspension cable system of an unmanned helicopter with sensor fault under complex environment, this paper studies the robust antiswing tolerant control scheme. To suppress the swing of the hanging load when the unmanned helicopter is in the forward flight state, a nonlinear line motion model is firstly established. Considering the sensor fault of the unmanned helicopter, a sensor fault estimator is developed. By using the fault estimator output, the robust antiswing tolerant controller is proposed using the backstepping technique and sliding mode control method. Under the designed robust antiswing tolerant controller, the desired tracking control performance can be obtained and the swing angle of the load is guaranteed small under the sensor fault. Furthermore, the closed-loop system stability is analyzed by using the Lyapunov technique. Simulation studies are given to show the efficiency of the designed robust antiswing control strategy.

1. Introduction

Due to the unique characteristics such as vertical takeoff, vertical landing, good maneuverability, high work capability in complex environments, and hovering low-speed flying, the helicopter has been widely used in different practical areas during the past several decades [1]. Specially, considering the superiorities of long flight distance, high altitude, strong robustness, and large load, the medium-scale helicopter receives great attention [2]. By using the ability of long flight distance and large load of medium-scale helicopter, we can carry out the external transport by using the suspension cable, which is one of the main applications of medium-scale helicopters in the military and civil application area [3]. Suspension flight can transport bulk goods, which does not need to consider the load capacity of helicopters and appearance of the goods. Compared with the helicopter free flying state, helicopter flight with additional load will increase the load gravity load and load disturbance. Thus, it is necessary to consider the influence of the hanging load on the system. The characteristics of the low-speed stability were analyzed for a helicopter with a sling load in [4]. In [5], the

flight dynamics were established for the articulated rotor helicopter by considering an hanging load.

Because the working environment is complicated and dynamically changeable, it is a challengeable task to develop a good control law for a helicopter suspension cable system [6–8]; for example, the medium-scale helicopter can be used to monitor and suppress the forest fire. However, the complex and changeable forest environment will increase the probability of a crash for the medium-scale helicopter. To avoid human sacrifice in the work process, the suspension cable system of an unmanned helicopter is developed in recent years. In [9], to deliver airborne cargo precisely, an active control scheme was designed for an unmanned helicopter with a slung load. An antiswing controller was designed for the unmanned helicopter with slung load by nonlinear path tracking in [10]. In [11], an adaptation controller was developed for autonomous helicopter slung load operations. A nonlinear controller design was developed for a helicopter slung load system in [12]. In [13], an adaptation backstepping controller by using prescribed performance method was proposed for carrier used unmanned aerial vehicle. Furthermore, in order to improve the security and the economic efficiency, the efficient controller

should be further designed for the suspension cable system of an unmanned helicopter with sensor fault.

Sensor fault is an important fault of the practical systems due to the external circuit fault and mechanical fault of sensor and so on [14–18]. In general, there are four main types of sensor faults which include the complete failure fault, the fixed deviation fault, the drift deviation fault, and the precision reduction fault [19]. Now, there are many research results for the tolerant control problem of the various system with faults and unknown disturbance [20–27]. In [28], the local stabilization was studied for the Takagi–Sugeno (T-S) fuzzy discrete-time time-delay system in the presence of sensor fault. The sensor fault diagnosis technique and fault-tolerant control methods were developed for stochastic time-delayed control systems in [29]. In [30], the observer design method of the fault estimation was given for descriptor switched systems with sensor and actuator faults. The robust sensor fault estimator was studied for the continuous interconnected system in [31]. In [32], the tolerant control strategy was designed for an autonomous vehicle with proprioceptive sensors' fault. A fault-tolerant control was developed for the linear system with sensor and actuator faults by using observer-based \mathcal{H}_∞ method in [33]. However, the fault-tolerant control methods need further development to improve their safety of various aircrafts with sensor fault.

In the past several years, various tolerant control schemes of the aircrafts with sensor fault have been studied [34]. In [35], the detection and recovery method was studied for the satellite attitude control system with sensor fault. Aiming at the hypersonic flight vehicle with multisensor faults, a nonlinear fault-tolerant control scheme was developed in [36]. In [37], an observer-based tolerant control scheme was designed for unmanned aerial vehicle with sensor faults. The reconstruction method was proposed for the aircraft with sensor fault under disturbances in [38]. In [39], the estimation method was studied for flight control systems with sensor fault based on the identification of aerodynamic parameters. In [40], a fuzzy adaptive tolerant control scheme was proposed for a quadrotor unmanned aerial vehicle with nonlinear sensor fault. However, the fault-tolerant control methods are rare for the suspension cable system of an unmanned helicopter with sensor fault which needs to be further studied.

Motivated by above analysis, the scheme of a robust antiswing control is studied for the suspension cable system of an unmanned helicopter with sensor fault to improve its reliability. The key innovations of this paper are stated as follows:

- (1) A design method of estimator is proposed to solve the estimation problem of sensor fault in suspension cable system of an unmanned helicopter.
- (2) The robust antiswing control law is developed for the suspension cable system of an unmanned helicopter by using sensor fault estimator, backstepping technology, and the desired tracking trajectory.
- (3) The closed-loop system stability of an unmanned helicopter with sensor faults is strictly analyzed under the robust antiswing control scheme.

The organization of this paper is described as follows. Section 2 gives the problem description, and the system model is introduced. The sensor fault estimator is proposed for the antiswing control in Section 3. Section 4 designs the robust antiswing control scheme based on sensor fault estimator. In Section 5, simulation results and analysis are given to show the effectiveness of the studied antiswing control scheme, and some conclusions are drawn in Section 6.

2. Problem Description

In this paper, the load uses a single-point hanging method to connect with the unmanned helicopter, which is the most widely used one because of compact and simple structure. To simplify the design of the controller, only the linear motion of the suspension cable system of the unmanned helicopter suspension system is considered, as shown in Figure 1 [3].

To establish the line motion model of the suspension cable system of an unmanned helicopter, we assume that the load swing amplitude at the initial time is zero; the unmanned helicopter sling is assumed to be massless, always straightened during flight, and will not come loose; the distance between the sling point and the unmanned helicopter particle is ignored; unmanned helicopter and hanging object are rigid bodies, without considering elastic deformation, and the load is regarded as a particle without considering the shape of the load. Ignore the influence of rotor airflow on unmanned helicopter and load movement and ignore the external disturbances, such as the wind, during the movement [3].

Under above assumptions, by using the Lagrange method, the nonlinear line motion model of the suspension cable system of an unmanned helicopter can be described as follows [3]:

$$\begin{cases} \ddot{x} = \frac{m_l \sin \theta (L \dot{\theta}^2 + g \cos \theta) + T}{M_h + m_l \sin^2 \theta}, \ddot{\theta} = \frac{-(M_h + m_l)g \sin \theta - m_l L \dot{\theta}^2 \sin \theta \cos \theta - T \cos \theta}{L(M_h + m_l \sin^2 \theta)}, \end{cases} \quad (1)$$

where T is the lift of unmanned helicopter which is the control input. x is the unmanned helicopter displacement,

and θ is the swing angle of hanging load which are system outputs. M_h is the unmanned helicopter mass, m_l is the

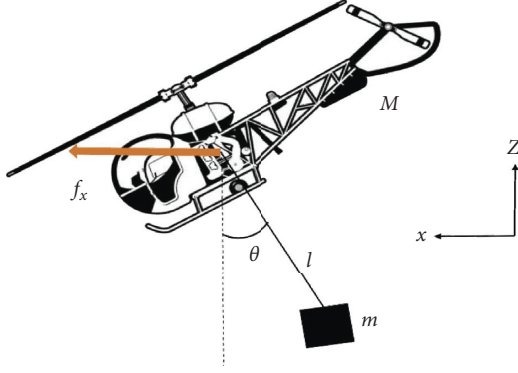


FIGURE 1: The straight line motion of the suspension cable system of the unmanned helicopter with single point hanging.

suspension load mass, g is the gravity acceleration, and L is the length of suspension cable.

Define $X = [cx_1 \ x_2 \ x_3 \ x_4]^T = [cx \ \theta \ \dot{x} \ \dot{\theta}]^T$ and $u = T$. Then, system (1) can be expressed as

$$\begin{cases} \dot{x}_1 = x_3, \\ \dot{x}_2 = x_4, \\ \dot{x}_3 = f_1(X) + g_1(X)u, \\ \dot{x}_4 = f_2(X) + g_2(X)u, \end{cases} \quad (2)$$

where x_1 denotes the helicopter displacement x , x_2 is the swing angle of hanging load θ , x_3 is the forward flight speed of helicopter \dot{x} , x_4 denotes the angular velocity $\dot{\theta}$ of the swing angle of hanging load θ , and u is the lift T . $f_1(X)$, $g_1(X)$, $f_2(X)$, and $g_2(X)$ are given by

$$\begin{aligned} f_1(X) &= \frac{m_l \sin \theta (L\dot{\theta}^2 + g \cos \theta)}{M_h + m_l \sin^2 \theta}, \\ g_1(X) &= \frac{1}{M_h + m_l \sin^2 \theta}, \\ f_2(X) &= \frac{-(M_h + m_l)g \sin \theta - m_l L \dot{\theta}^2 \sin \theta \cos \theta}{L(M_h + m_l \sin^2 \theta)}, \\ g_2(X) &= \frac{-\cos \theta}{L(M_h + m_l \sin^2 \theta)}. \end{aligned} \quad (3)$$

Define $X_1 = [x_1, x_2]^T$ and $X_2 = [x_3, x_4]^T$. Assume that the system output is y and $y = X_1$. Then, we have

$$\begin{cases} \dot{X}_1 = X_2, \\ \dot{X}_2 = F(X) + G(X)u, \\ y = X_1, \end{cases} \quad (4)$$

where $F(X) = [f_1(X), f_2(X)]^T$ and $G(X) = [g_1(X), g_2(X)]^T$.

In this paper, the forward speed \dot{x} is measured by airspeed head and the swing angle of hanging load θ is surveyed using an optical camera. We assume that accuracy losses of the airspeed head sensor and the optical camera sensor are

considered in this study. The loss of accuracy of two sensors is given as follows [40]:

$$\chi = X_2 + D(t), \quad \forall t \geq t_f, \quad (5)$$

where χ is the measured value of the airspeed head sensor and the optical camera sensor with fault $D(t)$ and t_f is the sensor fault occurring time.

The control design goal of this paper is to develop a robust antiswing control plan such that the system actual output y can track the required one y_c and the tracking error $e_1 = y - y_c$ is convergent when the suspension cable system of the unmanned helicopter (3) suffers from the sensor fault.

To promote the design of the robust antiswing control scheme for the suspension cable system of the unmanned helicopter with sensor fault, some assumptions and lemmas are required.

Assumption 1. (see [40]). For the sensor fault D , there exist corresponding inequality conditions $\|D\| \leq \tau_0$, $\|\dot{D}\| \leq \tau_1$, and $\|\ddot{D}\| \leq \tau_2$ with $\tau_i > 0, i = 0, 1, 2$.

Assumption 2. (see [2]). All states of the studied system are measurable and available. The desired system tracking signal y_c and its derivative are always bounded. Furthermore, there exists an unknown positive constant Δ_0 which renders $C_0 = \{(y_c, \dot{y}_c, \ddot{y}_c): \|y_c\|^2 + \|\dot{y}_c\|^2 + \|\ddot{y}_c\|^2 \leq \Delta_0\}$.

Lemma 1. (see [41]). For the any bounded initial conditions, if there is a C^1 positive and continuous Lyapunov function $V(x)$ which satisfies $\rho_1(\|x\|) \leq V(x) \leq \rho_2(\|x\|)$ and makes that $\dot{V}(x) \leq -k_1 V(x) + k_2$, where $\rho_1, \rho_2: \mathbb{R}^n \rightarrow \mathbb{R}$ are class K functions and k_1, k_2 are positive constants, then its solution $x(t)$ is uniformly bounded.

Remark 1. In this paper, the sensor fault is considered for the suspension cable system of an unmanned helicopter. As well known, the sensor fault and its derivatives should be bounded in the practical system. If they are not bounded, the system controllability cannot be guaranteed. Thus, Assumption 1 is reasonable for the helicopter suspension cable system. On the other hand, to design the robust antiswing control law for the suspension cable system of an unmanned helicopter, all states are needed. Thus, we assume that all states are measurable and available. Furthermore, the desired system tracking signal y_c and its derivative are also bounded. If they are not bounded, the tracking control goal of the suspension cable system of an unmanned helicopter with sensor fault cannot be realized. From above analysis, we can conclude that Assumption 2 is also reasonable.

3. Design of Sensor Fault Estimator

In this section, the design of sensor fault estimator will be given for the suspension cable system of the unmanned helicopter. Considering (4) and (5), we have

$$\begin{aligned} \dot{\chi} &= \dot{X}_2 + \dot{D}(t) \\ &= F(X) + G(X)u + \dot{D}(t). \end{aligned} \quad (6)$$

Since $F(X)$ includes $\dot{\theta}$, $F(X)$ can be written as $F(X) = F(X_1, \chi) + \Delta F(X_2)$. At the same time, $G(X) = G(X_1)$. Thus, we obtain

$$\dot{\chi} = F(X_1, \chi) + \Delta F(X_2) + G(X_1)u + \dot{D}(t). \quad (7)$$

Without loss of generality, we assume $\Delta F(X_2)$ and its derivative are bounded. Then, $\|\Delta F(X_2)\| \leq \tau_3$ with $\tau_3 > 0$. Because the uncertain term $\Delta F(X_2)$ is generated by the sensor fault, this assumption is reasonable. Under the sensor faults, the nonlinear line motion model (4) of the suspension cable system of an unmanned helicopter can be described as

$$\begin{cases} \dot{X}_1 = X_2, \\ \dot{\chi} = F(X_1, \chi) + G(X_1)u + \dot{D}(t) + \Delta F(X_2), \\ y = X_1. \end{cases} \quad (8)$$

Invoking (5), system (8) can be modified as

$$\begin{cases} \dot{X}_1 = \chi - D(t), \\ \dot{\chi} = F(X_1, \chi) + G(X_1)u + \dot{D}(t) + \Delta F(X_2), \\ y = X_1. \end{cases} \quad (9)$$

To estimate the unknown sensor fault of the suspension cable system of the unmanned helicopter, the sensor fault estimator can be designed as follows [41]:

$$\begin{aligned} \hat{D} &= z_1 + L_1(X_1), \\ \dot{z}_1 &= -P_1(X_1)(\chi - \hat{D}) + \hat{D}, \\ \hat{D} &= z_2 + L_2(\chi), \\ \dot{z}_2 &= -P_2(\chi)(F(X_1, \chi) + G(X_1)u + \hat{D}), \end{aligned} \quad (10)$$

where \hat{D} and $\dot{\hat{D}}$ are the corresponding estimations of the sensor fault D and the derivation of the sensor fault \dot{D} .

$L_1(X_1)$ and $L_2(\chi)$ are the parameters of the studied sensor fault estimator which needs to be designed. The parameters $P_1(X_1)$ and $P_2(\chi)$ should satisfy $P_1(X_1) = (\partial L_1(X_1)/\partial X_1)$ and $P_2(\chi) = (\partial L_2(\chi)/\partial \chi)$. z_1 and z_2 are the internal states of the fault estimator.

Define the estimation errors of the fault estimator as $\tilde{D} = \hat{D} - D$ and $\tilde{\dot{D}} = \dot{\hat{D}} - \dot{D}$. Then, considering (9) and (10), there yields

$$\begin{aligned} \dot{\tilde{D}} &= \dot{\hat{D}} - \dot{D} = \dot{\hat{D}} - \dot{z}_1 - P_1(X_1)\dot{X}_1 \\ &= -P_1(X_1)\tilde{D} + \tilde{\dot{D}}, \end{aligned} \quad (11)$$

$$\begin{aligned} \dot{\tilde{\dot{D}}} &= \dot{\hat{\dot{D}}} - \dot{\dot{D}} = \dot{\hat{\dot{D}}} - \dot{z}_2 - P_2(\chi)\dot{\chi} \\ &\quad - P_2(\chi)\tilde{D} + \tilde{\dot{\dot{D}}} - P_2(\chi)\Delta F(X_2). \end{aligned} \quad (12)$$

We define $e_f = [\tilde{D}^T, \tilde{\dot{D}}^T]^T$. Invoking (11) and (12), we obtain

$$\dot{e}_f = Q(\bar{x})e_f + H_1\tilde{\dot{D}} + H_2\Delta F(X_2), \quad (13)$$

where $\bar{x} = [X_1, \chi]$, $Q(\bar{x}) = \begin{bmatrix} -P_1(X_1) & I_2 \\ -P_2(\chi) & 0 \end{bmatrix}$, $H_1 = \begin{bmatrix} 0 \\ I_2 \end{bmatrix}$, $H_2 = \begin{bmatrix} 0 \\ -P_2(\chi) \end{bmatrix}$, and $P_2(\chi)$ is designed as bounded function.

To investigate the convergence of the estimation error of the designed sensor fault estimator, the Lyapunov function is given as follows:

$$V_f = \frac{1}{2}e_f^T A e_f, \quad (14)$$

where A is a designed positive definite matrix.

Invoking (13) and Assumption 1, the time derivative of V_f can be described as

$$\begin{aligned} \dot{V}_f &= e_f^T \dot{e}_f = e_f^T (0.5(Q^T(\bar{x})A + A^T Q(\bar{x}))e_f + H_1\tilde{\dot{D}} + H_2\Delta F(X_2)) \\ &\leq e_f^T (0.5(Q^T(\bar{x})A + A^T Q(\bar{x})) + 0.5\|H_1\|^2 I_4 + 0.5\|H_2\|^2 I_4) e_f + 0.5\tau_2^2 + 0.5\tau_3^2. \end{aligned} \quad (15)$$

From above process analysis, the following theorem can be obtained for the sensor fault estimator of suspension cable system of the unmanned helicopter.

Theorem 1. Consider the suspension cable system of the unmanned helicopter (4) with sensor faults satisfying Assumption 1, the nonlinear sensor estimator is designed as (10). If the designed function parameters $L_1(X_1)$ and $L_2(\chi)$ are chosen to render the following inequality valid:

$$Q^T(\bar{x})A + A^T Q(\bar{x}) + \|H_1\|^2 I_4 + \|H_2\|^2 I_4 < 0, \quad (16)$$

then the sensor fault estimation error is uniformly ultimately bounded.

In accordance with equations (15) and (16) and Lemma 1, we can make a conclusion that the sensor estimation error \tilde{D} is uniformly ultimately bounded.

4. Design of Robust Antiswing Control Plan Based on Sensor Fault Estimator

In this section, the fault-tolerant antiswing control law will be developed for the suspension cable system of the unmanned helicopter (4) with sensor faults based on the backstepping technique. The particular design steps are as follows.

Step 1: considering $e_1 = y - y_c$ and (9) yields

$$\dot{e}_1 = \dot{y} - \dot{y}_c = \dot{X}_1 - \dot{y}_c = \chi - D - \dot{y}_c. \quad (17)$$

To design the robust antiswing fault-tolerant control plan, we define

$$e_2 = \chi - \alpha_1, \quad (18)$$

where α_1 is a designed virtual control law. Substituting (18) into (17) yields

$$\dot{e}_1 = e_2 + \alpha_1 - D - \dot{y}_c. \quad (19)$$

The virtual control law can be proposed as

$$\alpha_1 = -K_1 e_1 + \dot{y}_c + \hat{D}, \quad (20)$$

where $K_1 = K_1^T > 0$ is a given parameter.

Substituting (20) into (19), we have

$$\dot{e}_1 = -K_1 e_1 + e_2 + \tilde{D}. \quad (21)$$

Choose the Lyapunov function in the form of

$$V_1 = 0.5e_1^T e_1. \quad (22)$$

Differentiating (22) and considering (21) yields

$$\begin{aligned} \dot{V}_1 &= e_1^T \dot{e}_1 = -e_1^T K_1 e_1 + e_1^T e_2 + e_1^T \tilde{D} \\ &\leq -e_1^T (K_1 - 0.5I) e_1 + e_1^T e_2 + 0.5\|\tilde{D}\|^2. \end{aligned} \quad (23)$$

Step 2: differentiating (18), we obtain

$$\dot{e}_2 = \dot{\chi} - \dot{\alpha}_1. \quad (24)$$

Considering (9), (24) can be described as

$$u = -G(X_1)^T (G(X_1)G(X_1)^T)^{-1} \left(K_2 e_2 + F(X_1, \chi) + e_1 + \hat{D}(t) - \dot{\alpha}_1 - \beta \text{Sign}(e_2) \right). \quad (30)$$

where $K_2 = K_2^T > 0$, $\beta > \tau_3$, are designed parameter and $\text{Sign}(e_2) = [\text{sign}(e_{21}), \text{sign}(e_{22})]^T$.

$$\dot{e}_2 = F(X_1, \chi) + \dot{D}(t) - \dot{\alpha}_1 + \Delta F(X_2)$$

$$-G(X_1)G(X_1)^T (G(X_1)G(X_1)^T)^{-1} \left(K_2 e_2 + F(X_1, \chi) + \hat{D}(t) - \dot{\alpha}_1 + e_1 - \beta \text{sign}(e_2) \right) = -K_2 e_2 + \tilde{D} + \dot{e}_1 - e_1 + \Delta F(X_2) - \beta \text{Sign}(e_2). \quad (31)$$

Considering (27), (31) can be written as

$$\dot{e}_2 = -K_2 e_2 + \tilde{D} - e_1 - \gamma^{-1} e_1 + \Delta_1(\dot{y}_c, \hat{D}, e_1) + \Delta F(X_2) - \beta \text{sign}(e_2). \quad (32)$$

Choose the Lyapunov function candidate as

$$V_2 = 0.5e_2^T e_2. \quad (33)$$

Differentiating (33) and considering (31) yields

$$\dot{e}_2 = F(X_1, \chi) + G(X_1)u + \dot{D}(t) + \Delta F(X_2) - \dot{\alpha}_1. \quad (25)$$

In this paper, the method of the dynamic surface control technique is introduced to acquire the derivatives of the virtual control law α_1 to handle the sensor fault D in the first step. Consider the following first-order filter $\bar{\alpha}_1$ as follows [2]:

$$\begin{aligned} \gamma \dot{\bar{\alpha}}_1 + \bar{\alpha}_1 &= \alpha_1, \\ \bar{\alpha}_1(0) &= \alpha_1(0), \end{aligned} \quad (26)$$

where $\gamma = \text{diag}\{r_1, r_2\} > 0$ is a time constant.

By defining $e_l = \bar{\alpha}_1 - \alpha_1$, we have

$$\dot{e}_l = \dot{\bar{\alpha}}_1 - \dot{\alpha}_1 = -\gamma^{-1} e_l + \Delta_1(\dot{y}_c, \hat{D}, e_1), \quad (27)$$

where $\Delta_1(\dot{y}_c, \hat{D}, e_1)$ is the sufficiently smooth vector in regard to $\Omega_1(\dot{y}_c, \hat{D}, e_1)$. It can be obtained that the smooth function $\Delta_1(\cdot)$ is bounded on set $\Omega_1(\cdot)$ with the maximum being Δ_{1m} [2].

In order to analyze the convergence of the first-order filter (24), the Lyapunov function candidate is given by

$$V_l = 0.5e_l^T e_l. \quad (28)$$

Differentiating (29) and invoking (27), we obtain

$$\begin{aligned} \dot{V}_l &= e_l^T \dot{e}_l = e_l^T (-\gamma^{-1} e_l + \Delta_1(\dot{y}_c, \hat{D}, e_1)) \\ &\leq -e_l^T (\gamma^{-1} - 0.5I) e_l + 0.5\Delta_{1m}^2. \end{aligned} \quad (29)$$

To handle $\dot{D}(t)$, the sensor fault estimator (10) is used. Using the outputs of the sensor fault estimator and the first-order filter, the controller law is designed as

Substituting (30) into (25), we obtain

$$\begin{aligned} \dot{V}_2 &= e_2^T \dot{e}_2 = -e_2^T K_2 e_2 - e_2^T e_1 + e_2^T \tilde{D} - e_2^T (-\gamma^{-1} e_l + \Delta_1(\dot{y}_c, \hat{D}, e_1)) \\ &\quad + e_2^T \Delta F(X_2) - \beta e_2^T \text{sign}(e_2). \end{aligned} \quad (34)$$

Considering $e_2^T \Delta F(X_2) - \beta e_2^T \text{Sign}(e_2) < 0$, we have

$$\dot{V}_2 \leq -e_2^T (K_2 - I_2 - 0.5\gamma^{-2} I_2) e_2 - e_2^T e_1 + 0.5\|\tilde{D}\|^2 + 0.5e_l^2 + 0.5\Delta_{1m}^2. \quad (35)$$

The following theorem is given to summarize the design of the robust antiswing control scheme studied for the suspension cable system of an unmanned helicopter with sensor fault.

Theorem 2. *For the studied suspension cable system of an unmanned helicopter with sensor fault, based on the nonlinear dynamic (4) satisfying Assumption 1 and Assumption 2, the nonlinear sensor fault estimator is designed as (10). By utilizing the designed sensor fault estimator, the robust antiswing control scheme is designed as (20), (26), and (30). Under the sensor fault estimator-based antiswing control of*

the suspension cable system of an unmanned helicopter, the closed-loop system is convergent and all signals are bounded.

Proof. In order to prove the stability of the whole system, the Lyapunov function is selected as

$$V = V_f + V_1 + V_2 = \frac{1}{2}e_f^T e_f + 0.5e_1^T e_1 + 0.5e_2^T e_2. \quad (36)$$

Differentiating (36) and considering (15), (23), and (35), we have

$$\begin{aligned} \dot{V} \leq & e_f^T \left(0.5(Q^T(\bar{x})A + A^T Q(\bar{x})) + 0.5\|H_1\|^2 I_4 + 0.5\|H_2\|^2 I_4 \right) e_f + 0.5\tau_2^2 \\ & - e_1^T (K_1 - 0.5I_2)e_1 + e_1^T e_2 + 0.5\|\tilde{D}\|^2 \\ & - e_2^T (K_2 - I_2 - 0.5\gamma^{-2}I_2)e_2 - e_2^T e_1 + 0.5\|\tilde{D}\|^2 + 0.5e_l^2 + 0.5\Delta_{1m}^2 \\ & - e_l^T (\gamma^{-1} - 0.5I_2)e_l + 0.5\Delta_{1m}^2 + 0.5\tau_2^2 + 0.5\tau_3^2. \end{aligned} \quad (37)$$

Using the definition $e_f = [\tilde{D}^T, \tilde{D}^T]^T$, we have

$$\begin{aligned} \dot{V} \leq & e_f^T \left(0.5(Q^T(\bar{x})A + A^T Q(\bar{x})) + 0.5\|H_1\|^2 I_4 + 0.5\|H_2\|^2 I_4 + I_4 \right) e_f - e_1^T (K_1 - 0.5I_2)e_1 \\ & - e_2^T (K_2 - I_2 - 0.5\gamma^{-2}I_2)e_2 \\ & - e_l^T (\gamma^{-1} - I_2)e_l + \tau_2^2 + \Delta_{1m}^2 + 0.5\tau_3^2. \end{aligned} \quad (38)$$

Define

$$\begin{aligned} \kappa = & \lambda_{\min} \left(\left(0.5(Q^T(\bar{x})A + A^T Q(\bar{x})) + 0.5\|H_1\|^2 I_4 + 0.5\|H_2\|^2 I_4 + I_4 \right) - (K_1 - 0.5I_2), \right. \\ & \left. (K_2 - I_2 - 0.5\gamma^{-2}I_2), (\gamma^{-1} - I_2) \right) \\ \rho = & \tau_2^2 + \Delta_{1m}^2 + 0.5\tau_3^2. \end{aligned} \quad (39)$$

Then, we have

$$\dot{V} \leq -\kappa V + \rho. \quad (40)$$

From (40) and Lemma 1, we can conclude that $e_1 \rightarrow 0$ when $t \rightarrow \infty$. Thus, the tracking control goal is realized for the suspension cable system of an unmanned helicopter with sensor fault. This concludes the proof. \square

5. Simulation Study

In the following, simulation results and analysis are given to illustrate the validity of the designed antiswing control of the suspension cable system of an unmanned helicopter based on the sensor fault estimator. The basic parameters of the

suspension cable system of an unmanned helicopter are referred to [2] and are presented in Table 1.

In the given simulation analysis, the system initial conditions are given by $[X_1^T(0), X_2^T(0)] = [0, 0, 0, 0]^T$. The desired trajectories of an unmanned helicopter are as follows:

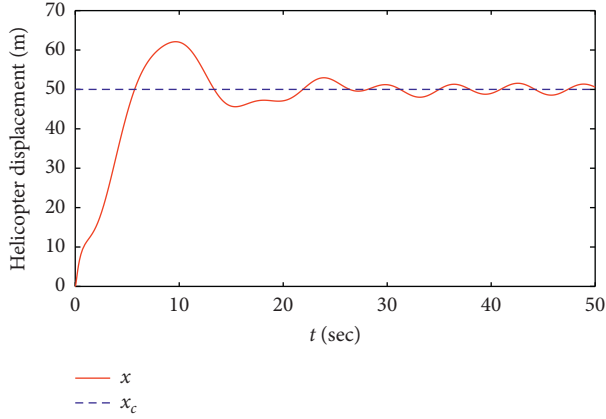
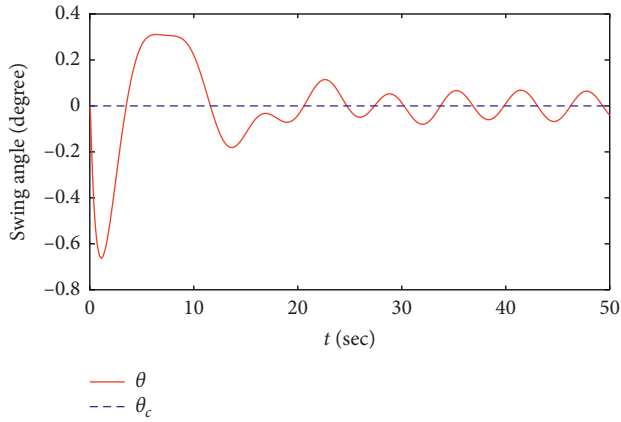
$$\begin{aligned} x_c &= 50\text{m}, \\ \theta_c &= 0^\circ. \end{aligned} \quad (41)$$

Furthermore, the sensor fault vector is given by

$$D = \begin{bmatrix} 0.15 \sin(t) \\ 0.1 \sin(t) \end{bmatrix}. \quad (42)$$

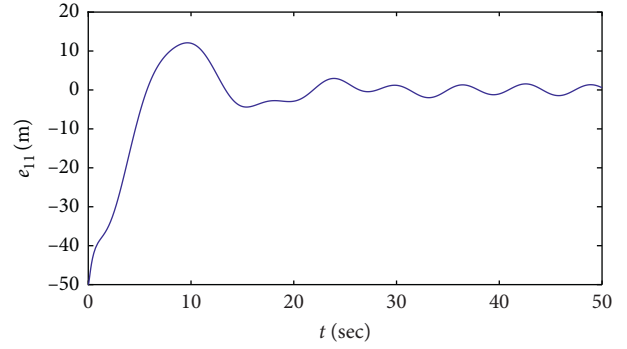
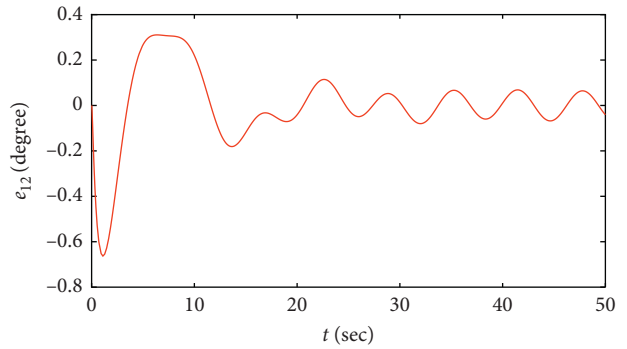
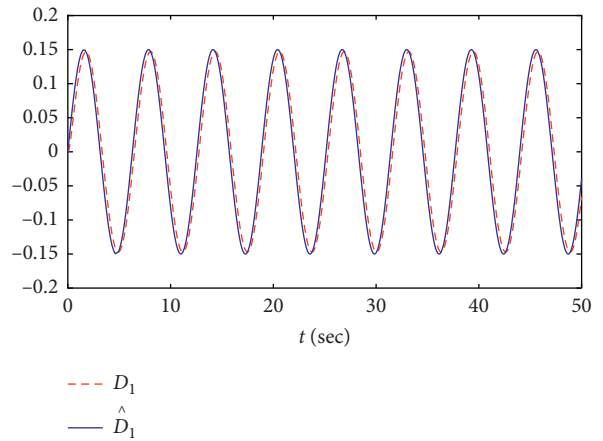
TABLE 1: System parameters.

Symbol	Unit	Value
M	kg	2000
m	kg	500
g	m/s ²	9.8
l	m	15

FIGURE 2: The tracking control result of the helicopter displacement x .FIGURE 3: The tracking control result of the swing angle of hanging load θ .

The corresponding design parameters of the developed antiswing control of the suspension cable system of an unmanned helicopter based on the sensor fault estimator are chosen as $K_1 = I_2$, $K_2 = 150I_2$, and $\gamma = 0.5$. The nonlinear sensor fault estimator is designed as (10). Based on the designed sensor fault estimator, the robust antiswing control scheme is developed as (20), (26), and (30). The simulation results are presented in Figures 2–7.

Figures 2 and 3 show the tracking control result of the suspension cable system of an unmanned helicopter with sensor fault by using the designed robust antiswing control plan which can verify the effectiveness of the developed antiswing control method. The tracking control errors are

FIGURE 4: The tracking control error of the helicopter displacement x .FIGURE 5: The tracking control error of the swing angle of hanging load θ .FIGURE 6: The estimation result of the sensor fault D_1 .

presented in Figures 4 and 5. They can be seen that the tracing errors of the suspension cable system of an unmanned helicopter with sensor fault are convergent and bounded. Then, Figures 6 and 7 show the effectiveness of the developed sensor fault estimator. We can note that the estimation results \hat{D} are good and meet the antiswing performance requirement by using the developed sensor fault estimator. Specially, we can see that the swing angle of the load is small in the flight process. Thus, the antiswing performance is achieved under the sensor fault.

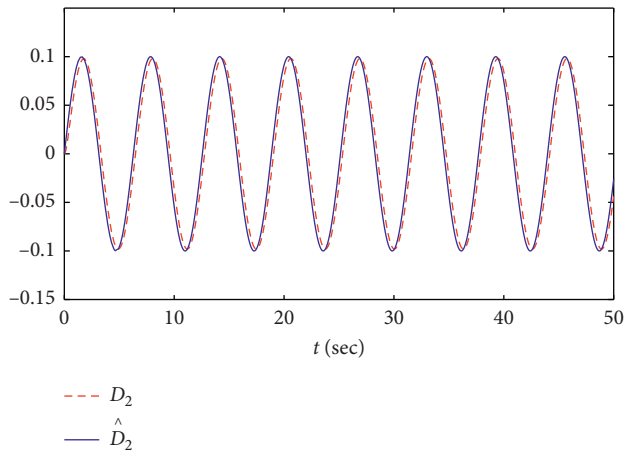


FIGURE 7: The estimation result of the sensor fault D_2 .

On the basis of the given simulation results and analysis, we can draw a conclusion that the satisfactory antiswing control performance can be guaranteed for the suspension cable system of an unmanned helicopter based on the sensor fault estimator. Thus, the designed antiswing control scheme is valid for the suspension cable system of an unmanned helicopter.

6. Conclusion

The robust antiswing control scheme has been proposed for the suspension cable system of an unmanned helicopter with sensor fault. In order to tackle the sensor fault, a sensor fault estimator has been designed to estimate it. By using the output of sensor fault estimator, the robust tolerant control scheme has been developed to maintain the desired tracking control performance during the flight progress. In accordance with the designed antiswing controller, the system can track the desired trajectory and the whole system stability is guaranteed by using the Lyapunov method. Simulation results have been presented to show the efficiency of the studied antiswing control law for the suspension cable system of an unmanned helicopter with sensor fault. In the future work, the finite time sensor fault estimator can be designed for the suspension cable system of an unmanned helicopter.

Data Availability

The data used to support the findings of this study are available upon request to the corresponding author.

Conflicts of Interest

The author declares that there are no conflicts of interest.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (Grant no. 61803207) and Qinglan Project in Jiangsu Province.

References

- [1] K. Dalamagkidis, K. P. Valavanis, and L. A. Piegl, "Nonlinear model predictive control with neural network optimization for autonomous autorotation of small unmanned helicopters," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 4, pp. 818–831, 2010.
- [2] K. Yan, M. Chen, Q. Wu, and K. Lu, "Robust attitude fault-tolerant control for unmanned autonomous helicopter with flapping dynamics and actuator faults," *Transactions of the Institute of Measurement and Control*, vol. 41, no. 5, pp. 1266–1277, 2019.
- [3] H. N. Wang, *Decreasing Oscillations of Aerial Cranes*, Beijing Institute of Technology, Beijing, China, 2015.
- [4] B. Nagabhushan, "Low-speed stability characteristics of a helicopter with a sling load," *Vertica*, vol. 9, no. 4, pp. 345–361, 1985.
- [5] D. Fusato, G. Guglieri, and R. Celi, "Flight dynamics of an articulated rotor helicopter with an external slung load," *Journal of the American Helicopter Society*, vol. 46, no. 1, pp. 3–13, 2001.
- [6] M. Chen, Y. Ren, and J. Y. Liu, "Anti-disturbance control for a suspension cable system of helicopter subject to input nonlinearities," *IEEE Systems, Man, and Cybernetics: Systems*, vol. 48, no. 12, pp. 2292–2304, 2017.
- [7] F. Omar, F. Karray, O. Basir et al., "Autonomous overhead crane system a fuzzy logic controller," *Journal of Vibration and Control*, vol. 10, no. 6, pp. 1255–1270, 2004.
- [8] K. Yomchinda and K. Thanan, "Development of dynamic inversion based outer-loop anti-swing control law for a helicopter slung-load system," *AIAA Journal*, vol. 10, no. 6, pp. 205–209, 2015.
- [9] K. Kang, J. V. R. Prasad, and E. Johnson, "Active control of a UAV helicopter with a slung load for precision airborne cargo delivery," *Unmanned Systems*, vol. 04, no. 03, pp. 213–226, 2016.
- [10] S. El-Ferik, A. H. Syed, H. M. Omar, and M. A. Deriche, "Anti-swing nonlinear path tracking controller for helicopter slung load system," *Ifac Proceedings Volumes*, vol. 46, no. 30, pp. 134–141, 2013.
- [11] M. Bisgaard, A. la Cour-Harbo, and J. Dimon Bendtsen, "Adaptive control system for autonomous helicopter slung load operations," *Control Engineering Practice*, vol. 18, no. 7, pp. 800–811, 2010.
- [12] K. Thanapalan, "Nonlinear controller design for a helicopter with an external slung load system," *Systems Science & Control Engineering*, vol. 5, no. 1, pp. 97–107, 2017.
- [13] Z. Yang, S. H. Wang, B. Chang, and W. H. Wu, "Adaptive constrained backstepping controller with prescribed performance methodology for carrier-based UAV," *Aerospace Science and Technology*, vol. 5, no. 7, pp. 90–92, 2019.
- [14] G. Peng, C. Yang, W. He, and C. L. P. Chen, "Force sensorless admittance control with neural learning for robots with actuator saturation," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 4, pp. 3138–3148, 2020.
- [15] Z. Gao and D. W. C. Ho, "State/noise estimator for descriptor systems with application to sensor fault diagnosis," *IEEE Transactions on Signal Processing*, vol. 54, no. 4, pp. 1316–1326, 2006.
- [16] C. Yang, D. Huang, W. He, and L. Cheng, "Neural control of robot manipulators with trajectory tracking constraints and input saturation," *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–12, 2020.
- [17] M. Chen, P. Shi, and C. C. Lim, "Adaptive neural fault-tolerant control of a 3-DOF model helicopter system," *IEEE*

- Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 2, pp. 260–270, 2015.
- [18] G. Tao, S. Chen, and S. M. Joshi, “An adaptive control scheme for systems with unknown actuator failures,” *Automatica*, vol. 38, no. 6, pp. 1027–1034, 2002.
- [19] X. Wei, M. Verhaegen, and T. V. Engelen, “Sensor fault detection and isolation for wind turbines based on subspace identification and Kalman filter techniques,” *International Journal of Adaptive Control & Signal Processing*, vol. 24, no. 8, pp. 687–707, 2010.
- [20] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, “Robot learning system based on adaptive neural control and dynamic movement primitives,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 777–787, 2019.
- [21] M. Chen, S.-Y. Shao, and B. Jiang, “Adaptive neural control of uncertain nonlinear systems using disturbance observer,” *IEEE Transactions on Cybernetics*, vol. 47, no. 10, pp. 3110–3123, 2017.
- [22] C. Yang, X. Wang, L. Cheng, and H. Ma, “Neural-Learning-based telerobot control with guaranteed performance,” *IEEE Transactions on Cybernetics*, vol. 47, no. 10, pp. 3148–3159, 2017.
- [23] M. Chen, B. B. Ren, and Q. X. Wu, “Anti-disturbance control of hypersonic flight vehicles with input saturation using disturbance observer,” *Science China Information Science*, vol. 58, no. 1, pp. 1–13, 2015.
- [24] H. Lin, T. Zhang, Z. Chen, H. Song, and C. Yang, “Adaptive fuzzy Gaussian mixture models for shape approximation in robot grasping,” *International Journal of Fuzzy Systems*, vol. 21, no. 4, pp. 1026–1037, 2019.
- [25] H. Huang, C. Yang, and C. L. P. Chen, “Optimal robot-environment interaction under broad fuzzy neural adaptive control,” *IEEE Transactions on Cybernetics*, pp. 1–12, 2020.
- [26] M. Chen and J. Yu, “Adaptive dynamic surface control of NSVs with input saturation using a disturbance observer,” *Chinese Journal Of Aeronautics*, vol. 28, no. 3, pp. 853–864, 2015.
- [27] C. Yang, G. Peng, L. Cheng, J. Na, and Z. Li, “Force sensorless admittance control for teleoperation of uncertain robot manipulator using neural networks,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–11, 2019.
- [28] Y. Wu and J. Dong, “Local stabilization for discrete-time T-S fuzzy time-delay systems with sensor fault,” *Fuzzy Sets and Systems*, vol. 374, no. 11, pp. 115–137, 2019.
- [29] H. Wang and L. Yao, “Sensor fault diagnosis and fault-tolerant control for stochastic distribution time-delayed control systems,” *International Journal of Adaptive Control and Signal Processing*, vol. 33, no. 9, pp. 1395–1406, 2019.
- [30] L. Chen, Y. Zhao, S. Fu, M. Liu, and J. Qiu, “Fault estimation observer design for descriptor switched systems with actuator and sensor failures,” *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 66, no. 2, pp. 810–819, 2019.
- [31] J. Xia, B. Jiang, and K. Zhang, “Robust asymptotic estimation of sensor faults for continuous-time interconnected systems,” *International Journal of Control, Automation and Systems*, vol. 17, no. 12, pp. 3170–3178, 2019.
- [32] B. Mohamed, C. Ahmed, B. Moussa et al., ““Proprioceptive sensors’ fault tolerant control strategy was designed for an autonomous vehicle,” *Sensors*, vol. 18, no. 6, p. 1893, 2018.
- [33] T. H. Lee, C. P. Lim, S. Nahavandi et al., “Observer-based \mathcal{H}_∞ fault-tolerant control for linear systems with sensor and actuator faults,” *IEEE Systems Journal*, vol. 13, no. 2, pp. 1981–1990, 2019.
- [34] M. Taimoor and L. Aijun, “Lyapunov theory based adaptive neural observers design for aircraft sensors fault detection and isolation,” *Journal of Intelligent & Robotic Systems*, vol. 98, no. 3, pp. 311–323, 2019.
- [35] S. S. Nasrolahi and F. Abdollahi, “Sensor fault detection and recovery in satellite attitude control,” *Acta Astronautica*, vol. 145, no. 4, pp. 275–283, 2018.
- [36] F. Chen, J. Niu, and G. Jiang, “Nonlinear fault-tolerant control for hypersonic flight vehicle with multi-sensor faults,” *IEEE Access*, vol. 6, p. 25427, Article ID 25436, 2018.
- [37] Z. Liu, D. Theilliol, L. Yang, Y. He, and J. Han, “Observer-based linear parameter varying control design with unmeasurable varying parameters under sensor faults for quad-tilt rotor unmanned aerial vehicle,” *Aerospace Science and Technology*, vol. 92, no. 9, pp. 696–713, 2019.
- [38] P. Lu, E.-J. van Kampen, C. de Visser, and Q. Chu, “Nonlinear aircraft sensor fault reconstruction in the presence of disturbances validated by real flight data,” *Control Engineering Practice*, vol. 49, pp. 112–128, 2016.
- [39] J. C. Wang and X. H. Qi, “Sensor fault estimation method for flight control systems based on aerodynamic parameter identification,” *Binggong Xuebao/Acta Armamentarii*, vol. 36, no. 1, pp. 103–110, 2015.
- [40] C. Hu, L. Cao, X. Zhou, B. Sun, and N. Wang, “Fuzzy adaptive nonlinear sensor-fault tolerant control for a quadrotor unmanned aerial vehicle,” *Asian Journal of Control*, vol. 22, no. 3, pp. 1163–1176, 2020.
- [41] M. Chen, “Robust tracking control for self-balancing mobile robots using disturbance observer,” *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 3, pp. 458–465, 2017.

Research Article

Multiscale Receptive Fields Graph Attention Network for Point Cloud Classification

Xi-An Li ^{1,2}, Li-Yan Wang ¹ and Jian Lu ³

¹Department of Stomatology, Foshan Woman and Children's Hospital, Foshan, Guangdong 528000, China

²School of Mathematical Sciences, Shanghai Jiao Tong University, Shanghai 200240, China

³Guangdong Institute of Aeronautics and Astronautics Equipment and Technology, Zhuhai 519000, China

Correspondence should be addressed to Xi-An Li; lixa0415@sjtu.edu.cn, Li-Yan Wang; wangliyankmmc@163.com, and Jian Lu; luj@giaaet.com

Received 12 September 2020; Revised 19 November 2020; Accepted 22 January 2021; Published 24 February 2021

Academic Editor: Rongxin Cui

Copyright © 2021 Xi-An Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Understanding the implication of point cloud is still challenging in the aim of classification or segmentation for point cloud due to its irregular and sparse structure. As we have known, PointNet architecture as a ground-breaking work for point cloud process can learn shape features directly on unordered 3D point cloud and has achieved favorable performance, such as 86% mean accuracy and 89.2% overall accuracy for classification task, respectively. However, this model fails to consider the fine-grained semantic information of local structure for point cloud. Then, a multiscale receptive fields graph attention network (named after MRFGAT) by means of semantic features of local patch for point cloud is proposed in this paper, and the learned feature map for our network can well capture the abundant features information of point cloud. The proposed MRFGAT architecture is tested on ModelNet datasets, and results show it achieves state-of-the-art performance in shape classification tasks, such as it outperforms GAPNet (Chen et al.) model by 0.1% in terms of OA and compete with DGCNN (Wang et al.) model in terms of MA.

1. Introduction

Point cloud as a simple and efficient representation for 3D shapes and scenes has become more and more popular in the fields of both academia and industry. For example, autonomous vehicle [1–4], robotic mapping and navigation [5–7], 3D shape representation and modelling [8, 9], and other relevant applications [10–15]. Lots of ways can be used to obtain 3D point cloud data, such as utilizing 3D scanners including physical touch or noncontact measurements with light, sound, LiDAR, etc.

Up to now, a variety of approaches have been developed to handle this kind of data, such as the commonly used traditional handcraft algorithms [16–18]. In terms of these methods, it is significant to classify or segment point cloud by choosing salient features of point cloud, such as normals, curvatures, and colors. Handcrafted features are usually employed to address specific problems but tough to transfer

to new tasks. Then, it is a hot topic, in last decades, that how to overcome the shortcomings for traditional methods.

With the development of deep learning, some existed end-to-end neural networks have overcome many challenges' stem from 3D data and made great breakthrough for point cloud, see Figure 1. In particular, the modificatory works of convolutional neural networks (CNNs) have achieved significant success for point cloud data in computer vision tasks, such as PointNet [19] and its improved version [20], PointCNN [21, 22], and PointSift [23]. Unfortunately, lots of neural networks for point cloud only capture global feature without local information which are also an import semantic feature for point cloud. Hence, exploiting reasonably the local information of point cloud has become a new research hotspot, and some valuable works also have sprung up recently. PointNet++ [20] extends the PointNet model by constructing a hierarchical neural network that recursively applies PointNet with designed sampling and

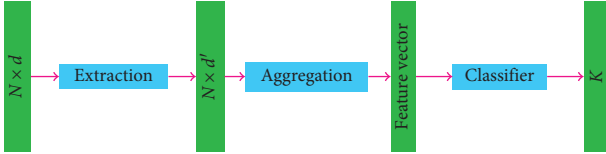


FIGURE 1: A typical framework of a 3D point cloud classification process. The model takes N points in d -dimensional space as its input and then extracts point features of dimension d' , followed by an aggregation module used to form a feature vector which is invariant to point permutation. Finally, a classifier is used to classify the resulting feature vector into K categories.

grouping layers to extract local features. Graph neural networks [24, 25] can not only directly address a more general class of graphs, e.g., cyclic, directed, and undirected graphs, but also be applied to deal with point cloud data. Recently, DGCNN [26] and its variant [27] well utilized the graph network with respect to the edges' convolution on points and then obtained the local edges' information of point cloud. Other relevant works applying the graph structure of point cloud can be found in [28–30].

Attention mechanism plays a significant role in machine translation task [31], vision-based task [32], and graph-based task [33]. Combining graph structure and attention mechanism, some favorable network architectures are constructed which leverage well the local semantic features of point cloud. Readers can refer to [34–36].

However, the scale of different graphs for the existed graph networks are fixed; then, the semantic expression of the point will not be good. Hence, in this work, inspired by graph attention network [33], graph convolution network [37], and local contextual information networks, we design a multiscale receptive fields' graph attention network for point cloud classification. Unlike previous models that only consider the attribute information such as coordinate of each single point or only exploit local semantic information of point, we pay attention to the spatial context information of both local and global structure for point cloud. Finally, like the standard convolution in grid domain, our model can also be efficiently implemented for the graph representation of a point cloud.

The key contributions of our work are summarized as follows:

- (i) We construct graph of local patch for point cloud and then enhance the feature representation of point in point cloud by combining edges' information and neighbors' information
- (ii) We introduce a multiscale receptive fields' mechanism to capture the local semantic features in various ranges for point cloud
- (iii) We balance the influence between neighbors and centroid in the local graph by means of attention mechanism
- (iv) We release our code to facilitate reproducibility and future research (<https://github.com/Blue-Giant/MRFGAT-NET>)

The rest parts of this paper are structured as follows. In Section 2, we review the most closely related literatures on point

cloud. In Section 3, we introduce our proposed MRFGAT architecture and provide the details of our framework in terms of shape classification for point cloud. We describe the dataset and design comparison algorithms in Section 4, followed by the experiments' results and discussion. Finally, some concluding remarks are made in Section 5.

2. Related Works

2.1. Pointwise MLP and Point Convolution Networks. Utilizing the deep learning technique, the classical PointNet [19] was proposed to deal with directly unordered point clouds without using any volumetric or grid-mesh representation. The main idea of this network is as follows. At first, a Spatial Transformer Network (STN) module similar to feature-extracting process is constructed which guarantees the invariance of transformations. Then, a shared pointwise Multilayer-Perceptron (MLP) module is introduced which is used to extract semantic features from point sets. At last, the final semantic information of point cloud is aggregated by means of a max pooling layer. Due to the favorable ability to approximate any continuous function for MLP which is easy to implement by point convolution, some related works were presented according to the PointNet architecture [38, 39].

Similar to convolution operator in 2D space, some convolution kernels for points in 3D space are designed which can capture the abundant information of point cloud. PointCNN [21] used a local \mathcal{X} -transformation kernel to fulfill the invariance of permutation for points and then generalized this technique to the hierarchical form in analogy to that of image CNNs. The authors in [40–42] extended the convolution operator of 2D space, applied at individual point in local region of point cloud, and then collected the neighbors' information in the hierarchical convolution layer to the center point. Kernel Point Convolution (KPConv) [43] consists of a set of local 3D filters and overcomes stand point convolution limitation. This novel kernel structure is very flexible to learn local geometric patterns without any weights.

2.2. Learning Local Features. In order to overcome the shortcoming for PointNet-like networks which fail to exploit local features, some hierarchical architectures have been developed, for example, PointNet [20] and So-Net [38], to aggregate local information with MLP operation by considering local spatial relationships of 3D data. In contrast to the previous type, these methods can avoid sparsity and update dynamically in different feature dimensions. According to a Capsule Networks, 3D Capsule Convolutional Networks were developed which can learn well the local features of point cloud, see [44–46].

2.3. Graph Convolutional Networks. Graph Convolutional Neural Networks (GCNNs) have gained more and more attraction to address irregularly structured data, such as citation networks and social networks. In terms of 3D point cloud data, GCNNs have shown its powerful ability on classification and segmentation tasks. Using the convolution

operator with respect to the graph in the spectral domain is an important approach [47–49], but it needs to calculate a lot of parameters on polynomial or rational spectral filters [50]. Recently, many researchers constructed local graph of point cloud by utilizing each point’s neighbors in low-dimensional manifold based on N -dimensional Euclidean distance and then grouped each point’s neighbors in the form of high-dimensional vectors, such as EdgeConv-like works [26, 27, 51] and graph convolutions [37, 52]. Compared with the spectral methods, its main merit is that it is more consistent with the characteristics of data distribution. Specially, EdgeConv extracts edge features through the relationship between central point and neighbor points by successively constructing graph in the hierarchical model. To sum up, the graph convolution network combines features on local surface patches which are invariant to the deformations of patches for point cloud in Euclidean space.

2.4. Attention Mechanism. The idea of attention has been successfully used in natural language processing (NLP) [31] and graph-based work [33, 53]. Attention module can balance the weight relationship of different nodes in graph structure data or different parts in sequence data.

Recently, the attention idea has obtained more and more attraction and made a great contribution to point cloud processes [34, 35]. In these works, it is significant to aggregate point or edge features by means of attention module. Unlike the existing methods, we try to enhance the high-level representation of point cloud by capturing the relation of points and local information along its channel.

3. Our Approach

The framework of point cloud classification includes two contents: taking the 3D point cloud as input and assigning one semantic class label for each point. Based on the technique of extracting features from the local directed graph and attention mechanism, a new architecture for shape classification task is proposed to better learn point’s representation for unstructured point cloud. This new architecture is composed of three components which are the point enhancement, the feature representation, and the prediction. These three components are fully coupled together, which leads to an end-to-end training pipeline.

3.1. Problem Statement. At first, we let $P = \{p_i \in \mathbb{R}^d, i = 1, 2, \dots, N\}$ represent a raw set of unordered points as the input for our mode, where N is the number of the points and p_i is a feature vector with a dimension d . In actual applications, the feature vector p_i might contain 3D space coordinates (x, y, z) , color, intensity, surface normal, etc. For the sake of simplicity, we set $d = 3$ in our work and only take 3D coordinates of point as the feature representation for point. A classification or semantic segmentation of a point cloud are map Φ_c or Φ_s , respectively, which assign individual point semantic labels or point cloud semantic labels, respectively, i.e.,

$$\Phi: P \longrightarrow L^k. \quad (1)$$

Here, Φ represents the map Φ_c or Φ_s . The objective of our model is finding the optimal map that can obtain accurate semantic labels.

The above map should satisfy some constraints including the following. (1) Permutation invariance: the order of points may vary but does not influence the category of the point or point cloud. (2) Transformation invariance: for the uncertain translation and rotation of point cloud, the results of classification or segmentation should not be changed for point or point cloud.

3.2. Graph Generation for Point Cloud. Some works indicate that local features of point cloud can be used to improve the discriminability of point; then, exploring the relationship among points in a whole sets or local patch is a keypoint for our work. Graph neural network is a feasible approach to process point cloud because it propagates on each node for the whole sets or a local patch of point cloud individually, ignores the permutation order of nodes, and then extracts the local information between nodes. To apply the graph neural network on the point cloud, we firstly convert it to a directed graph. Like DGCNN [26, 27] and GAPNet [34], we can obtain the neighbors (including self) of each point in point cloud by means of K -NN algorithm and then construct a local directed graph in Euclidean space. Figure 2 depicts the directed graph $G = (V; E)$ of local patch for point cloud, $V = \{1, 2, \dots, K\}$ is the vertice set of G , namely, the nodes of local patch, E stands for the edge set of G , and each edge is $e_{ij} = p_i - p_{ij}$ with $p_i \in P$ and $p_{ij} \in V$ being centroid and neighbors, respectively.

3.3. Single Receptive Field Graph Attention Layer (SRFGAT). In order to aggregate the information of neighbors, we use a neighboring-attention mechanism which is introduced to obtain attention coefficients of neighbors for each point, see Figure 3. Additionally, edge features are important local features which can enhance the semantic expression of point; then, an edge-attention mechanism is also introduced to aggregate information of different edges, see Figure 3. In light of the attention mechanism [33, 34], we firstly transform the neighbors and edges into a high-level feature space to obtain sufficient expressive power. To this end, as an initial step, a parametric nonlinear function $h(\cdot)$ is applied to every neighbor and edge, and the results are defined by

$$\begin{aligned} e'_{ij} &= h(e_{ij}, \theta) \in \mathbb{R}^{F'}, \\ p'_{ij} &= h(p_{ij}, \theta) \in \mathbb{R}^{F'}, \end{aligned} \quad (2)$$

respectively, where θ is a set of learnable parameters of the filter and F' is output dimension. In our method, function $h(\cdot)$ is set to a single-layer neural network.

It is worthwhile to noting that edges in Euclidean space not only stand for the local features but also indicate the dependency between centroid and neighbor. We then obtain attentional coefficients of edges and neighbors which are

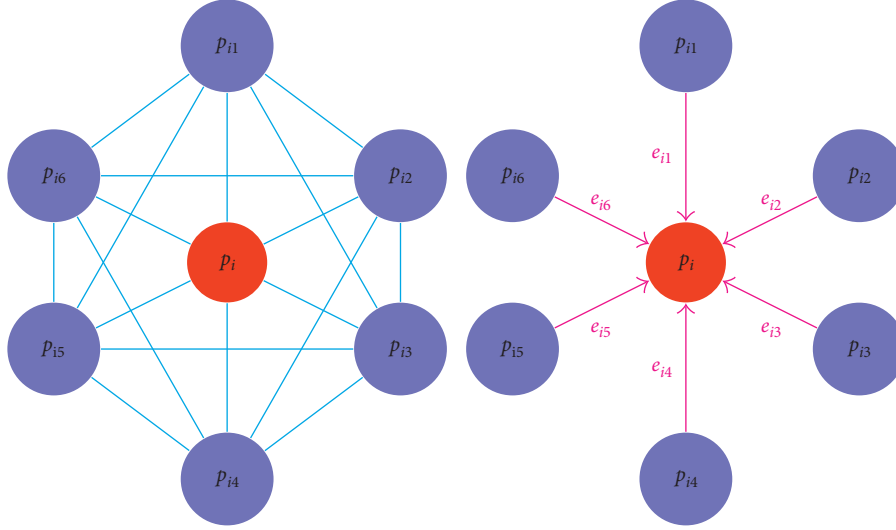


FIGURE 2: Local graph of point cloud. p_i and $\{p_{11}, p_{12}, p_{13}, p_{14}, p_{15}, p_{16}\}$ are a central point and its neighbors, respectively. The directed edges from the neighbors to the central point are denoted by $\{e_{11}, e_{12}, e_{13}, e_{14}, e_{15}, e_{16}\}$.

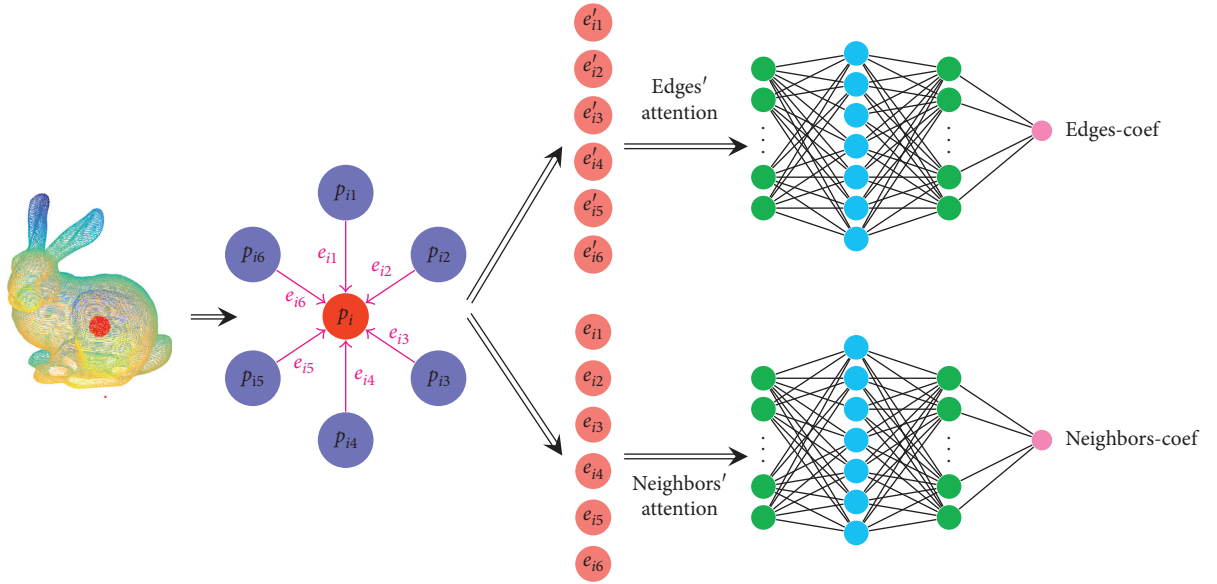


FIGURE 3: An illustration of attention coefficients' generation. The edge feature not only serves as local information to center point but also responds the effect of between neighbors and centroid. Edge attention and neighbor attention reflect the important of edge features and neighbor features to centroid, respectively.

$$a_{ij} = \text{Leakly ReLU}(g(e'_{ij}, \theta)) \text{ and } b_{ij} = \text{Leakly ReLU}(g(e_{ij}, \theta)), \quad (3)$$

respectively, where $g(e'_{ij}, \theta)$ and $g(e_{ij}, \theta)$ are single-layer neural network with 1-dimensional output. Leakly ReLU(\cdot) denotes nonlinear activation function leaky ReLU with $\text{ReLU} = \max\{0, x\}$. To make coefficients easily comparable across different neighbors and edges, we use a softmax operation to normalize the above coefficients which are defined as

$$\alpha_{ij} = \frac{\exp(a_{ij})}{\sum_k \exp(a_{ik})}, \quad (4)$$

$$\beta_{ij} = \frac{\exp(b_{ij})}{\sum_k \exp(b_{ik})},$$

respectively; then, the normalized coefficients are used to compute contextual feature for every point, and it is

$$\tilde{x}_i = f\left(\left(\sum_j \alpha_{ij} e'_{ij}\right) \parallel \left(\sum_j \beta_{ij} d'_{ij}\right)\right), \quad (5)$$

where $f(\cdot)$ is a nonlinear activation function and \parallel is concatenation operation. In our model, we choose ReLU function as $f(\cdot)$.

3.4. Multiscale Receptive Fields Graph Attention Layer (MRFGAT). In order to obtain sufficient feature information and stabilize the network, the multiscale receptive field strategy analogous to multiheads mechanism is proposed, see Figure 4. Unlike previous works, the sizes of receptive fields in our model are different for various branches. Therefore, we concatenate M independent SRFGAT module and generate a semantic feature with $M \times F^l$ channels:

$$\tilde{x}_i = \parallel_{m=1}^M \tilde{x}_i^{(m)}, \quad (6)$$

where $\tilde{x}_i^{(m)}$ is the receptive field feature of the m th branch, M is the total number of branches, and \parallel is the concatenation operation over feature channels.

3.5. MRFGAT Architecture. Our MRFGAT model shown in Figure 5 considers shape classification task for point cloud. The architecture is similar to PointNet [19]. However, there are three main differences between the architectures of MRFGAT and PointNet. Firstly, according to the analyses of LinkDGCNN model, we remove the transformation network which is used in many architectures such as PointNet, DGCNN, and GAPNet. Secondly, instead of only processing individual points of point cloud, we also exploit local features by a SRFGAT-layer before the stacked MLP layers. Thirdly, an attention pooling layer is used to obtain local feature information that is connected to the intermediate layer for forming a global descriptor. In addition, we aggregate individually the original edge feature of every SRFGAT channel and then obtain local features which can enhance the semantic feature of MRFGAT.

4. Experiments

In this section, we evaluate our MRFGAT model on 3D point cloud analysis for the classification tasks. To demonstrate effectiveness of our model, we then compare the performance for our model to recent state-of-the-art methods and perform ablation study to investigate different design variations.

4.1. Classification

4.1.1. Dataset. We demonstrate the feasibility and effectiveness of our model on the ModelNet dataset such as ModelNet40 benchmarks [54] for shape classification. The ModelNet40 dataset contains 12,311 meshed CAD models

that are classified to 40 man-made categories. In this work, we divide the ModelNet40 dataset into two parts: the part one is named as training set which includes 9843 models and the part two is called as testing set includes 2468 models. Then, we normalize the models in the unit sphere and uniformly sample 1,024 points over model surface. Besides, we further augment the training dataset by randomly rotating, scaling the point cloud, and jittering the location of every point by means of Gaussian noise with zero mean and 0.01 standard deviation for all the models.

4.1.2. Implementation Details. According to the analysis of the LinkDGCNN model [27], we omit the spatial transformation network to align the point cloud to a canonical space. The network employs four SRFGAP layer modules with (8, 16, 16, 24) channels to capture attention features, respectively. Then, four shared MLP layers with sizes (128, 64, 64, 64), respectively, followed by it are used to aggregate the feature information. Next, the output features are fed into an aggregation operation followed by the MLP layer with 1024 neurons. In the end of network, a max pooling operation and two full-connected layers (512, 256) are used to finally obtain the classification score. The training is carried out using Adam optimizer with minibatch training (batch size of 16) and an initial learning rate of 0.001. The ReLU activate function and Batch Normalization (BN) are also used in both the SRFGAP module and MLP layer. At last, the network was implemented using TensorFlow and executed on the server equipped with four NVIDIA GTX2080Ti.

4.1.3. Results. Figures 6–8 depict the process for training and testing. From the figures, we see that our model will quickly attain the stage of high accuracy, which means our model is highly efficient. Table 1 lists the results of our method and several recent state-of-the-art works. The methods listed in Table 1 have one thing in common. The input is only raw point cloud with 3D coordinates (x_i, y_i, z_i) . Based on these results, we can conclude that our model performs better than other methods and obtains wonderful performance on both the ModelNet40 benchmark. Compared to other point-based methods, the performance for our model is only a little weaker than that of DGCNN in terms of MA on ModelNet 40. However, it outperforms the previous state-of-the-art model GAPNet by 0.1% accuracy in terms of OA. These phenomena show that the strategy employing local and global features in different receptive fields is efficient, and it will help us to capture the prominent semantic feature for point cloud. And, in our model, since we introduce the structure of the data by providing the local interconnection between points and explore graph features from different scale field levels by the localized graph convolutional layers, it guarantees the exploration of more distinctive latent representations for each object class.

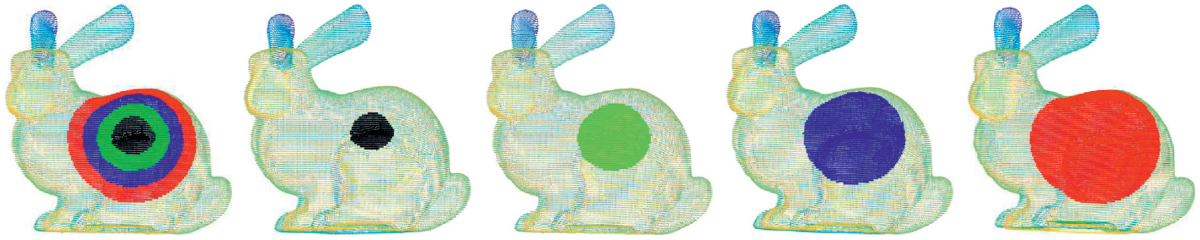


FIGURE 4: Multiscale receptive fields.

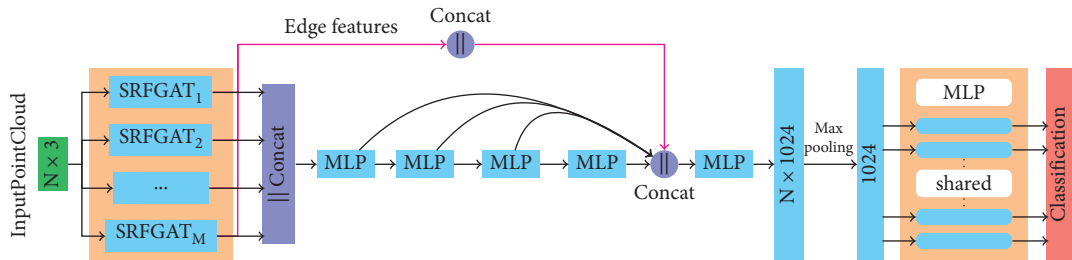


FIGURE 5: The architecture of classification. In this framework, it takes N points as input and applies M individual SRFGAT modules to obtain multiattention features on multilocal graphs; then, the output features are recast by means of five shared MLP layers and attention pooling layer, respectively. Finally, a shared full-connected layer is employed to form a global feature and then classification scores for c categories are obtained.

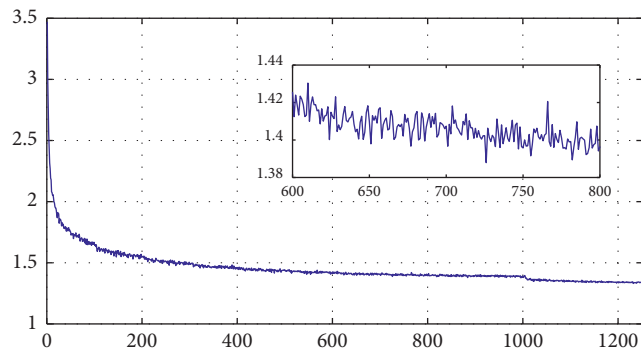


FIGURE 6: The curve of loss varying with epochs for ModelNet40, and it includes five training in one epoch.

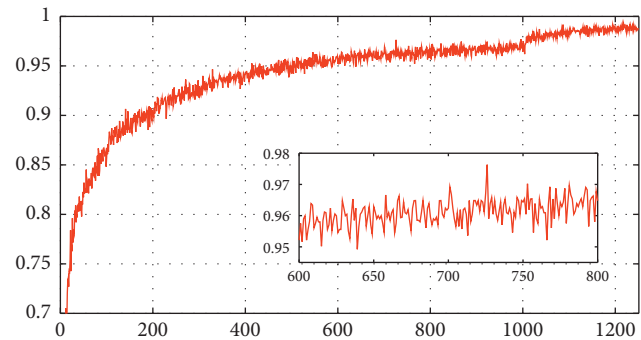


FIGURE 7: The curve of training accuracy varying with epochs for ModelNet40, and it includes five training in one epoch.

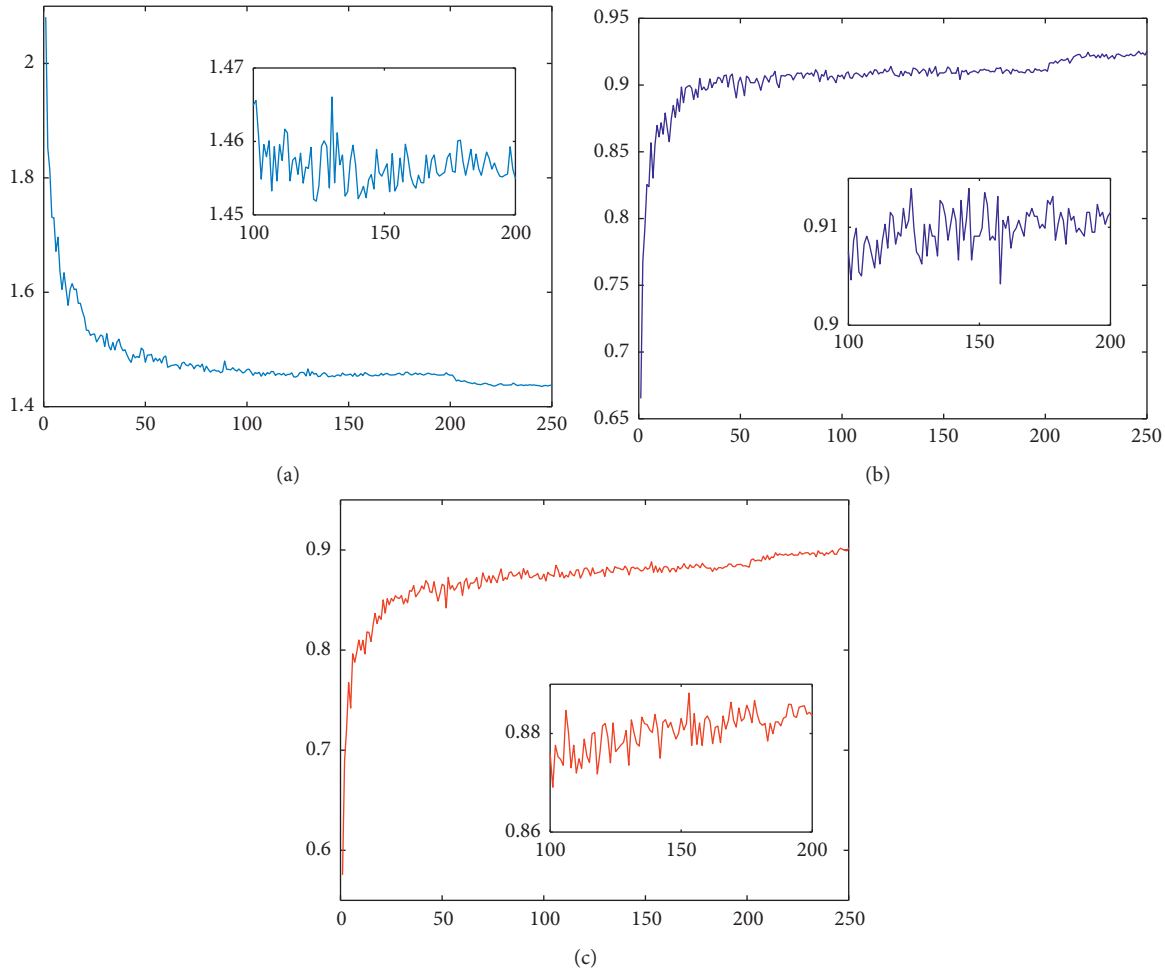


FIGURE 8: The curves of testing loss and accuracy varying with epochs for ModelNet40. (a) Loss. (b) Accuracy. (c) Mean accuracy.

TABLE 1: Classification results on ModelNet40. MA represents mean per-class accuracy, and the per-class accuracy is the ratio of the number of correct classifications to that of the objects in a class. OA denotes the overall accuracy, which is the ratio of the number of overall correct classifications to that of overall objects.

Method	Points	MA (%)	OA (%)
SO-Net [38]	2048×3	87.3	90.9
KD-Net [40]	1024×3	88.5	91.8
PointNet [19]	1024×3	86.0	89.2
PointNet++ [20]	1024×3	—	90.7
PointCNN [21]	1024×3	88.1	92.2
DGCNN [26]	1024×3	90.2	92.2
PCNN [22]	1024×3	—	92.3
GAPNet [34]	1024×3	89.7	92.4
Ours	1024×3	90.1	92.5

5. Conclusion

Enlightening by graph convolutional networks for the task of classification in 3D computer vision, we design a novel MRFGAT-based modules for point feature and context aggregation. Utilizing different receptive fields and attention strategies, the pipeline MRFGAT can capture

more fine features of point clouds for classification task. In addition, we list some comparable results with recent works which show that our model can achieve the state-of-the-art performance on the dataset ModelNet for classification task of point clouds; it outperforms the GAPNet model by 0.1 % in terms of OA and competes with the DGCNN model in terms of MA. It is necessary to point out that our model will have some burden for constructing varying scale graphs. Based on the state-of-the-art Graph Convolution Networks (GCN) for semantic segmentation in point cloud, it would be interesting to introduce our model to address this problem for unstructured data in the future.

Data Availability

This dataset used in this manuscript is available at https://shapenet.cs.stanford.edu/media/modelnet40_ply_hdf5_2048.zip.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by GDAS' Project of Science and Technology Development (no. 2018GDASCX-0804) and Project of Guangdong Engineering Technology Research Center (no. 810115228131).

References

- [1] C. R. Qi, W. Liu, C. Wu, H. Su, and L. J. Guibas, "Frustum PointNets for 3D object detection from RGB-D data," in *Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018*, pp. 918–927, Salt Lake City, UT, USA, June 2018.
- [2] J. Ku, M. Mozifian, J. Lee, A. Harakeh, and S. L. Waslander, "Joint 3D proposal generation and object detection from view aggregation," in *Proceedings of the 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1–8, Madrid, Spain, October 2018.
- [3] M. Liang, B. Yang, S. Wang, and R. Urtasun, "Deep continuous fusion for multi-sensor 3D object detection," in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 663–678, Munich, Germany, September 2018.
- [4] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, "Robot learning system based on adaptive neural control and dynamic movement primitives," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 777–787, 2019.
- [5] J. Biswas and M. Veloso, "Depth camera based indoor mobile robot localization and navigation," in *Proceedings of the 2012 IEEE International Conference on Robotics and Automation*, pp. 1697–1702, St Paul, MIN, USA, May 2012.
- [6] Y. Zhu, R. Mottaghi, E. Kolve et al., "Target-driven visual navigation in indoor scenes using deep reinforcement learning," 2017, <http://arxiv.org/abs/1609.05143>.
- [7] H. Wang, S. Wang, J. Yao, R. Pan, and J. Yang, Effective Anti-collision Algorithms for RFID Robots System, Assembly Automation Ahead-Of-Print (2019).
- [8] A. Golovinskiy, V. G. Kim, and T. Funkhouser, "Shape-based recognition of 3D point clouds in urban environments," in *Proceedings of the 2009 IEEE 12th International Conference on Computer Vision*, Kyoto, Japan, October 2009.
- [9] P. Gu, F. Zhou, D. Yu, F. Wan, W. Wang, and B. Yu, "A 3D reconstruction method using multisensor fusion in large-scale indoor scenes," *Complexity*, vol. 2020, Article ID 6973790, 14 pages, 2020.
- [10] H. Qiao, Y. Li, T. Tang, and P. Wang, "Introducing memory and association mechanism into a biologically inspired visual model," *IEEE Transactions on Cybernetics*, vol. 44, pp. 1485–1496, 2014.
- [11] H. Qiao, M. Wang, J. Su, S. Jia, and R. Li, "The concept of "attractive region in environment" and its application in high-precision tasks with low-precision systems," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 5, pp. 2311–2327, 2015.
- [12] C. Yang, H. Wu, Z. Li, W. He, N. Wang, and C.-Y. Su, "Mind control of A robotic arm with visual fusion Technology," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 3822–3830, 2018.
- [13] C. Yang, C. Zeng, C. Fang, W. He, and Z. Li, "A DMPs-based framework for robot learning and generalization of humanlike variable impedance skills," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 3, pp. 1193–1203, 2018.
- [14] Z. Zhao, C. K. Ahn, and H.-X. Li, "Boundary antidisturbance control of a spatially nonlinear flexible string system," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 6, pp. 4846–4856, 2020.
- [15] Z. Zhao, C. K. Ahn, and H.-X. Li, "Dead zone compensation and adaptive vibration control of uncertain spatial flexible riser systems," *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 3, pp. 1398–1408, 2020.
- [16] G. Vosselman and S. Dijkman, "3D building model reconstruction from point clouds and ground plans," in *Proceedings of the ISPRS Workshop: Land Surface Mapping and Characterization Using Laser Altimetry*, pp. 37–43, Annapolis, MD, USA, October 2001.
- [17] R. B. Rusu, N. Blodow, and M. Beetz, "Fast point feature histograms (FPFH) for 3D registration," in *Proceedings of the 2009 IEEE International Conference on Robotics and Automation*, Kobe, Japan, May 2009.
- [18] F. Tombari, S. Salti, and L. D. Stefano, "Unique signatures of histograms for local surface description," in *Proceedings of the 11th European Conference on Computer Vision ECCV 2010*, Heraklion, Crete, Greece, September 2010.
- [19] R. Q. Charles, H. Su, M. Kaichun, and L. J. Guibas, "PointNet: deep learning on point sets for 3D classification and segmentation," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 77–85, Honolulu, HI, USA, July 2017.
- [20] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: deep hierarchical feature learning on point sets in a metric space," 2017, <http://arxiv.org/abs/1706.02413>.
- [21] M. S. W. W. X. D. Yangyan Li, R. Bu, and B. Chen, "PointCNN: convolution on \mathcal{X} -transformed points," in *Proceedings of the Advances in Neural Information Processing Systems (NIPS)*, pp. 828–838, Denver, CO, USA, December 2018.
- [22] M. Atzmon, H. Maron, and Y. Lipman, "Point convolutional neural networks by extension operators," *International Conference on Computer Graphics and Interactive Techniques*, vol. 37, p. 71, 2018.
- [23] M. Jiang, Y. Wu, T. Zhao, Z. Zhao, and C. Lu, "PointSIFT: a SIFT-like network module for 3D point cloud semantic segmentation," 2018, <http://arxiv.org/abs/1807.00652>.
- [24] M. Gori, G. Monfardini, and F. Scarselli, "A new model for learning in graph domains," in *Proceedings of the 2005 IEEE International Joint Conference on Neural Networks*, pp. 729–734, Montreal, Que, Canada, December 2005.
- [25] F. Scarselli, M. Gori, A. C. Ah Chung Tsoi, M. Hagenbuchner, and G. Monfardini, "The graph neural network model," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 61–80, 2009.
- [26] Y. Wang, Y. Sun, Z. Liu, S. E. Sarma, M. M. Bronstein, and J. M. Solomon, "Dynamic graph CNN for learning on point clouds," *ACM Transactions on Graphics*, vol. 38, p. 146, 2019.
- [27] K. Zhang, M. Hao, J. Wang, C. W. de Silva, and C. Fu, "Linked dynamic graph CNN: learning on point cloud via linking hierarchical features," 2019, <http://arxiv.org/abs/1904.10014>.
- [28] G. Te, W. Hu, Z. Guo, and A. Zheng, "RGCNN: regularized graph CNN for point cloud segmentation," 2018, <http://arxiv.org/abs/1806.02952>.
- [29] X. Gao, W. Hu, and Z. Guo, "Exploring structure-adaptive graph learning for robust semi-supervised classification," 2020, <http://arxiv.org/abs/1904.10146>.
- [30] Q. Lu, C. Chen, W. Xie, and Y. Luo, "PointNGCNN: deep convolutional networks on 3D point clouds with neighborhood graph filters," *Computers & Graphics*, vol. 86, pp. 42–51, 2020.

- [31] A. Vaswani, N. Shazeer, N. Parmar et al., “Attention is all you need,” 2017, <http://arxiv.org/abs/1706.03762>.
- [32] V. Mnih, N. Heess, A. Graves, and K. Kavukcuoglu, “Recurrent models of visual attention,” 2014, <http://arxiv.org/abs/1406.6247>.
- [33] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Liò, and Y. Bengio, “Graph attention networks,” 2018, <http://arxiv.org/abs/1710.10903>.
- [34] C. Chen, L. Z. Fragonara, and A. Tsourdos, “GAPNet: graph attention based point neural network for exploiting local feature of point cloud,” 2019, <http://arxiv.org/abs/1905.08705>.
- [35] L. Wang, Y. Huang, Y. Hou, S. Zhang, and J. Shan, “Graph attention convolution for point cloud semantic segmentation,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 10296–10305, Long Beach, CA, USA, January 2019.
- [36] M. Feng, L. Zhang, X. Lin, S. Z. Gilani, and A. Mian, “Point Attention network for semantic segmentation of 3D point clouds,” 2019, <http://arxiv.org/abs/1909.12663>.
- [37] M. Niepert, M. Ahmed, and K. Kutzkov, “Learning convolutional neural networks for graphs,” 2016, <http://arxiv.org/abs/1605.05273>.
- [38] J. Li, B. M. Chen, and G. H. Lee, “SO-Net: self-organizing network for point cloud analysis,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 9397–9406, Salt Lake City, UT, USA, June 2018.
- [39] L. Yu, X. Li, C.-W. Fu, D. Cohen-Or, P.-A. Heng, and PU-Net, “Point cloud upsampling network,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 2790–2799, Salt Lake City, UT, USA, June 2018.
- [40] R. Klokov and V. Lempitsky, “Escape from cells: deep kd-networks for the recognition of 3D point cloud models,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 863–872, Honolulu, HI, USA, July 2017.
- [41] Y. You, Y. Lou, Q. Liu et al., “Pointwise rotation-invariant network with adaptive sampling and 3D spherical voxel convolution,” 2018, <http://arxiv.org/abs/1811.09361>.
- [42] W. Wu, Z. Qi, and L. Fuxin, “PointConv: deep convolutional networks on 3D point clouds,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 9621–9630, Long Beach, CA, USA, June 2019.
- [43] H. Thomas, C. R. Qi, J.-E. Deschaud, B. Marcotegui, F. Goulette, and L. J. Guibas, “KPConv: flexible and deformable convolution for point clouds,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 6411–6420, Long Beach, CA, USA, June 2019.
- [44] Y. Zhao, T. Birdal, H. Deng, and F. Tombari, “3D point Capsule networks,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1009–1018, Long Beach, CA, USA, June 2019.
- [45] N. Srivastava, H. Goh, and R. Salakhutdinov, “Geometric capsule autoencoders for 3D point clouds,” 2019, <http://arxiv.org/abs/1912.03310>.
- [46] A. Cheraghian and L. Petersson, “3DCapsule: extending the capsule architecture to classify 3D point clouds,” in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1194–1202, Long Beach, CA, USA, June 2019.
- [47] Y. Xu, T. Fan, M. Xu, L. Zeng, and Y. Qiao, “SpiderCNN: deep learning on point sets with parameterized convolutional filters,” in *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 87–102, Munich, Germany, September 2018.
- [48] D. Boscaini, J. Masci, S. Melzi, M. M. Bronstein, U. Castellani, and P. Vandergheynst, “Learning class-specific descriptors for deformable shapes using localized spectral convolutional networks,” *Proceedings of the Eurographics Symposium on Geometry Processing*, vol. 34, pp. 13–23, 2015.
- [49] L. Yi, H. Su, X. Guo, and L. Guibas, “SyncSpecCNN: synchronized spectral CNN for 3D shape segmentation,” in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6584–6592, Honolulu, HI, USA, July 2017.
- [50] M. Defferrard, X. Bresson, and P. Vandergheynst, “Convolutional neural networks on graphs with fast localized spectral filtering,” in *Proceedings of the 30th International Conference on Neural Information Processing Systems NIPS’16*, pp. 3844–3852, Barcelona, Spain, December 2016.
- [51] H. Xiu, T. Shinohara, and M. Matsuoka, “Dynamic-scale graph convolutional network for semantic segmentation of 3d point cloud,” in *Proceedings of the 2019 IEEE International Symposium on Multimedia (ISM)*, pp. 271–2717, Diego, CA, USA, December 2019.
- [52] N. Verma, E. Boyer, and J. Verbeek, “FeaStNet: feature-steered graph convolutions for 3D shape analysis,” 2018, <http://arxiv.org/abs/1706.05206>.
- [53] J. B. Lee, R. A. Rossi, S. Kim, N. K. Ahmed, and E. Koh, “Attention models in graphs,” *ACM Transactions on Knowledge Discovery From Data*, vol. 13, no. 6, pp. 1–25, 2019.
- [54] Z. Wu, S. Song, A. Khosla et al., “3D ShapeNets: a deep representation for volumetric shapes,” in *Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1912–1920, Boston, MA, USA, June 2015.

Research Article

Research on the Precession Characteristics of Hemispherical Resonator Gyro

Li-Jun Song ¹, Rui Yang,¹ Wang-Liang Zhao,² Xing He,¹ Shaoliang Li,² and You-Jun Ding¹

¹*Xi'an University of Architecture and Technology, School of Information & Control Engineering, Xi'an, Shaanxi 710055, China*

²*Shanghai Aerospace Control Technology Institute, Shanghai 201109, China*

Correspondence should be addressed to Li-Jun Song; songlijun9071@sina.com

Received 18 September 2020; Revised 16 October 2020; Accepted 27 November 2020; Published 28 January 2021

Academic Editor: Ning Wang

Copyright © 2021 Li-Jun Song et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Hemispherical Resonator Gyro (HRG) is a new type gyro with high precision, high reliability, shock resistance, no need of preheating, short start time, and long life. It is a kind of vibrating gyro with standing wave rotating along the sensitive base of annular precession, has a unique application prospect in the field of high precision inertial sensors, and is widely used in unmanned aerial vehicle control in complex environments. Based on the theory of the structure characteristics of the hemispherical resonator, the mathematical model of energy of the resonator is established to research the rule of resonant frequency when the hemispherical resonator is rotated around the central axis. In this paper, the influence of precession factor, which are the top angle, the bottom angle, and wall unevenness of the hemispherical resonator, are analyzed. A series of hemispherical resonator models are constructed by ANSYS software to prove the results of theoretical research. The simulation results show that precession factor of the hemispherical resonator is more sensitive of the top angle than the bottom angle, and the error of angular velocity which is caused by the change of the top angle is larger than that which is caused by the change of the bottom angle.

1. Introduction

The main sensitive part of the HRG is the hemispherical resonator with ultralow damping [1–3]. It is the superior performance of the hemispherical resonator that the HRG which can be widely used in the fields of land, sea, air, and sky electricity. In this paper, the characteristics of the HRG are analyzed with the resonator forming process, performance indicators, and user requirements, the mathematical model of the resonator is established, and the optimum mode and precession factor are determined. It provides the theoretical research basis for large quantity and low cost, so as to accelerate the development and application process of the HRG.

2. The Equation of Motion of Resonator

The hemispherical resonator consists of two parts: the thin-walled hemispherical shell and the supporting rod. The resonator material reaches the maximum value of bending

energy storage because of the thin-walled hemispherical shell, and the support rod is fixed by indium welding, which acts as a constraint and supports the resonator and also carries on the transmission of electrical signals. The paper discussed the mathematical model of the energy of the hemispherical resonator based on the Kirchhoff and gave the specific expressions of resonant frequency and precession factor based on the principle of energy conservation. The influence of resonator parameters on the precession factor caused by process defects such as the bottom angle φ_0 , top angle φ_F , and nonideal wall thickness $h(\varphi)$ is also analyzed by using the mathematical model of the energy resonator.

The coordinates of the hemispherical resonator is shown in Figure 1 [4, 5].

The x -axis is the central axis of the resonator; R and $h(\varphi)$ are the center radius and wall thickness of the hemispherical shell, respectively, φ_0 is the bottom angle of the resonator, which is the angle between the constrained end of the resonator and the center axis, which is determined by the ratio of the radius of the supporting rod of the resonator to

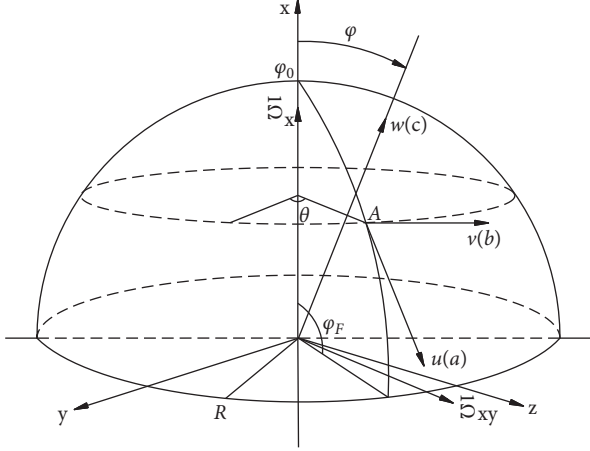


FIGURE 1: The coordinate of the hemispherical resonator.

the radius of the hemispherical shell, and φ_F is the top angle of the resonator, which is the angle between the free end of the resonator and the center axis without any constraint, $\varphi_0 = 0^\circ$ and $\varphi_F = 90^\circ$.

The displacement vector of point A on the shell is

$$\mathbf{V} = \mathbf{a}u + \mathbf{b}v + \mathbf{c}w, \quad (1)$$

where u , v , and w are the displacement components of shell, tangent, and radial, respectively, and \mathbf{a} , \mathbf{b} , and \mathbf{c} are the corresponding dynamic vectors, respectively.

The deformation and stress of the resonator are constrained by the elastic mechanics of the thin shell. Assuming that the resonator rotates at an angular velocity $\Omega = \Omega_x + \Omega_{yz}$ in the inertial space, the mode of the resonator in the rotational space can be expressed as follows:

$$\begin{cases} u(\varphi, \theta, t) = u(\varphi)\cos n(\theta + \psi)\cos \omega_n t, \\ v(\varphi, \theta, t) = v(\varphi)\sin n(\theta + \psi)\cos \omega_n t, \\ w(\varphi, \theta, t) = w(\varphi)\cos n(\theta + \psi)\cos \omega_n t, \end{cases} \quad (2)$$

$$\psi = \int_{t_0}^t P dt = K \int_{t_0}^t \Omega dt,$$

where $u(\varphi)$, $v(\varphi)$, and $w(\varphi)$ are the modes distributed along each direction; ψ is the precession angle of the modes; n is circumferential waves; ω_n is the resonant frequency; K is the precession factor.

According to the theory of elastic thin shell, the elastic potential energy of the hemispherical resonator is as follows:

$$U = \frac{E}{2(1-\mu^2)} \int_S \left[\varepsilon_\theta^2 + \varepsilon_\varphi^2 + 2\mu\varepsilon_\theta\varepsilon_\varphi + \frac{1-\mu}{2}\varepsilon_{\varphi\theta}^2 + \frac{h^2(\varphi)}{12} \left(\lambda_\theta^2 + \lambda_\varphi^2 + 2\mu\lambda_\theta\lambda_\varphi + \frac{1-\mu}{2}\lambda_{\varphi\theta}^2 \right) \right] h(\varphi) dS, \quad (3)$$

where E is the elastic modulus and μ is Poisson's ratio of the material.

In the hemispherical shell, the midplane strain and the midplane bending deformation of the stretchable hemispherical thin shell in formula (3) are as follows:

$$\begin{cases} \varepsilon_\varphi = \frac{(\partial u/\partial \varphi + w)}{R}, \\ \varepsilon_\theta = \frac{(\partial v/\partial \theta + u \cos \varphi + w \sin \varphi)}{R \sin \varphi}, \\ \varepsilon_{\varphi\theta} = \frac{(\partial u/\partial \theta + \partial v/\partial \varphi \sin \varphi - v \cos \varphi)}{R \sin \varphi}, \\ \lambda_\varphi = \frac{(\partial u/\partial \varphi - \partial^2 w/\partial \varphi^2)}{R^2}, \\ \lambda_\theta = \frac{(-1/\sin \varphi \partial^2 w/\partial \theta^2 - \cos \varphi \partial w/\partial \varphi + u \cos \varphi + \partial v/\partial \theta)}{R^2}, \\ \lambda_{\varphi\theta} = \frac{[\partial u/\partial \theta + \sin \varphi \partial v/\partial \varphi - v \cos \varphi + 2(\cos \varphi/\sin \varphi \partial w/\partial \theta - \partial^2 w/\partial \varphi \partial \theta)]}{(R^2 \sin \varphi)}. \end{cases} \quad (4)$$

In the practical system, the top of the hemispherical shell is free and the bottom is constrained and is in the state of microamplitude vibration, so the hemispherical shell satisfies the theory that the normal and tangential strain of the middle plane is zero. $u(\varphi)$, $v(\varphi)$, and $w(\varphi)$ have the following relations:

$$\begin{cases} w(\varphi) = -\frac{du(\varphi)}{d\varphi}, \\ nv(\varphi) + u(\varphi)\cos(\varphi) - \frac{du(\varphi)}{d\varphi}\sin\varphi = 0, \\ mu(\varphi) + v(\varphi)\cos(\varphi) - \frac{dv(\varphi)}{d\varphi}\sin\varphi = 0. \end{cases} \quad (5)$$

Due to the constraint boundary condition at the bottom,

$$u(\varphi_0) = v(\varphi_0) = w(\varphi_0) = 0, \quad (6)$$

it can be obtained

$$\begin{cases} u(\varphi) = v(\varphi) = C_1 \sin\varphi \tan\frac{n\varphi}{2}, \\ w(\varphi) = -C_1(n + \cos\varphi)\tan\frac{n\varphi}{2}, \end{cases} \quad (7)$$

where C_1 is a constant to be determined by the initial excitation condition of the vibration [6–9].

3. The Precession Characteristics of the HRG

3.1. The Working Principle of HRG. HRG is measured by the angular velocity or angle of the input based on the Coriolis principle. When HRG does not rotate, the position of the wave belly point and the wave node is stationary relative to the hemispherical shell, and when the external excitation is applied to the resonator, the energy of the resonator can be converted between different modes to form a stable vibration shape. When the HRG is rotated, the resonant shape of the stable vibration will lag the physical rotation of the gyroscope body, and its hysteresis is about 0.3 times of the rotation angle; the principle is shown in Figure 2.

3.2. The Resonant Frequency of the HRG. When the resonant frequency of the hemispherical resonator is researched, the inertial force acting on the hemispherical resonator should be considered, and it is also necessary to consider the external force acting on the edge of the surface of the resonator.

Assuming the elastic force of the virtual work is δW_K , external force caused by angular velocity Ω of the virtual work is δW_e and vibration inertia force of the virtual work is δT .

According to the reference [10], with the displacement V , the virtual work of the vibration inertia force F can be obtained as follows:

$$\begin{aligned} \delta T &= \int V dF \\ &= \rho\pi R^2 e^{2i\omega_n t} \int_{\varphi_0}^{\varphi_F} (n^2 P^2 + \omega^2)(\sin^2\varphi + 2n\cos\varphi \\ &\quad + n^2 + 1)\tan^{2n}\left(\frac{\varphi}{2}\right)\sin\varphi h(\varphi)d\varphi C_1 \delta C_1. \end{aligned} \quad (8)$$

The external force caused by the angular velocity needs to be considered because the rotating inertia force causes the rotating acceleration $\mathbf{a}(\Omega) = \mathbf{a}(\Omega_x) + \mathbf{a}(\Omega_{yz})$. The virtual work of the rotating inertia force can be obtained as follows:

$$\begin{aligned} \delta W_e &= \rho\pi R^2 e^{2i\omega_n t} \int_{\varphi_0}^{\varphi_F} \left[-4n^2 \sin^3\varphi \tan^{2n}\left(\frac{\varphi}{2}\right)\Omega_x P \right. \\ &\quad \left. + \sin^3\varphi(1+n^2)an^{2n}\left(\frac{\varphi}{2}\right)\Omega_x^2 + f_1(n, \varphi)\Omega_{yz}^2 \right] \\ &\quad \cdot h(\varphi)d\varphi C_1 \delta C_1, \end{aligned}$$

$$\begin{aligned} f_1(n, \varphi) &= \frac{1}{2}(\sin^2\varphi(3 + 2n\cos\varphi + \cos^2\varphi) \\ &\quad + (\cos^2\varphi + n)^2(1 + \cos^2\varphi))\tan^{2n}\left(\frac{\varphi}{2}\right). \end{aligned} \quad (9)$$

The elastic potential energy of the shell is produced by the elastic force, and the virtual work of the elastic force is as follows:

$$\delta W_K = \delta(-U). \quad (10)$$

According to the reference [10], the virtual work of elastic force can be obtained as follows:

$$\begin{aligned} \delta W_K &= -\pi e^{2i\omega_n t} \left\{ \int_{\varphi_0}^{\varphi_F} \frac{E}{(1+\mu)R^2} \frac{n^2(n^2-1)^2}{3\sin^3\varphi} \tan^{2n}\left(\frac{\varphi}{2}\right) h^3(\varphi)d\varphi + R^2 \rho \int_{\varphi_0}^{\varphi_F} [(f_2(n, \varphi)\Omega_x^2 + f_3(n, \varphi)\Omega_{yz}^2)] h(\varphi)d\varphi \right\} C_1 \delta C_1, \\ f_2(n, \varphi) &= (n + \cos\varphi)^2(\sin^2\varphi + n^2)(\varphi \cos\varphi + \sin^2\varphi)\tan^{2n}\left(\frac{\varphi}{2}\right) + \sin^4\varphi(\sin^2\varphi - 2n(n + \cos\varphi))\tan^{2n}\left(\frac{\varphi}{2}\right), \\ f_3(n, \varphi) &= \frac{1}{2}(n^2 + \sin^2\varphi)(\cos\varphi + n)^2(\varphi \cos\varphi + 1 + \cos^2\varphi)\tan^{2n}\left(\frac{\varphi}{2}\right) + (\sin^2\varphi(\cos^2\varphi + 1))(\sin^2\varphi - 2n(n + \cos\varphi))\tan^{2n}\left(\frac{\varphi}{2}\right). \end{aligned} \quad (11)$$

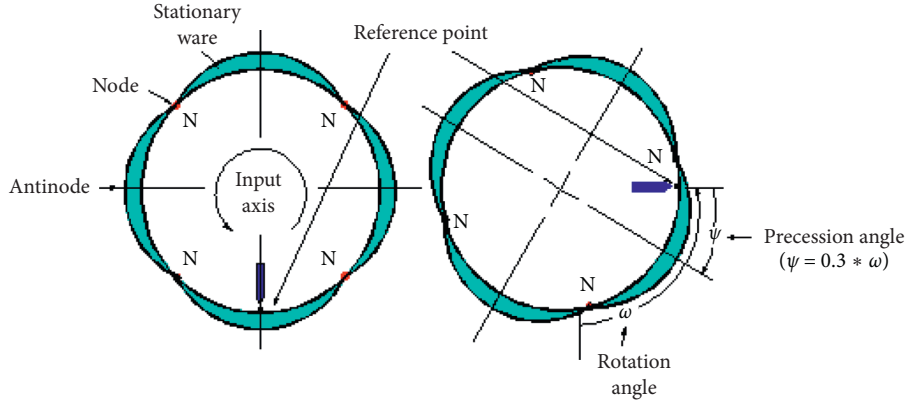


FIGURE 2: The angle hysteresis principle of the hemispherical resonator.

According to the principle of virtual displacement, the total sum of work by virtual displacement is zero, it remains still is that and the result can be obtained:

$$\delta T + \delta W_e + \delta W_K = 0. \quad (12)$$

The frequency of the hemispherical resonator is as follows:

$$\omega_n = \sqrt{\frac{k_0}{k_m}}, \quad (13)$$

where

$$k_m = \int_{\varphi_0}^{\varphi_F} (\sin^2 \varphi + 2n \cos \varphi + n^2 + 1) \tan^{2n} \left(\frac{\varphi}{2} \right) \sin \varphi h(\varphi) d\varphi,$$

$$k_0 = \frac{En^2(n^2 - 1)^2}{\rho(1 + \mu)R^4} \int_{\varphi_0}^{\varphi_F} \frac{1}{3 \sin^3 \varphi} \tan^{2n} \left(\frac{\varphi}{2} \right) h(\varphi) d\varphi. \quad (14)$$

3.3. The Precession Factor of the HRG Mode. When the HRG rotates at an angular velocity Ω in the inertial space, the circular mode turns the ψ angle in reverse at the rate $P = K\Omega_x$ [11, 12].

The inertial forces at any point in the hemispherical resonator shell are

$$F = F_0 + F(\Omega_x) + F(\Omega_{yz}), \quad (15)$$

where F_0 , $F(\Omega_x)$, and $F(\Omega_{yz})$ are the inertia forces when the hemispherical resonator does not rotate, respectively, and the virtual work of the virtual displacement is

$$\delta T = \delta T_0 + \delta T(\Omega_x) + \delta T(\Omega_{yz}) + \delta T(\Omega) - \delta W(\Omega), \quad (16)$$

where

$$\begin{aligned} \delta T_0 &= \omega^2 \pi \rho \cos^2 \omega t \int_{\varphi_0}^{\varphi_F} [u(\varphi) \delta u + v(\varphi) \delta v + w(\varphi) \delta w] Rh(\varphi) d\varphi, \\ \delta T(\Omega_x) &= n^2 P^2 \pi \rho \cos^2 \omega t \int_{\varphi_0}^{\varphi_F} [u(\varphi) \delta u + v(\varphi) \delta v + w(\varphi) \delta w] Rh(\varphi) d\varphi + 2\pi \rho n P \Omega_x \cos^2 \omega t \\ &\quad \cdot \int_{\varphi_0}^{\varphi_F} [\cos \varphi v(\varphi) \delta u + \cos \varphi u(\varphi) \delta v + \sin \varphi v(\varphi) \delta w + \sin \varphi w(\varphi) \delta v] Rh(\varphi) d\varphi, \\ \delta T(\Omega_{yz}) &= 0. \end{aligned} \quad (17)$$

$\delta W(\Omega)$ is the virtual work of the initial elastic force caused by the Ω , that is, δW_{K_0} , and it can be obtained from the principle of virtual displacement:

$$\delta T + \delta W_{K_0} = 0, \quad (18)$$

that is,

$$\delta W_{K_0} - \delta W_K(\Omega) + \delta T_0 + \delta T(\Omega) = 0. \quad (19)$$

In the vibration mode of the resonator, $\delta W_{K_0} - \delta W_K(\Omega)$ is the virtual work of modal elastic force and δT_0 is the virtual work of elastic force. When the HRG rotates at angular velocity $\Omega = \Omega_x + \Omega_{yz}$ in the inertial space, Ω_x is the main vibration. According to the principle that the main vibration energy remains unchanged, when the hemispherical resonator shell rotates, the sum of the main vibration energy and potential energy remains unchanged, and it can be obtained

$$\delta T(\Omega_x) = 0. \quad (20)$$

Suppose

$$\begin{aligned} G_1 &= \int_{\varphi_0}^{\varphi_F} [\cos \varphi v(\varphi) \delta u + \cos \varphi u(\varphi) \delta v + \sin \varphi v(\varphi) \delta w \\ &\quad + \sin \varphi w(\varphi) \delta v] R h(\varphi) d\varphi, \\ G_2 &= \int_{\varphi_0}^{\varphi_F} [u(\varphi) \delta u + v(\varphi) \delta v + w(\varphi) \delta w] R^2 h(\varphi) d\varphi. \end{aligned} \quad (21)$$

Then,

$$nPG_1 + 2G_2\Omega_x = 0. \quad (22)$$

The precession factors can be obtained by finishing:

$$K = \frac{P}{\Omega_x} = -\frac{2}{n} \frac{G_2}{G_1}, \quad (23)$$

where G_1 and G_2 are only related to the shape of vibration, the constraint relation of resonator, and the geometric properties of shell; it also reflects the Coriolis effect and inertial force on the shell.

4. The Parameter of Resonator Effect on the Parameter of the HRG

According to the requirement of high precision, assuming the radius is 15 mm, the error of spherical shell radius is not bigger than 0.014% (about $2 \mu\text{m}$) to ensure the stability of the resonant frequency. The thickness of resonator wall is not bigger than 0.6 mm, and the error is not bigger than 0.067%, about $0.402 \mu\text{m}$. Spherical shell sphericity is not bigger than 0.0003 mm. Bracket radius is 2 mm, and the error support rod is not bigger than 0.13%, about $7.8 \mu\text{m}$. The bottom angle $\varphi_0 = 7.3^\circ$ can be obtained from the ratio of the radius of the supporting rod and the hemispherical shell. The precision of the top angle and the bottom angle is not greater than 0.01° ; the brace bar and hemispherical coaxiality are not bigger than 0.0015 mm, and the quality factor Q is not less than 1×10^7 .

4.1. The Optimal Number of Circumferential Waves. Equations (13) and (23) were used to calculate the resonant frequency and precession factor of the hemispherical resonator, as shown in Table 1.

With the increase of the number of circumferential wave number n , the mode of the resonator is becoming more and more complex, and the resonant frequency and the energy which is to maintain the resonator vibration is also increasing continuously. The precession factor of the hemispherical resonator decreases with the number of circumferential wave.

From the view of the energy loss of the hemispherical resonator, it is hoped that less energy is needed to maintain the mode of hemispherical resonator. Considering the application and vibration realization condition of the HRG, the optimal number of circumferential waves is $n = 2$, and at this time, $K \approx 0.3$.

TABLE 1: The frequency and precession factor of the hemispherical resonator.

	$n = 2$	$n = 3$	$n = 4$	$n = 5$
ω_n (Hz)	905	2804	5620	9307
K $\varphi_0 = 0^\circ$	-0.2978239	-0.0659867	-0.0231877	-0.0102203
K $\varphi_0 = 7.3^\circ$	-0.2981762	-0.0659901	-0.0231877	-0.0102203

4.2. Effect of Top Angle and Bottom Angle on Precession Factor. Assuming the change rate of the mode precession factor K of the resonator is

$$\sigma_K = \frac{\Delta K}{K_0}, \quad (24)$$

where K_0 is the precession factor when the structural parameters of the resonator meets the requirement of design ($\varphi_0 = 7.3^\circ$ and $\varphi_F = 90^\circ$) and ΔK is the variable of precession factor [13, 14].

The precession factor K is

$$K = -\frac{2}{n} \frac{\int_{\varphi_0}^{\varphi_F} 4 \sin^2 \varphi \cos \varphi \tan^n \varphi / 2h(\varphi) d\varphi}{\int_{\varphi_0}^{\varphi_F} (n^2 + 2n \cos \varphi + \cos^2 \varphi - 3 \sin^2 \varphi) \tan^n \varphi / 2h(\varphi) d\varphi}. \quad (25)$$

When the number of circumferential waves is 2, the relationship between φ_F and φ_0 of the resonator and K is shown in Figures 3 and 4:

When the structure parameters of the resonator meet the requirement of design, that is, $\varphi_0 = 7.3^\circ$ and $\varphi_F = 90^\circ$, the precession factor is $K_0 = -0.2981762$; when the change is $[7^\circ, 8^\circ]$, σ_K change is $[-0.0139\%, 0.00037\%]$, and the precession factor K is insensitive to the change of the bottom angle; while when the change is $[89^\circ, 91^\circ]$, change of σ_K is $[1.2425\%, -1.4761\%]$, and the precession factor K is relatively sensitive to the change of the top angle.

The design requires that the precision of the top angle is not bigger than 0.01.

When $\Delta\varphi_0 = 0.01^\circ$ and $\Delta K = 1.2 \times 10^{-6}$, if the input angular velocity is $1^\circ/\text{s}$, the error of angular velocity caused by ΔK is $\Delta\Omega = 0.00432^\circ/\text{h}$.

When $\Delta\varphi_F = 0.01^\circ$ and $\Delta K = 4 \times 10^{-5}$, if the input angular velocity is $1^\circ/\text{s}$, the error of angular velocity caused by ΔK is $\Delta\Omega = 0.144^\circ/\text{h}$.

It can be seen that the error of angular velocity caused by $\Delta\varphi_0$ is far less than that caused by the same $\Delta\varphi_F$.

4.3. Effect of Uneven Wall Thickness on Precession Factor. The processing of high-precision spherical shell structure is the core of the hemispherical resonator gyro technology. According to the material and structural characteristics of the hemispherical resonator, the structure of the resonator is analyzed, which is combined with the technical bottleneck existing in the processing of fused quartz glass, and the processing performance is studied.

For the vibration mode with fixed number of circumferential waves, the uneven wall thickness caused by machining will cause the change of resonance frequency and also affects the precession factor. The influence can be

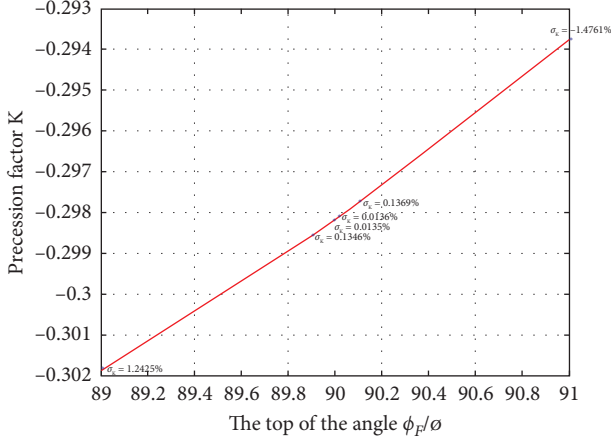


FIGURE 3: The relation between φ_F and K ($\varphi_F = 7.3^\circ$).

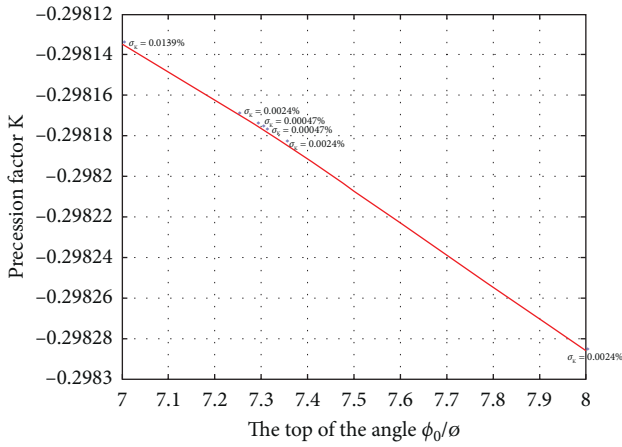


FIGURE 4: The relation between φ_0 and K ($\varphi_F = 90^\circ$).

divided into two cases. (1) When the wall thickness of the bottom of the resonator increases, the variation of the wall thickness is

$$h(\varphi) = h(1 + \alpha \cos \varphi). \quad (26)$$

(2) When the wall thickness of the top of the resonator increases, the function of the wall thickness is

$$h(\varphi) = h(1 + \alpha \sin \varphi). \quad (27)$$

The change of spherical angle φ of the resonator is $0^\circ \leq \varphi \leq 90^\circ$, and α is the change factor of wall thickness.

When the mode shape circumferential wave number is 2, the wall thickness of the bottom and top increases by K ; the changes are shown in Figures 5 and 6.

The absolute value of the precession factor K decreases with the increase of the wall thickness of the bottom. When the change of α is $[0, 0.25]$, the change of σ_K is $[-0.012308\%, -1.1021\%]$.

The absolute value of the precession factor K increases with the increase of the wall thickness of the bottom. When

the change of α is $[0, 0.25]$, the change of σ_K is $[0.0126435\%, 1.0503186\%]$.

5. Design and Analysis of Simulation

5.1. Finite Element Analysis (FEA) of the Resonator. FEA is used to simulate and analyze the actual physical system with the mathematical approximation method. The characteristic changes caused by the rotation of the hemispherical resonator will affect the performance of the HRG, especially the structure of the hemispherical resonator is the key factor affecting the vibration characteristics of the HRG.

Assuming the hemispherical resonator is damped or damped in a free state without rotation, the differential equation of multi-degree-of-freedom motion is

$$M\ddot{x}(t) + C\dot{x}(t) + Kx(t) = Q(t), \quad (28)$$

where $\ddot{x}(t)$ is the acceleration vector, $\dot{x}(t)$ is the velocity vector, $x(t)$ is the displacement vector, and $Q(t)$ is the resonator node load vector. M , C , and K are the mass matrix, damping matrix, and stiffness matrix of the hemispherical resonator, respectively [15].

Equation (29) in the time domain is transformed into a complex variable in the Laplace domain p , and the initial displacement and velocity are assumed to be zero, then the Laplace domain equation is obtained:

$$Mp^2 + Cp + K = Q(p). \quad (29)$$

The transfer function is

$$H(p) = \frac{1/M}{p^2 + (C/M)p + (K/M)}. \quad (30)$$

Its root is the pole:

$$\lambda_{1,2} = -\left(\frac{C}{2M}\right) \pm \sqrt{\left(\frac{C}{2M}\right)^2 - \left(\frac{K}{M}\right)}. \quad (31)$$

The natural frequencies are defined as $\omega_n = \sqrt{K/M}$.

It is assumed that the energy loss of the hemispherical resonator is very small during an oscillation period, and the damping of the hemispherical resonator is approximately ignored in the analysis. The differential equation is simplified as follows:

$$M\ddot{x}(t) + Kx(t) = Q(t). \quad (32)$$

When the node load vector of the hemispherical resonator is 0, the differential equation is further simplified as follows:

$$M\ddot{x}(t) + Kx(t) = 0. \quad (33)$$

Equation (33) is the free vibration equation of the hemispherical resonator, also known as the dynamic characteristic equation. The natural frequencies and modes of the resonator can be solved by this equation.

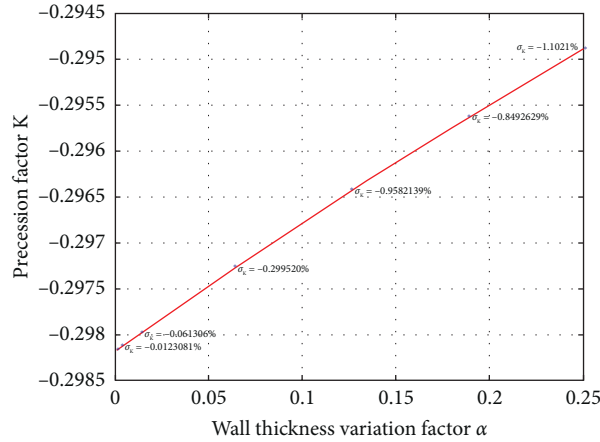


FIGURE 5: The precession factor with increased thickness of the bottom.

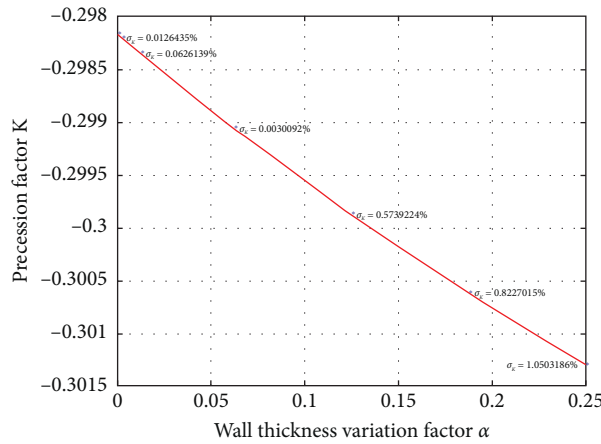


FIGURE 6: The precession factor with increased thickness of the top.

5.2. *Finite Element Simulation of Resonator.* For high-quality hemispherical resonator, on the one hand, it is necessary to ensure the stability of resonant frequency and, on the other hand, to ensure the stability of mode. At present, the ideal material of the hemispherical resonator is fused silica glass. The hemispherical resonator made of this material has stable vibration frequency, short delay time, high mechanical stability, low thermal expansion rate, and large internal stress. The hemispherical resonator made of fused silica glass can work in a wide temperature range and reduces the energy loss needed to maintain the gyro vibration.

The hemispherical resonator requires the fused silica glass to have isotropic and high quality factor, and the temperature coefficient is small and stable; usually, elastic modulus is $E = 7.67 \times 10^{10}$ Pa, Poisson's ratio is $\mu = 0.17$, and density is $\rho = 2500$ kg/m³. In this paper, the boundary constraint of the hemispherical resonator gyro is that the bottom is fixed and the top is free. The specific method used in the simulation is to constrain all the degrees of freedom of the upper end of the support in ANSYS, and the resonator and the following components adopt the free state.

To reduce the workload of the hemispherical resonator modeling, AutoCAD is used to establish the model and then imported into the ANSYS. Finally, the grid generation,

hemispherical resonator model, and grid generation are mapped, as shown in Figures 7(a) and 7(b).

When the bottom of the middle support of the hemispherical resonator is fixed, a force is applied to the hemispherical resonator. The results show that the color of the resonator changes with the color of the lower part of the strut. From the total deformation level of the resonator, the deformation of the hemispherical resonator and the rod is closer to the edge of the hemisphere, and the greater the deformation, the greater the tendency of gradual increase. The total deformed flat view and bottom view of the hemispherical resonator are shown in Figures 8(a) and 8(b).

Under the condition that the bottom of the middle supporting rod of the hemispherical resonator is fixed and the top is free, the stress change is the largest between the hemispherical resonator and the rod, and it gradually decreases from the supporting rod to the outside. The contact surface of the central rod and the hemispherical resonator is the place where the stress changes greatly, and it extends from inside to outside, and with the movement of the hemispherical resonator, some deformation occurs. The resonator's potential view and bottom view are shown in Figures 9(a) and 9(b).

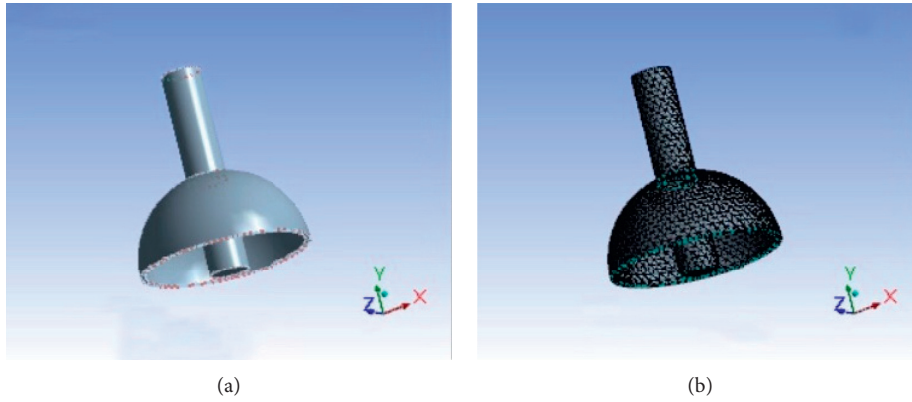


FIGURE 7: The grid and model of the hemispherical resonator.

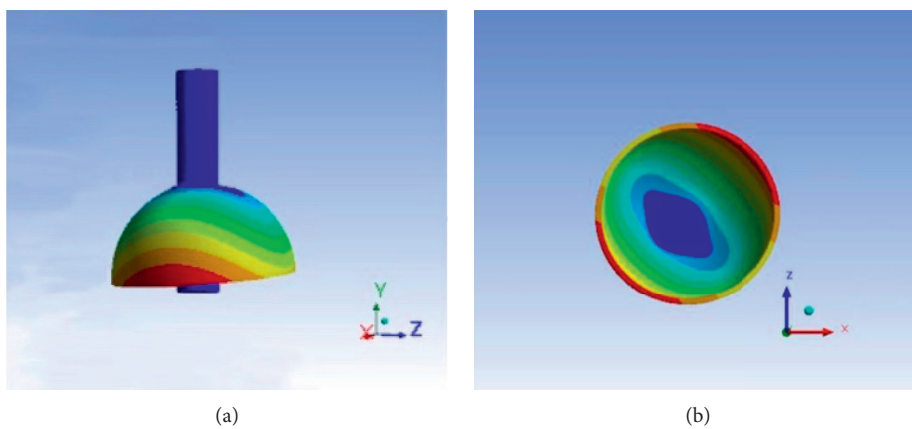


FIGURE 8: The plane view and bottom view of the hemispherical resonator's deformation.

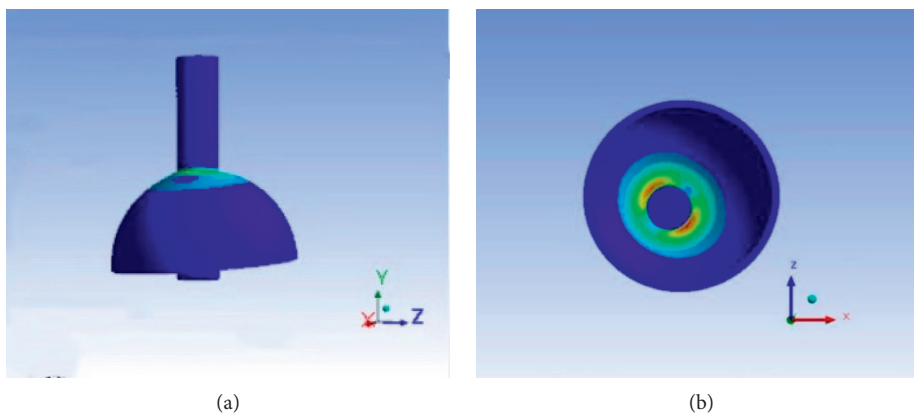


FIGURE 9: The plane view and bottom view of hemispherical resonator's SEQV.

6. Conclusion

The mathematical model of the energy resonator is presented in the paper. The influence of nonideal factors on resonant frequency and precession factor is studied, and the influence of nonideal factors on the precession factor is analyzed. The precession factor of the HRG is more sensitive

to the change of the top angle than to the change of the bottom angle. The angular velocity error of the HRG caused by the change of the top angle is much larger than that caused by the change of the bottom angle. Therefore, in the actual manufacturing process, the manufacturing precision of the top corner is much higher than that of the bottom corner. With the increase of the thickness of the

hemispherical resonator, the stiffness and resonant frequency of the hemispherical resonator will increase. The increase of the resonant frequency makes the hemispherical resonator difficult to excite. However, when the wall thickness of the hemispherical shell decreases, the measurement error will increase due to the asymmetry. Therefore, in the actual manufacturing process, the selection of wall thickness needs comprehensive consideration, and its manufacturing precision is also the most difficult to ensure in precision machining, which is the technical bottleneck of the development of the resonator.

Data Availability

The data required to reproduce these findings cannot be shared at this time as the data also forms part of an ongoing study.

Conflicts of Interest

The authors declare that they do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

Acknowledgments

This work was partially supported by the Natural Science Foundation of Shaanxi Province, project no. 2020JM-488, National Program on Key Basic Research Project, project no. 2017YFC0704200, and Special Scientific Research Project of the Education Department of Shaanxi Province, project no. 20JK0728.

References

- [1] F. Delhaye, "HRG by SAFRAN, the game-changing technology," in *Proceedings of the 5th IEEE International Symposium on Inertial Sensors and Systems*, Moltrasio, Italy, March 2018.
- [2] A. Jeanroy, G. Grosset, J. C. Goudon, and F. Delhaye, "HRG by Sagem from laboratory to mass production," in *Proceedings of the 2016 IEEE International Symposium on Inertial Sensors and Systems*, pp. 1–4, Laguna Beach, CA, USA, February 2016.
- [3] P. Yao, T. Qu, K. Yang, and H. Luo, "Hemispherical resonator gyroscope: current advances and future trends," *Navigation Positioning & Timing*, vol. 4, no. 2, pp. 9–13, 2017.
- [4] S.-L. Gao and J. Wu, "Drift Mechanisms&Control of hemispherical resonant gyro," *Journal of Projectiles, Rockets, Missiles and Guidance*, vol. 28, no. 3, pp. 61–64, 2008.
- [5] Z. Fang, S. Liu, and B. Yu, "Study of basic theories of hemispherical resonator gyros," *Navigation Positioning&Timing*, vol. 28, no. 3, pp. 61–64, 2008.
- [6] V. A. Matveev, M. A. Basarab, and A. V. Alekin, *Etal Solide State Wave Gyro*, pp. 1–43, National Defense Industry Press, Beijing, China, 2009, in Chinese.
- [7] Z. Hong-bo, S. REN, and L. I. wei, "Establishment of dynamics equation of HRG resonator and calculation of natural frequency," *Journal of Harbin Institute of Technology*, vol. 42, no. 11, pp. 1702–1706, 2010.
- [8] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, "Robot learning system based on adaptive neural control and dynamic movement primitives," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 777–787, 2019.
- [9] X. Wang, Z. Fang, W. Wu, B. Lou, and L. I. Yun, "Procession analysis of HRG resonator based on two dimensional mass vibrations," *Journal of Chinese Inertial Technology*, vol. 19, no. 5, pp. 621–626, 2011.
- [10] S. Fan, G. Liu, and Z. Wang, "Theory and experiment on hemispherical resonator gyro," *Measurement&Control Technology*, vol. 1, pp. 29–33+2, 1994.
- [11] B. Qiu, J. Wang, and P. Li, "Full digital control of hemispherical resonator gyro under force-to-rebalance mode," *IEEE Sensors Journal*, vol. 15, no. 1, pp. 71–75, 2015.
- [12] Z. Xu, G. Yi, Z. Wei, and W. Zhao, "A dynamic modeling method for resonator of hemispherical resonator gyro," *Acta Aeronautica et Astronautica Sinica*, vol. 39, no. 3, pp. 139–149, 2018.
- [13] S. Fan and G. Liu, "Finite element analysis of coupling vibration of Resonator for HRG," *Chinese Journal of Scientific Instrument*, vol. 3, pp. 281–287, 1995.
- [14] B. Yu, Z. Fang, and C. Jiang, "Finite element analysis based on optimized HRG resonator," *Piezoelectrics &Acoustooptics*, vol. 37, no. 4, pp. 561–564, 2015.
- [15] B.-C. Shen, Y. I. Guo-xing, S.-Q. Ren, W. Chang-Hong, and Z. Fang, "Finite element analysis on resonator's characteristics of HRG," *Journal of Chinese Inertial Technology*, vol. 6, pp. 58–62, 2004.

Research Article

Effect of Bird Yaw/Pitch Angles on Soft Impact Damage of a Fan Assembly

Junjie Li ^{1,2} Yunfeng Lou ³ Gaoyuan Yu ^{1,2} Tong Li ^{1,2} and Xianlong Jin ^{1,2}

¹School of Mechanical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China

²State Key Laboratory of Mechanical System and Vibration, Shanghai Jiao Tong University, Shanghai 200240, China

³Department of Structural Design, Aerospace System Engineering Shanghai, Shanghai 201108, China

Correspondence should be addressed to Xianlong Jin; jxlong@sjtu.edu.cn

Received 15 September 2020; Revised 4 December 2020; Accepted 24 December 2020; Published 16 January 2021

Academic Editor: Ning Wang

Copyright © 2021 Junjie Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents a numerical investigation of bird attitude angles affecting the soft-impact damage of a full fan assembly. Firstly, considering the geometry of a mallard, a real bird model is established by the Smoothed Particle Hydrodynamics (SPH) method and calibrated with available test data. Then, complying with airworthiness requirements, simulations of a full-bladed fan assembly subjected to a real bird were conducted to determine the critical ingestion parameters (CIP). Furthermore, a real bird with different attitude angles aimed at a full fan assembly was simulated. Results show that attitude angles of the bird produce a significant impact on the effect of the bird strike on rotating blades and would increase the possibility of blade failures, especially for the yaw angle of -45° and the pitch angle of -60° . It is invaluable for commercial airlines and engine manufactures to provide safe flight and landing by adopting the real bird model with critical yaw and pitch angles in the design for resistance to bird ingestion.

1. Introduction

Bird strikes have been presenting the main threat to aircrafts since the beginning of aviation history. All available statistics indicate that the bird-strike hazard is increasing dramatically, due to the significant expansion of wild bird populations, as well as, to some extent, because of the steady increase in air transport [1]. Therefore, aviation authorities require that all forward facing components need to prove a certain level of bird-strike resistance in certification tests before they are allowed for operational use [2]. However, according to a large number of reported bird-strike incidents, substantial damage to aircraft structures occurs even though the involved energies of the bird did not achieve the aircraft certification standard. It indicates that only taking the mass and impact speed of the bird into account is far from enough.

Initially, early studies were commonly based on theoretical and experimental studies. Wilbeck [3, 4] conducted a comprehensive set of experimental studies on bird impacts

on various structures regarding different bird sizes, initial velocities, bird substitutes, and oblique impacts. It was found that real birds behave as fluids during impact at velocities larger than 100 m/s, and the impact process consists of four typical phases. In recent times, much effort and significant progress have been done to get better insight into simulating the process of bird-strike events, involving substitute bird models [5, 6], numerical methods [7–9], and material models [10, 11]. It can be concluded that the substitute model with bird shapes, namely, ellipsoid and hemispherical-ended cylinder (aspect ratio of 2), is recommended, associated with material properties of homogenized fluidic materials. The equation of state is used to describe the compressibility characteristics of bird material.

However, the values of Hugoniot pressure obtained from theoretical and experimental results are so far from each other, especially at lower velocities [12]. Meanwhile, the values obtained from numerical results calculated by different authors are in a wide range between the experimental and theoretical values [2, 12]. Therefore, some scholars

shifted the focus to the real bird model. Lakshmi [13] used a multimaterial bird model with a more realistic bird shape to capture a more detailed impact load spectrum. McCallum et al. [14] developed a physically representative bird model of a Canadian goose. The results show that, compared with the traditional model, the physically representative bird model produces a lower Hugoniot pressure and higher magnitude of peak impact force with longer duration. Hedayati et al. [12, 15] established a real bird model based on the mallard CT-scan image data. They found that the numerical results of the realistic bird model are closer to the available experimental results than those in the case of the traditional model.

Bird strikes are the major factor of blade damage for aircraft engines [16]. It is worth noting that, in a large number of incidents, the involved bird energy was lower than the magnitude of impact energy in certification, whereas aircraft structures can be substantially damaged [17]. In a typical field-event of the bird strike, it is common to observe a single bird coming into contact with multiple blades with respect to arbitrary attitude angle. Projectile yaw during impact would result in a variation in the impact loading history [4]. On that point, several research works have been done to capture the amount of damage imposed on blades due to bird impact, considering bird orientation or pitch/yaw angles [18–20]. The results reveal that bird orientation has a significant effect on the impact force. However, with respect to a realistic bird shape, not only is the bird attitude far more complicated, but also the effect of bird orientation on rotating blades during the bird strike is different for yaw and pitch angles. Moreover, it is difficult and complicated to record attitude angles of bird both in physical tests and in a field-event of bird strikes. The high-pressure gas cannon is not capable of firing a real bird or a substitute bird with arbitrary yaw/pitch angle. Biologically inspired motion modelling and control algorithms [21–25] may seem like a good choice to determine the bird attitude, which is urgent to study. Attitude angles in our research are assumed in a wide range from 0° to 60° .

The study presented in this paper aims to focus on attitude angles of a realistic bird affecting soft impact damage of jet engine blades. Considering the shape of a real mallard in a published literature [12], a new bird model was established using the SPH approach and validated against the latest published experimental results [26]. To determine the influence of bird orientation on the blade damage, the realistic bird model with various attitude angles targeted at a fan assembly has been developed. The simulations were performed based on Magic Cubic-II of Shanghai Super-computer Centre, using LS-DYNA FE code.

2. Theoretical Background

2.1. Soft Impact Theory. Figure 1 shows the main stages and the pressure profile of a typical bird-strike process, a normal impact of a flat cylinder on a rigid plate. At the moment of impact in Figure 1, the bird material is rapidly decelerated, and a shock wave is initiated at the bird-target interface, resulting in a sharp rise in pressure. The shockwave pressure

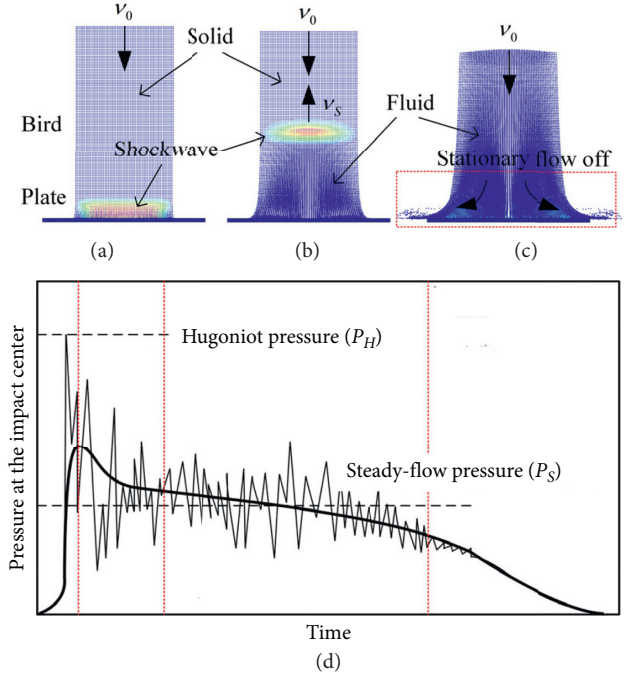


FIGURE 1: Main phases and pressure profile of a bird strike event. (a) Solid-structure impact. (b) Solid-fluid transition. (c) Fluid-structure impact. (d) Impact pressure profile.

exceeds the strength of bird material to a large extent. Consequently, as the shockwave propagates through the bird body in Figure 1, it rapidly breaks the internal bonds of birds, generating a transition from a solid towards a fluid phase. The bird material is transitioned to a fluid-like medium. The high-pressure gradient across the free surface of bird and the surrounding air forces out shocked material radially [27]. This behavior is known as shock release. With the propagation of release waves towards the center of the bird, the pressure of the bird material gradually decays to the fluid pressure. As the bird strike progresses in Figure 1, the bird material is progressively forced out of the original bird volume and spreads outwards nonlinearly. With the bird tail approaching the target, the bird-strike pressure decays to zero. Figure 1 shows the pressure at the center of the impact.

The shockwave pressure (Hugoniot pressure, P_H) at the initial impact (see Figure 1) is determined by [28]

$$P_H = \rho_0 \nu_S \nu_0, \quad (1)$$

in which

$$\nu_S = (1 - z)(C + S_1 \nu_0) + z \nu_a, \quad (2)$$

where ρ_0 is the initial density of the bird. ν_S , ν_0 , and ν_a represent the speed of shock wave, impact speed of the bird, and sound speed, respectively. z represents bird porosity. C and S_1 are coefficients of the relationship between the shock and particle velocities.

The stagnation pressure P_S during the liquid impact (see Figure 1) is given by the following equation [28]:

$$P_S = \frac{k}{1-z} \rho_0 v_0^2, \quad (3)$$

where the constant k is 0.5 for an incompressible fluid [27].

2.2. Loads on Blades during Impact. In a practical fan blade application, the impact progress is far more complex. Figure 2 indicates the bird-slicing action during the impact. The bird is represented as a flat cylinder. As soon as a bird comes into contact with blades, it undergoes cut into several slices by the blade leading edge in the direction of the relative velocity V_{Relative} .

Blades subjected to a bird are invariably an “oblique impact” event, consisting of two phases [29]: the bird-slicing action by multiple rotating blades; each individual bird slice travelling along the blade airfoil. Thus, the impact generates both a slicing-impact load on the blade leading edge and a bird-slice turning load acting on the concave surface, which can be expressed as [29]

$$F_{\text{Bird-slice}} = F_{\text{slicing}} + F_{\text{travel}}. \quad (4)$$

At the moment of the leading edge slicing a bird projectile, it produces a high-intensity shock wave. For the initial impact area on the concave surface is almost a point, the effective load generated by the shock-wave pressure is not significant, and thus its contribution is ignored [29]. Thus, the slicing-impact load is determined by the slicing stagnation pressure [29].

$$F_{\text{slicing}}(t) = B_a(t) P_{\text{stagnation}}, \quad (5)$$

in which the slicing stagnation pressure is determined by

$$P_{\text{stagnation}} = \frac{1}{2} \rho_0 (v_{\text{impact}})^2, \quad (6)$$

where $B_a(t)$ denotes the bird-foot-print area on the blade leading edge. v_{impact} represents the normal component of the relative velocity V_{Relative} (see Figure 2) on the blade leading edge.

2.3. Airworthiness Standards of Aircraft Engines. Currently, aviation authorities require that all new commercial aero-engines must substantiate physical certification tests before operational use. These requirements are compiled in the Federal Aviation Regulations (FAR), Chinese Civil Aviation Regulations (CCAR), and the Certification Specifications (CS) of the European Aviation Safety Agency (EASA) [2]. According to airworthiness standards of FAR §33.76 [30], for the inlet throat area 2.37 m^2 of the jet engine presented in this study, it must be substantiated that the fan assembly is subjected to a medium bird of 1.15 kg under FAR §33.76(c) (3), aimed at the most critical location outboard of the primary core flow path [30]. Figure 3 indicates the location of target point on the first exposed rotating stage of the engine. The airfoil height is measured at the leading edge of the blade. The target point for bird ingestion is determined by impact loading on rotating blades, as well as the possibility of blade failures [31].

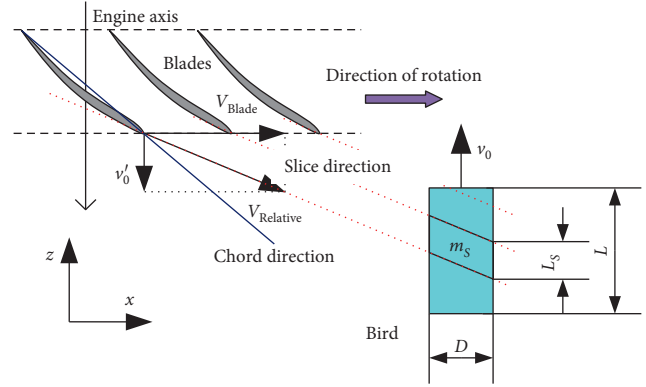


FIGURE 2: The bird slicing action by the blade leading edge.

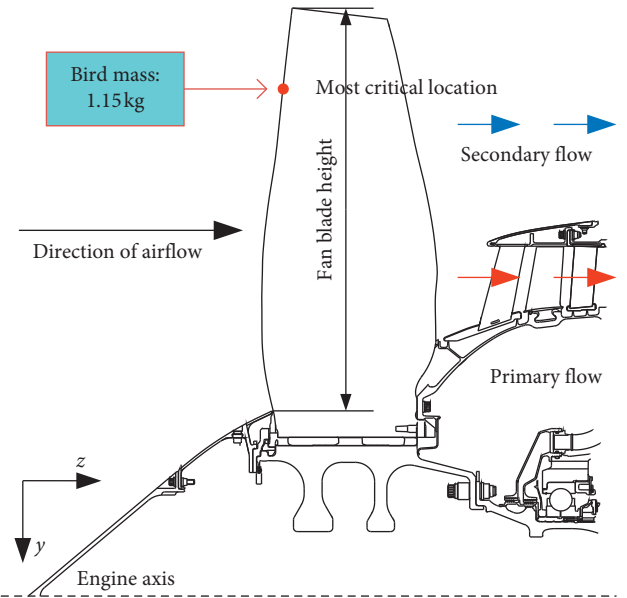


FIGURE 3: Location of target point on the leading edge of a fan blade.

3. Bird Modelling

3.1. The SPH Method. The SPH method is increasingly used in bird-strike simulations as it has already been proved to be quite capable of simulating high deforming matter with defragmentation [14]. With the SPH technique, the bird was represented as a set of discrete particles, in which the interaction between particles was achieved through a kernel function rather than a structured mesh [12]. The SPH method is recommended in the simulation of the bird-strike process, because of its high stability, low cost, and good correlation with experimental observations in terms of scattering particles [2]. Therefore, the SPH approach was adopted to model a real bird of mallard.

3.2. Geometry of a Real Bird. According to the reported bird strike incidents resulting in substantial damage to civil aircraft components in the period 1990–2017 [32], the waterfowl had been the most threatened species for civil aviation safety,

accounting for 28% of the total specified species. Therefore, the mallard (a typical species of waterfowl) is represented as a general realistic bird model. Since a real bird consists of several internal cavities, bone structures, etc. with complex geometry, the presented bird model can reflect the bird shape to some extent, not exactly the same as the real one. The main geometric characteristics are given as follows: (1) Wilbeck [3] found that bone effect of a bird can be assumed negligible, and thus a uniform density of the real bird model is used; (2) the head is simplified to be an ellipsoid, and the neck was considered as a circular-conical-frustum [19]; (3) geometric parameters of the bird torso were modelled considering the geometry of a real mallard bird in Ref. [12]; (4) the mass of bird wings accounted for 19% of the total mass in accordance with Ref. [14]. The interparticular distance is 2 mm, and the amount of SPH particles is 153,621, as shown in Figure 4.

3.3. Bird Material Constitutive Model. With a general material model for both solids and fluids to describe the fluidic behavior of the bird material, the Cauchy stress tensor is divided into a hydrostatic part and a deviatoric part [16].

$$\sigma_{ij} = -P\delta_{ij} + s_{ij}, \quad (7)$$

in which the hydrostatic pressure P is written as

$$P = -\frac{\sigma_{kk}}{3} = \frac{-(\sigma_{11} + \sigma_{22} + \sigma_{33})}{3}, \quad (8)$$

where σ_{ij} represents stress tensor, and s_{ij} represents deviatoric stress tensor. Variable δ_{ij} represents Kronecker delta symbol.

Since the compression of bird material during impact induces a change in the bird density [27], the Mie–Grüneisen equation of state (EOS) was used to reflect the relationship between the pressure and the density [15].

$$P = \frac{\rho_0 C^2 \mu \left[1 + (1 - (\gamma_0/2))\mu - (a/2)\mu^2 \right]}{\left[1 - (S_1 - 1)\mu - S_2(\mu^2/\mu + 1) - S_3(\mu^3/(\mu + 1))^2 \right]^2} + (\gamma_0 + a\mu)E. \quad (9)$$

A linear EOS is adopted for the bird material model [26], and equation (9) can be written as

$$P = \frac{\rho_0 C^2 \mu}{\left[1 - (S_1 - 1)\mu \right]^2}, \quad (10)$$

where C , S_1 , S_2 , and S_3 are coefficients of the relationship between the shock and particle velocities, and γ_0 represents Mie–Grüneisen gamma. a is the first-order volume correction to γ_0 . μ represents the relative change in density

Porcine gelatine with 10% porosity is used instead of the bird material [26]. Parameters of porcine gelatine can be obtained from Ref. [33]. Thus, for the developed bird model, parameters are given as follows: $\rho_0 = 954 \text{ kg/m}^3$, $C = 1447 \text{ m/s}$, $S_1 = 1.77$.

3.4. Attitude Angle Description. In this research work, bird attitude was defined based on the Tait–Bryan angles. The Tait–Bryan angles are three angles, named as yaw angle α , pitch angle β , and roll angle γ . They were introduced to describe the orientation of a bird. As shown in Figure 5, a right-handed Cartesian coordinate system was defined in the center of bird gravity, and any bird attitude can be parameterized by three Tait–Bryan angles α , β , and γ .

In a field event of the bird strike, it is difficult and complicated to record the Tait–Bryan angles of bird ingestion. Attitude angles of the real bird are assumed in a wide range from 0° to 60° and the orientation of bird based on an individual Tait–Bryan angle was studied at a time. Considering the weight of wings accounting for about 19% bird mass [14], the effect of the roll angle on the slicing action of a bird is relatively small. Therefore, bird attitude based on the roll angle was neglected. Yaw angle α and pitch angle β were selected as $\pm 15^\circ$, $\pm 30^\circ$, $\pm 45^\circ$ and $\pm 60^\circ$, respectively. Figure 6 shows the attitude angles of a realistic bird, where the attitude angle is represented as the angle between the roll axis and the impact velocity.

4. Numerical Model of a Fan Assembly

4.1. Fan Assembly Model. The fan assembly consists of 24 equally spaced (15°) wide-chord blades and a fan disc. The fan assembly was modelled with 8-noded solid elements. Bending is the basic mode of load carrying capacity for blades during impact. With a single-point integration at the element centroid, solid elements can carry membrane stresses only [34]. Thus, blades were assigned 3 layers of solid elements through the thickness. A tied-contact relationship was assumed to represent the attachment between the blade and the fan hub. Fixed boundary constraints of z -displacement direction were defined on the side of the hub component. To restrict nonphysical deformations relevant to zero energy modes, stiffness hourglass control with exact volume integration [34] was applied to the simulation model. Figure 7 shows the finite element model of the fan assembly. The fan assembly consists of 165,200 solid elements with a total of 222,912 nodes. The minimum element size of the fan assembly model was 0.42 mm at the tip of the blade leading edge, and the corresponding time step for explicit dynamic simulation was $4.06\text{E-}8 \text{ s}$. In addition, Figure 7 gives numbers to fan blades during slicing a bird so that it can identify the blade damage in the simulation of the impact on a fully bladed fan rotor.

4.2. Blade Material Constitutive Model. Bird strike events can be described as high strains and high strain rates in short duration with considerable intensity [35]. Therefore, the empirical Johnson–Cook relation was selected [36].

$$\sigma_y = (A + B\epsilon^n)(1 + C \ln \dot{\epsilon}^*) (1 - T^{*m}), \quad (11)$$

in which

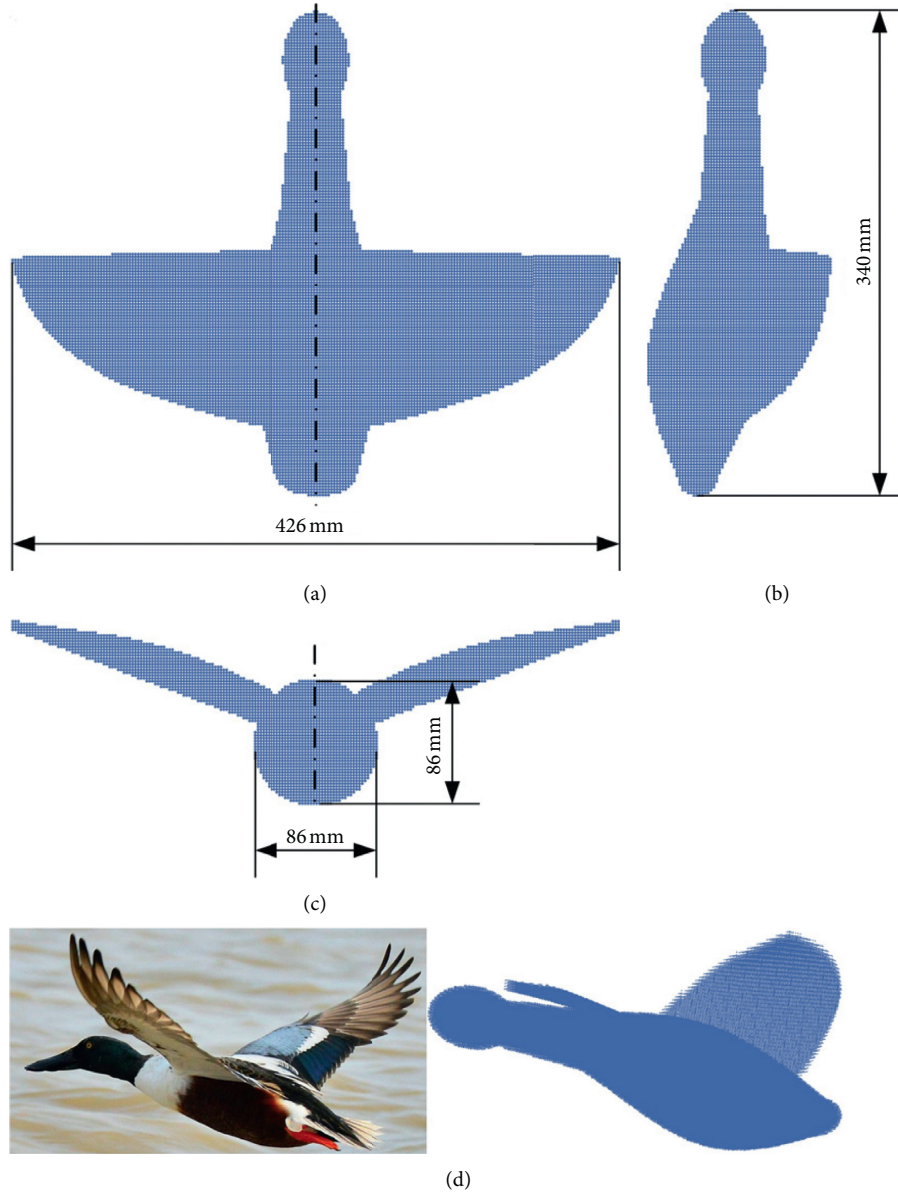


FIGURE 4: A realistic bird model by the SPH method. (a) Left view. (b) Front view. (c) Bottom view. (d) A real mallard compared to the SPH model.

$$T^* = (T - T_{\text{room}})(T_{\text{melt}} - T_{\text{room}}), \quad (12)$$

where σ_y represents the equivalent von Mises stress; ε represents the equivalent plastic strain; and $\dot{\varepsilon}^*$ represents the normalized equivalent plastic strain rate. The parameters A , B , C , m and n are material constants. T , T_{melt} and T_{room} represent the metal temperature, melting temperature, and room temperature, respectively.

Material fracture is determined by a cumulative damage law, a function of mean stress, strain rate, and temperature [36].

$$D = \sum \frac{\Delta \varepsilon}{\varepsilon_f}, \quad (13)$$

in which

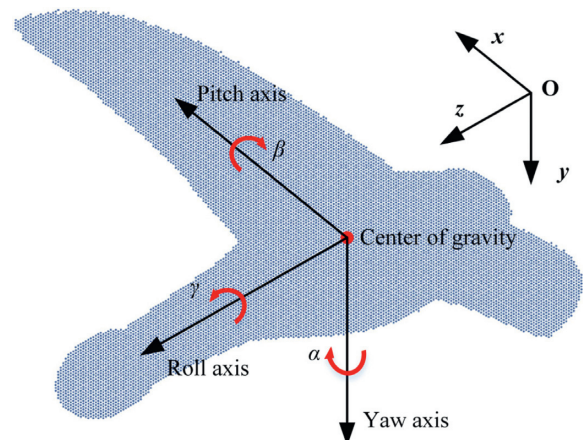


FIGURE 5: Definition of the Tait-Bryan angles.

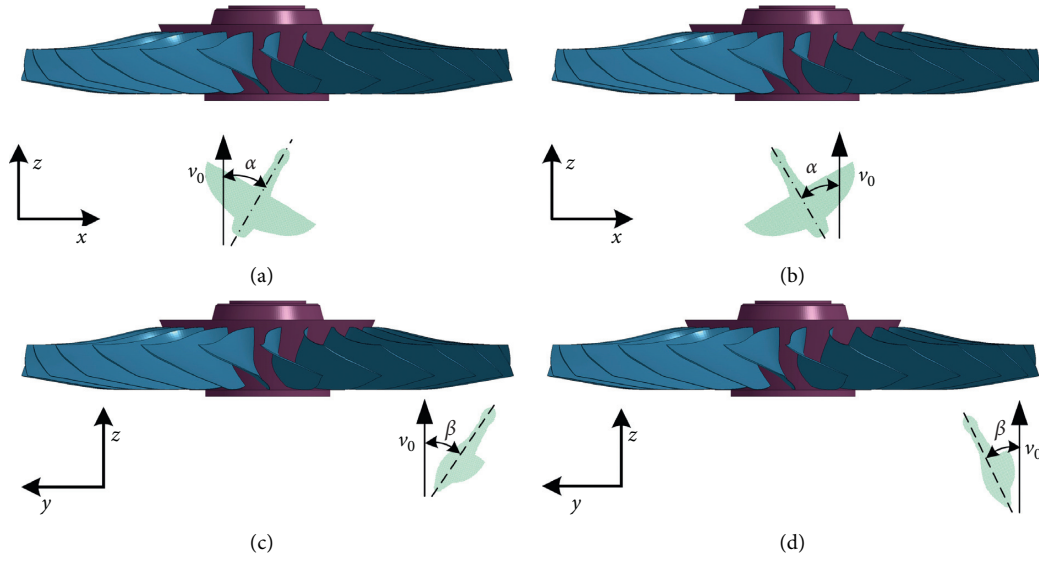


FIGURE 6: Schematic of yaw and pitch angles. (a) Positive yaw angle α . (b) Negative yaw angle α . (c) Positive pitch angle β . (d) Negative pitch angle β .

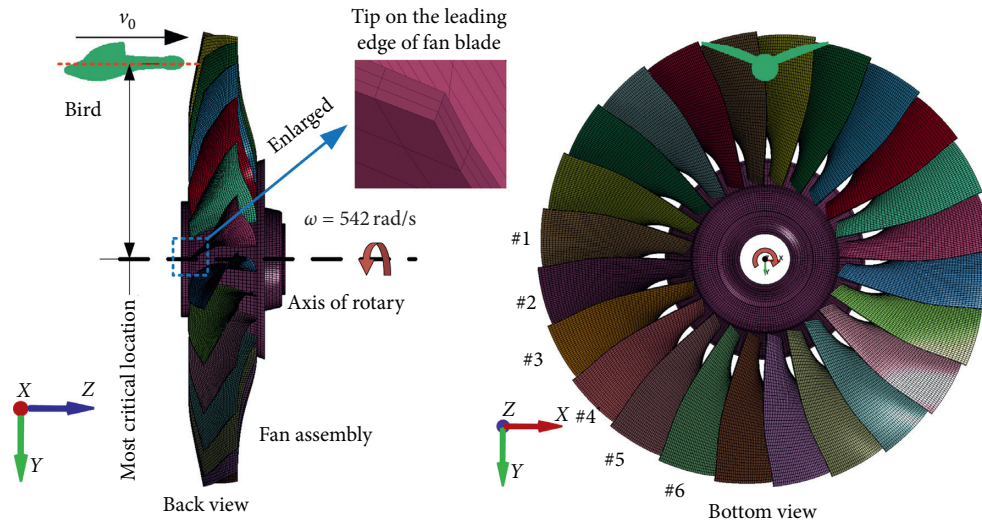


FIGURE 7: FE model of a fan assembly subjected to a bird.

$$\varepsilon_f = [D_1 + D_2 \exp(D_3 \sigma^*)][1 + D_4 \ln \varepsilon^*][1 + D_5 T^*], \quad (14)$$

where $\Delta\varepsilon$ is the increment of equivalent plastic strain, and σ^* is the mean normalized by the equivalent stress. D_1 , D_2 , D_3 , D_4 and D_5 are damage constants. Failures are assumed to occur when $D=1$.

The fan assembly is made of titanium alloy Ti-6Al-4V, and material parameters are derived from Ref. [37], as listed in Table 1. In addition, the Mie-Grüneisen EOS (equation (3)) was defined in conjunction with the material constitutive model, and parameters for Ti-6Al-4V are as follows: $C = 5.13 \times 10^3 \text{ m/s}$, $S_1 = 1.028$, $\gamma_0 = 1.23$ and $a = 0.17$ [35].

4.3. Stress Initialization. The rotating fan assembly undergoes a constant centrifugal force, resulting in significant deformation and initial stresses prior to impact. It is necessary to evaluate the prestress of the rotating components, especially for blades [16]. Thus, a preload analysis procedure was conducted by assigning rotational velocity of 542 rad/s in an implicit solution. The initial stress of blades assembly is shown in Figure 8.

5. Results and Discussion

5.1. Bird Model Calibration. The effect of the high-intensity shock wave on the blade damage can be ignored for the

TABLE 1: Material properties of Ti-6Al-4V.

Material	ρ (kg/m ³)	A (MPa)	B (MPa)	C	m	n	D_1, D_2, D_3, D_4, D_5
Ti-6Al-4V	4.4×10^3	1098	1092	0.014	1.1	0.93	-0.09, 0.25, -0.5, 0.14, 3.87

reason that the initial bird loading area is almost like a point loading [29]. The focus of blade damage analysis needs to be on accurately capturing the steady stagnation pressure phase of impact [27]. Thus, the stagnation pressure is represented as a criterion to calibrate bird models. Numerical simulation of a normal impact on a rigid plate was established in accordance with the experimental setup in Ref. [26]. A segment sensor was assigned at the center of the plate to extract the impact pressure. The stagnation pressures were estimated by averaging the pressure between $T_0/3$ and $2T_0/3$ [26], where T_0 denotes the duration of the bird strike event. The theoretical curve of the stagnation pressure was determined by equation (3), and test data was derived from Ref. [26], as shown in Figure 9. The numerical results were in a range between theoretical and experimental values. The stagnation pressures captured by simulation of the real bird were in closer agreement with the test data than those in the case of the hemispherical-ended cylinder.

5.2. Most Critical Ingestion Parameters. The critical ingestion parameters are required to be identified for the specified bird ingestion [30]. In this section, simulations of a full fan assembly subjected to a real bird were performed to determine the critical location and ingestion speed under FAR §33.76(c) (1) and FAR §33.76(c) (3).

5.2.1. Most Critical Ingestion Speed. Complying with FAR §33.76(c)(1), the critical ingestion speed is required to reflect the most severe situation within the range of speeds used for normal flight operations up to 460 m (1,500 feet), but not less than V1 minimum for airplanes [30]. Therefore, simulations of the bird ingested at different speeds were performed, aimed at the fan blade height of 80%. Figure 10(a) shows the effect of ingestion speed on impact loading history. The peak values of impact force generated by a real bird targeted at speeds of 60 m/s, 65 m/s, 70 m/s, 80 m/s, 105 m/s, and 130 m/s are 183.35 kN at 3.95 ms, 182.53 kN at 3.97 ms, 198.91 kN at 3.48 ms, 210.96 kN at 3.48 ms, 212.16 kN at 2.99 ms, and 170.95 kN at 2.50 ms, respectively. As the speed increases, the impact duration becomes shorter. It is also found that the higher the impact speed is, the earlier the peak force occurs. As the normalized sum of effective plastic strains shown in Figure 10(b), the ingestion speed of 65 m/s caused the most severe plastic strain. Therefore, 65 m/s was represented as the most critical speed.

5.2.2. Most Critical Exposed Location. To find out the most critical location under FAR §33.76(c)(1), the real bird aimed at different target points on the leading edge of blades was evaluated at a speed of 65 m/s. Figure 11(a) shows the effect of target location on the time histories of impact force. The peak values of impact force generated by a real bird aimed at

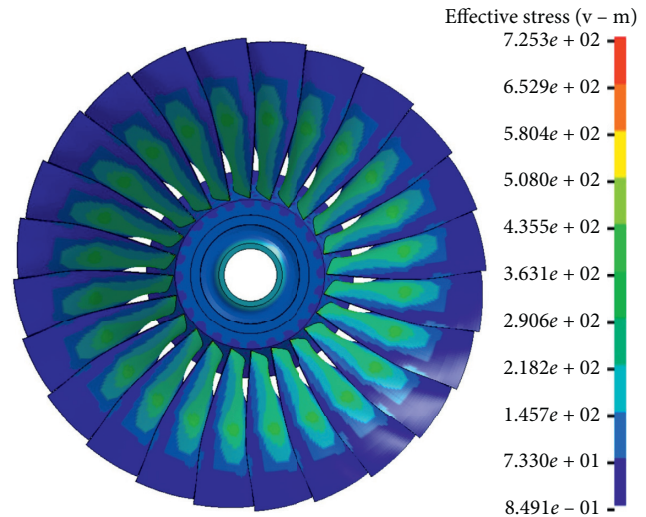


FIGURE 8: von Mises stress (MPa) of the fan assembly.

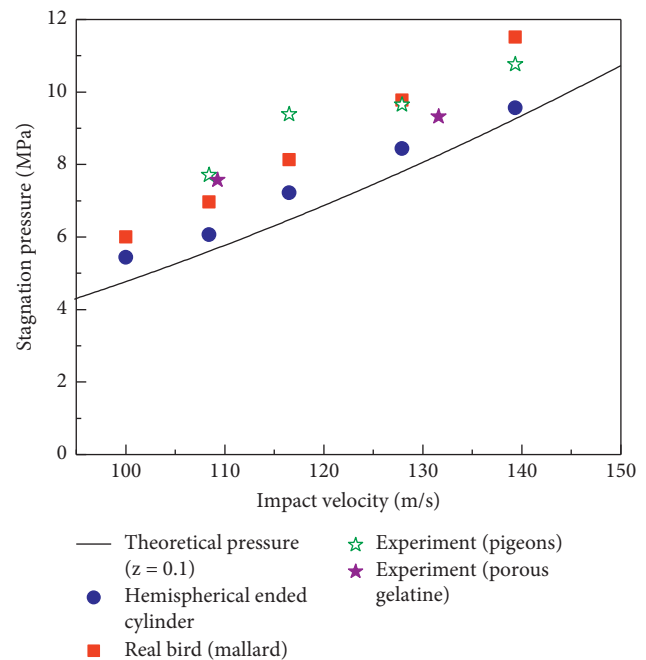


FIGURE 9: Calibration of stagnation pressures.

the fan blade height of 50%, 70%, 75%, 80%, 85%, and 90% are 163.79 kN at 3.84 ms, 168.25 kN at 4.30 ms, 177.30 kN at 3.92 ms, 182.53 kN at 3.97 ms, 155.92 kN at 4.39 ms, and 171.69 kN at 4.85 ms, respectively. It can be observed that the impact location of 80% blade height reaches the maximum value of peak impact force. As the blade damage indicated in Figure 11(b), with the single medium bird targeted at the fan blade height of 80%, it occurs to the most

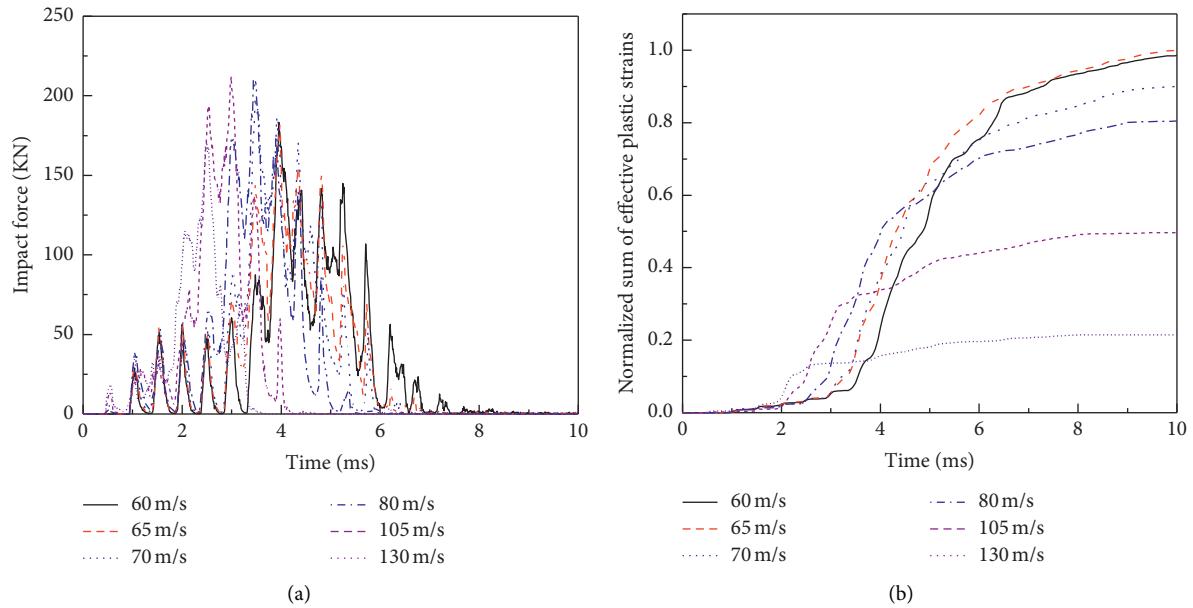


FIGURE 10: Effect of ingestion speed on the impact loading history and blade damage. (a) The impact loading history. (b) Normalized sum of effective plastic strains.

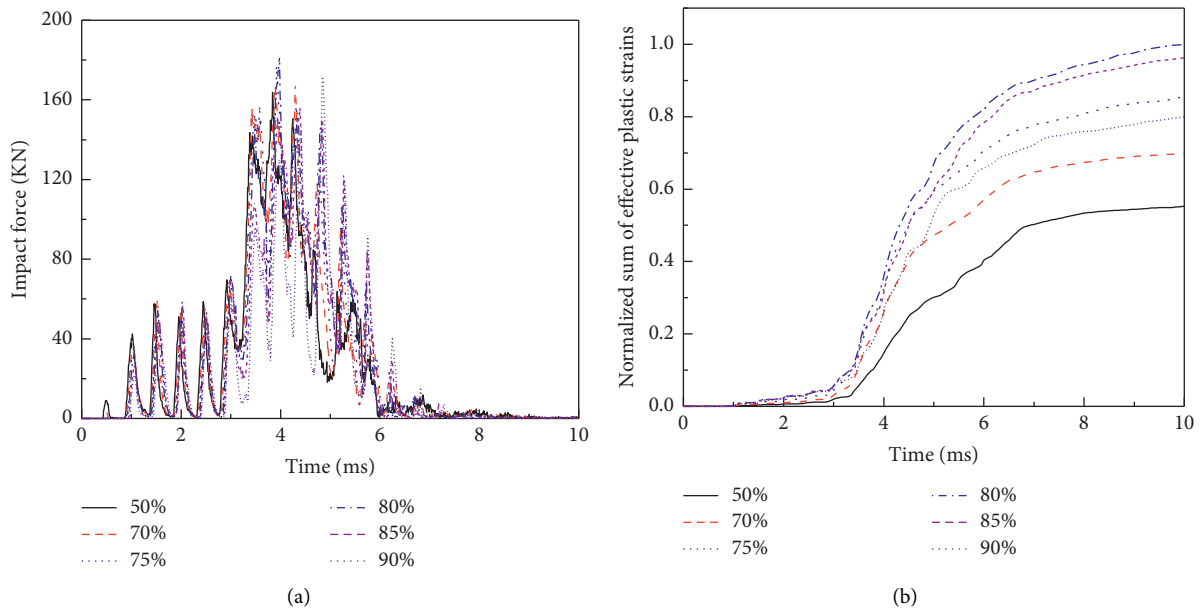


FIGURE 11: Effect of target location on the impact loading history and blade damage. (a) The impact loading history. (b) Normalized sum of effective plastic strains.

severe blade plastic strain. As a result, the most critical location for a middle-sized bird should be aimed at 80% of the fan blade height.

5.3. Effect of Attitude Angles on Soft-Impact Damage.

Figure 12 shows the deformation of bird oriented along the engine axis. As expected, it can be clearly observed that, prior to the impact of the bird torso, the real bird was sliced into more pieces. The orientation of bird has a direct effect on

slicing action of fan blades. With respect to different attitudes, the contact area and duration of bird strikes change, resulting in a significant impact on the effect of the bird strike on rotatory blades and the possibility of blade failures.

Figure 13 shows the effect of the yaw angle on impact loading history. As shown in Figure 13(a), peak values of impact force generated by a real bird with respect to yaw angle of 15° , 30° , 45° and 60° are 201.53 KN at 3.97 ms, 170.52 KN at 4.31 ms, 176.34 KN at 4.33 ms, and 186.97 KN at 4.28 ms, respectively. As shown in Figure 13(b), the peak

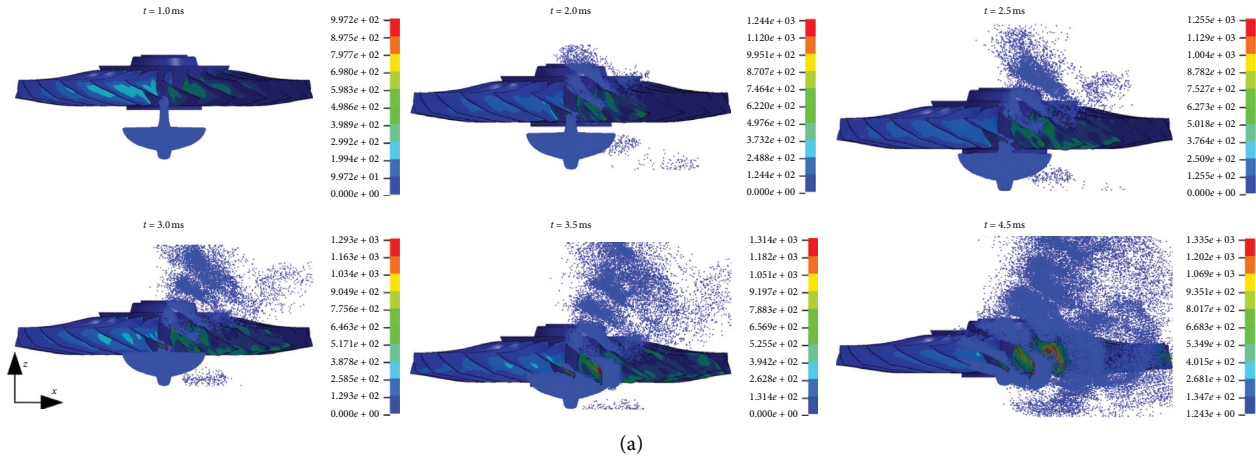


FIGURE 12: Bird deformation during the slicing action.

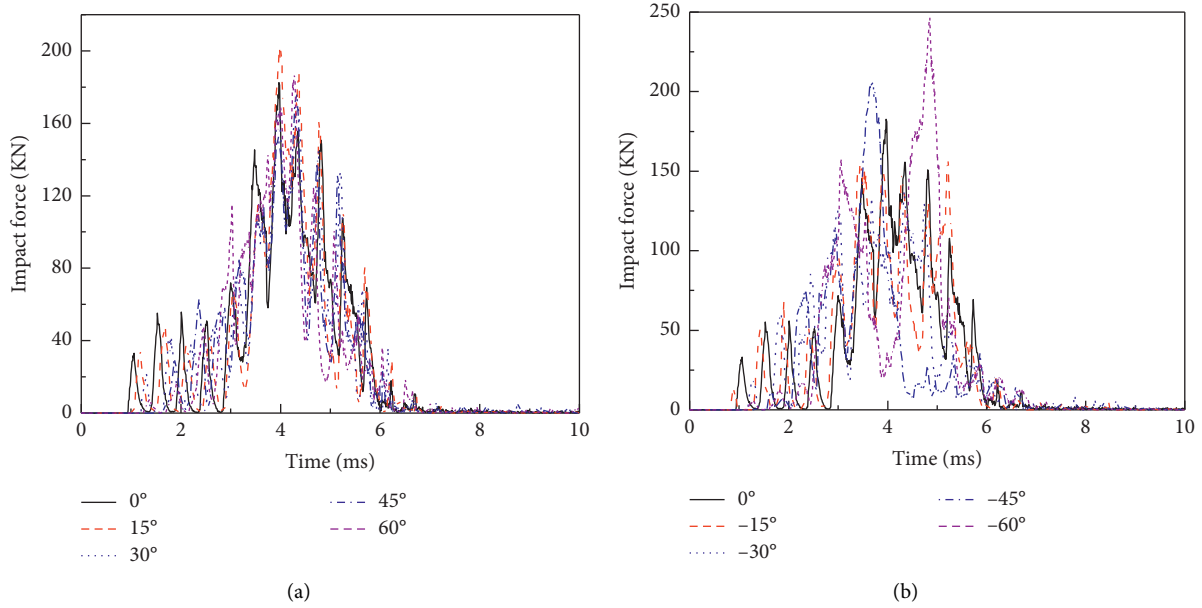


FIGURE 13: Yaw angle effect on impact loading history. (a) Positive yaw angle α . (b) Negative yaw angle α .

values of impact force derived from a real bird with yaw angle of -15° , -30° , -45° and -60° are 157.32 KN at 5.22 ms, 137.99 KN at 4.33 ms, 206.47 KN at 3.70 ms, and 246.66 KN at 4.85 ms, respectively. It can be concluded that a yaw angle of the bird would generate a variation in the impact loading history, and a negative yaw angle has a more significant effect on the impact loading history.

Figure 14 shows the effect of pitch angle on the time histories of impact loads. As shown in Figure 14(a), with respect to pitch angles of 15° , 30° , 45° , and 60° , the peak impact loads are 205.15 KN at 4.35 ms, 227.20 KN at 3.91 ms, 261.47 KN at 4.33 ms, and 269.40 KN at 3.92 ms, respectively. Obviously, the peak impact force increases as the positive pitch angle increases. As shown in Figure 14(b), peak values of impact loading obtained from a real bird model with pitch angles of -15° , -30° , -45° and -60° are 160.73 KN at 3.93 ms,

168.28 KN at 3.90 ms, 184.85 KN at 3.86 ms, and 224.06 KN at 3.85 ms, respectively. It is observed that a negative yaw angle would result in a variation in the impact loading history.

Figure 15 shows effective plastic strains of a single blade, which undergoes the most severe plastic deformation during impact. Obviously, with the change of the bird attitude angle, the blade undergoes more severe plastic deformation. Especially, as subjected to a real bird with a yaw angle of -45° or a pitch angle of -60° , the blade suffers more than twice the magnitude of effective plastic strain compared with a real bird oriented along the engine centerline. It can be concluded that attitude angles would increase the possibility of fan blade failure. Moreover, the yaw angle of bird ingestion has a significant effect on the location of the blade with the most severe plastic deformation.

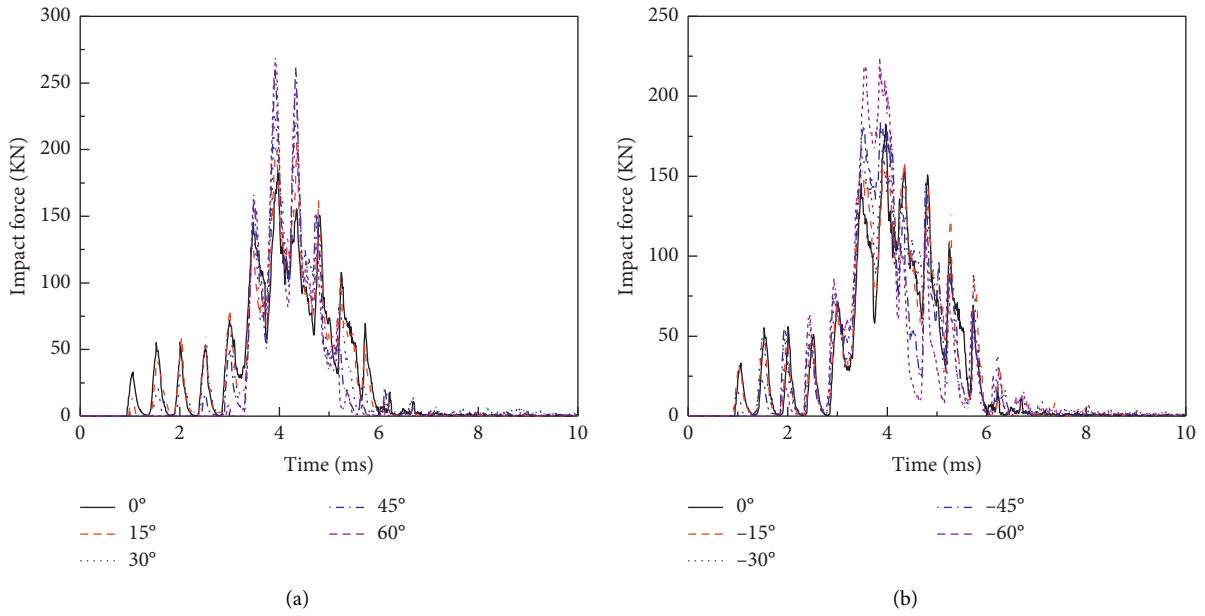


FIGURE 14: Pitch angle effect on impact loading history. (a) Positive pitch angle β . (b) Negative pitch angle β .

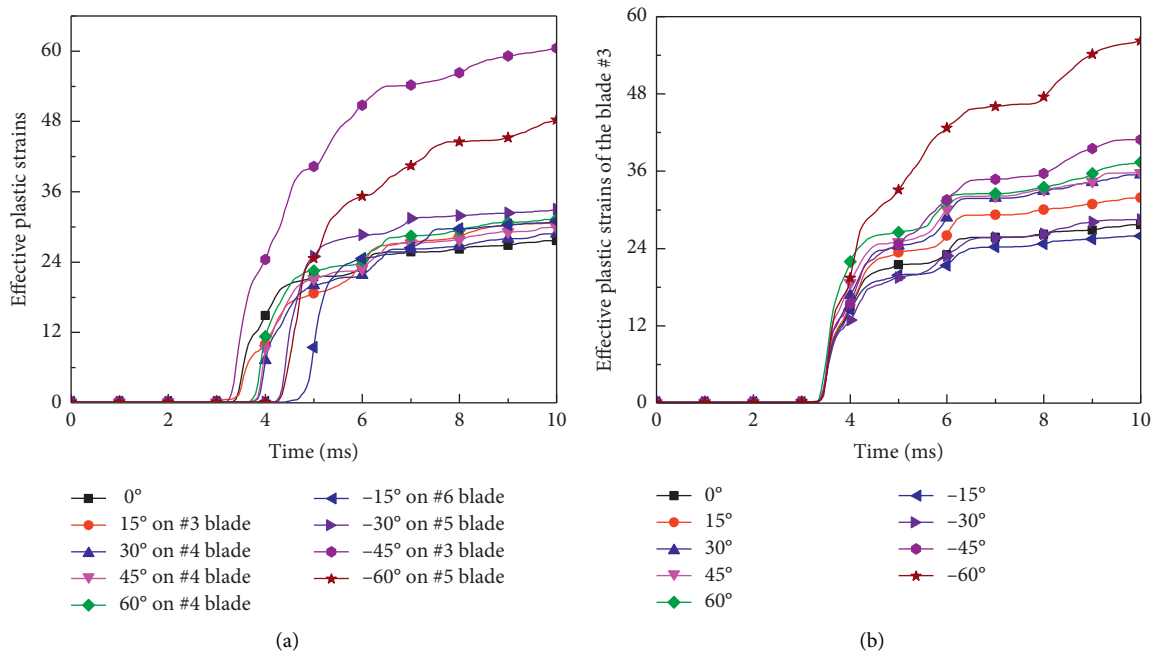


FIGURE 15: Attitude angle effect on blade damage. (a) Yaw angle α . (b) Pitch angle β .

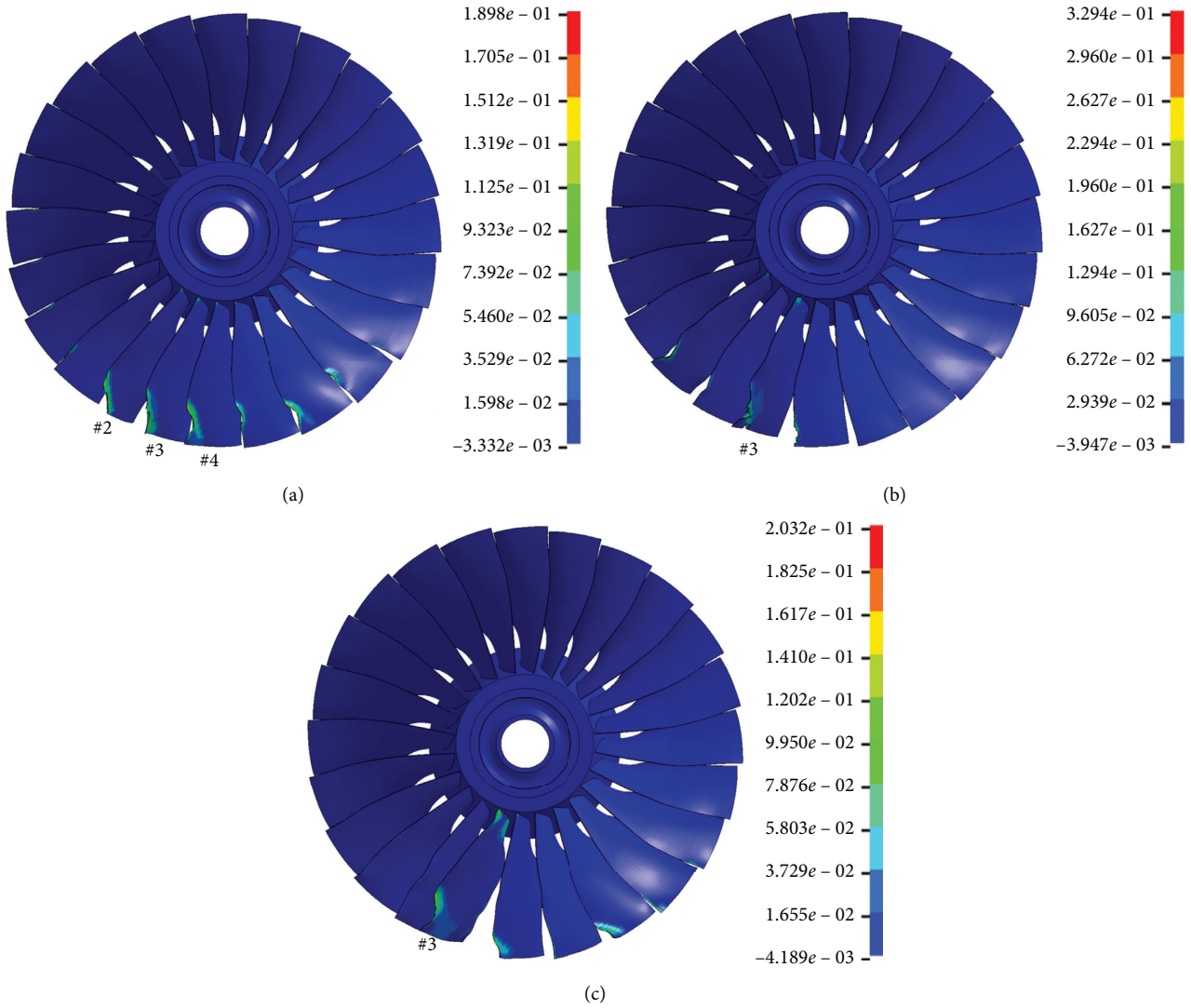


FIGURE 16: Effective plastic strain contours at time $t = 10$ ms. (a) 0° . (b) $\alpha = -45^\circ$. (c) $\beta = -60^\circ$.

Figure 16 shows the effective plastic strains of the fan assembly. It can be seen that a “cusp” plastic deformation occurs at the leading edge, and the maximum plastic strain occurred to the impact location. Compared with the plastic deformation of a fan assembly during a normal impact (see Figure 16(a)), subjected to a bird with an attitude angle of $\alpha = -45^\circ$ and $\beta = -60^\circ$, fan blade failure occurs at the impact location, and the plastic deformation at the root of the fan blade leading edge is more significant, which would increase the risk of a fan blade release event.

6. Conclusions

Considering a geometric shape similar to what has already been published as a real mallard [12], a real bird model was developed. Then, this paper discusses the effect of attitude angles of the realistic bird model on the soft impact damage of a full fan assembly. Besides, the stagnation pressure is represented as a criterion to calibrate the developed bird

model. Results show a good correlation with available test data [26].

In accordance with certification requirements, the most critical ingestion parameters for the new bird model were investigated. It is found that the real bird aimed at the fan blade height of 80% with an ingestion speed of 65 m/s produces the most severe damage to the full fan assembly.

Complying with the critical ingestion parameters of the bird, simulations of a full fan assembly subjected to a real bird with respect to various attitude angles reveal that both yaw and pitch angles of the bird have a significant effect on the impact loading history. The blade that undergoes the highest magnitude of effective plastic strain would be confronted with a more plastic deformation than that in the case of a normal impact (attitude angle 0°), even though the impact loading decreases. Especially, as subjected to a real bird with yaw angle of -45° or pitch angle of -60° , the blade suffers more than twice the magnitude of effective plastic strain compared with that of attitude angle 0° , and the plastic

deformation at the root of the leading edge is more significant, which would increase the risk of a fan blade release event.

Moreover, it is invaluable to ensure the safety of commercial airlines and engine manufacture by adopting the realistic bird model with the dangerous attitude angle in the certification of engine designs for resistance to bird strikes, which will provide sufficient resistance in actual bird-strike events. However, attitude angles of the bird were set up based on the assumption in this paper. If the bird attitude could be determined through experiments or control algorithms [38–42], it is of great significance to investigate the effect of bird attitude on soft impact damage, as well as the distribution of flocking bird.

Data Availability

All data included in this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was supported by the National Key Research and Development Program of China (Grant no. 2016YFB0201800) and the National Natural Science Foundation of China (Grant no. 11772192).

References

- [1] R. H. Mao, S. A. Meguid, and T. Y. Ng, “Transient three dimensional finite element analysis of a bird striking a fan blade,” *International Journal of Mechanics and Materials in Design*, vol. 4, no. 1, pp. 79–96, 2008.
- [2] S. Heimbs, “Computational methods for bird strike simulations: a review,” *Computers & Structures*, vol. 89, no. 23–24, pp. 2093–2112, 2011.
- [3] J. S. Wilbeck, *Impact Behavior of Low Strength Projectiles*, Air Force Materials Lab, Ohio, AFFDL-TR-75-5, 1978.
- [4] J. S. Wilbeck and J. L. Rand, “The development of a substitute bird model,” *Journal of Engineering for Power*, vol. 103, no. 4, pp. 725–730, 1981.
- [5] A. Airolidi and B. Cacchione, “Modelling of impact forces and pressures in Lagrangian bird strike analyses,” *International Journal of Impact Engineering*, vol. 32, no. 10, pp. 1651–1677, 2006.
- [6] M. A. Lavoie, A. Gakwaya, M. N. Ensan, D. G. Zimcik, and D. Nandlall, “Bird’s substitute test results and evaluation of available numerical methods,” *International Journal of Impact Engineering*, vol. 36, no. 10–11, pp. 1276–1287, 2009.
- [7] M. Guida, F. Marulo, M. Meo, and M. Riccio, “Evaluation and validation of multi-physics FE methods to simulate bird strike on a wing leading edge,” in *Proceedings of the 13th European Conference on Composite Materials (ECCM)*, Stockholm, Sweden, June 2008.
- [8] M. A. Lavoie, A. Gakwaya, M. N. Ensan, and D. G. Zimcik, “Review of existing numerical methods and validation procedure available for bird strike modelling,” *International Journal of Applied Mathematics and Computer Science*, vol. 2, no. 4, pp. 111–118, 2007.
- [9] M. A. Lavoie, A. Gakwaya, M. Nejad Ensan, and D. G. Zimcik, “Validation of available approaches for numerical bird strike modeling tools,” *International Review of Mechanical Engineering*, vol. 1, no. 4, pp. 380–389, 2007.
- [10] M. A. McCarthy, J. R. Xiao, C. T. McCarthy et al., “Modelling of bird strike on an aircraft wing leading edge made from fibre metal laminates-part 2: modelling of impact with SPH bird model,” *Applied Composite Materials*, vol. 11, no. 5, pp. 317–340, 2004.
- [11] L. Nizampatnam and W. Horn, “Investigation of equation of state models for predicting bird impact loads,” in *Proceedings of the 46th AIAA Aerospace Sciences Meeting and Exhibit*, pp. 7–10, 2013.
- [12] R. Hedayati, M. Sadighi, and M. Mohammadi-Aghdam, “On the difference of pressure readings from the numerical, experimental and theoretical results in different bird strike studies,” *Aerospace Science and Technology*, vol. 32, no. 1, pp. 260–266, 2014.
- [13] L. S. Nizampatnam, *Models and Methods for Bird Strike Load Predictions*, Wichita State University, Wichita, Kansas, 2007.
- [14] S. McCallum, H. Shoji, and H. Akiyama, “Development of an advanced multi-material bird-strike model using the smoothed particle Hydrodynamics method,” *International Journal of Crashworthiness*, vol. 18, no. 6, pp. 579–597, 2013.
- [15] R. Hedayati and S. Ziaei-Rad, “A new bird model and the effect of bird geometry in impacts from various orientations,” *Aerospace Science and Technology*, vol. 28, no. 1, pp. 9–20, 2013.
- [16] A. Serge, “Soft impacts on aerospace structures,” *Progress in Aerospace Sciences*, vol. 81, pp. 1–17, 2016.
- [17] J. Kou, F. Xu, S. Ji, and X. Zhang, “Influence of bird yaw/pitch orientation on bird-strike resistance of aircraft structures,” *Explosion and Shock Waves*, vol. 37, no. 5, pp. 937–944, 2017.
- [18] Z. Zhang, L. Li, and D. Zhang, “Effect of arbitrary yaw/pitch angle in bird strike numerical simulation using SPH method,” *Aerospace Science and Technology*, vol. 81, pp. 284–293, 2018.
- [19] D. Zhang and Q. Fei, “Effect of bird geometry and impact orientation in bird striking on a rotary jet-engine fan analysis using SPH method,” *Aerospace Science and Technology*, vol. 54, pp. 320–329, 2016.
- [20] R. H. Mao, S. A. Meguid, and T. Y. Ng, “Effects of incidence angle in bird strike on integrity of aero-engine fan blade,” *International Journal of Crashworthiness*, vol. 14, no. 4, pp. 295–308, 2009.
- [21] C. Yang, C. Chen, N. Wang et al., “Biologically inspired motion modeling and neural control for robot learning from demonstrations,” *IEEE Transactions on Cognitive and Developmental Systems*, vol. 11, no. 2, pp. 281–291, 2019.
- [22] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, “Robot learning system based on adaptive neural control and dynamic movement primitives,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 777–787, 2019.
- [23] C. Yang, D. Huang, W. He, and L. Cheng, “Neural control of robot manipulators with trajectory tracking constraints and input saturation,” *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–12, 2020.
- [24] H. Huang, T. Zhang, C. Yang, and C. L. P. Chen, “Motor learning and generalization using broad learning adaptive neural control,” *IEEE Transactions on Industrial Electronics*, vol. 67, no. 10, pp. 8608–8617, 2020.

- [25] H. Huang, C. Yang, and C. L. P. Chen, "Optimal robot-environment interaction under broad fuzzy neural adaptive control," *IEEE Transactions on Cybernetics*, pp. 1–12, 2020.
- [26] F. Allaey, G. Luyckx, W. Van Paeppegem, and J. Degrieck, "Numerical and experimental investigation of the shock and steady state pressures in the bird material during bird strike," *International Journal of Impact Engineering*, vol. 107, pp. 12–22, 2017.
- [27] M. Kim, A. Zammit, A. Siddens, and J. Bayandor, "An extensive crashworthiness methodology for advanced propulsion systems, part I: soft impact damage assessment of composite fan stage assemblies," in *Proceedings of the 49th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition*, Orlando, Florida, January 2011.
- [28] R. Hedayati and M. Sadighi, *Bird Strike: An Experimental, Theoretical and Numerical Investigation*, Woodhead Publishing in Mechanical Engineering, Elsevier, Amsterdam, Netherlands, 978-0-08-100093-9, 2016.
- [29] S. K. Sinha, K. E. Turner, and N. Jain, "Dynamic loading on turbofan blades due to bird-strike," *Journal of Engineering for Gas Turbines and Power*, vol. 133, no. 12, Article ID 122504, 2011.
- [30] Federal Aviation Administration, "Electronic code of federal Regulations," *Title 14: Aeronautics and Space*, National Archives and Records Administration, Washington, DC, USA, Chapter I, Subchapter C, Part 33-Airworthiness Standards: Aircraft Engines, Section 33.76, 2020.
- [31] Federal Aviation Administration, "Bird ingestion certification standards," *Advisory Circular*, Federal Aviation Administration, Washington, DC, USA, 33.76-1A, 2009.
- [32] R. A. Dolbeer, S. E. Wright, J. R. Weller, and M. J. Begier, *Wildlife Strikes to Civil Aircraft in the United States 1990-2017*, National Wildlife Strike Database, Washington, DC, USA, 2018.
- [33] C. J. Shepherd, G. J. Appleby-Thomas, P. J. Hazell, and D. F. Allsop, "The dynamic behaviour of ballistic gelatin," *AIP Conference Proceedings*, vol. 1195, pp. 1399–1402, 2009.
- [34] J. O. Hallquist, *LS-DYNA Theory Manual*, Livermore Software Technology Corporation, Livermore, CA, USA, 2007.
- [35] R. Vignjevic, M. Orłowski, T. De Vuyst, and J. C. Campbell, "A parametric study of bird strike on engine blades," *International Journal of Impact Engineering*, vol. 60, pp. 44–57, 2013.
- [36] G. R. Johnson and W. H. Cook, "Fracture characteristics of three metals subjected to various strains, strain rates, temperatures and pressures," *Engineering Fracture Mechanics*, vol. 21, no. 1, pp. 31–48, 1985.
- [37] D. Leseur, *Experimental Investigations of Material Models for Ti-6Al-4V and 2024-T3*, Lawrence Livermore National Lab., Livermore, CA, USA, UCRL-ID-134691, 1999.
- [38] C. Yang, Y. Jiang, W. He, J. Na, Z. Li, and B. Xu, "Adaptive parameter estimation and control design for robot manipulators with finite-time convergence," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 10, pp. 8112–8123, 2018.
- [39] C. Yang, G. Peng, L. Cheng, J. Na, and Z. Li, "Force sensorless admittance control for teleoperation of uncertain robot manipulator using neural networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, pp. 1–11, 2019.
- [40] G. Peng, C. Yang, W. He, and C. L. P. Chen, "Force sensorless admittance control with neural learning for robots with actuator saturation," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 4, pp. 3138–3148, 2020.
- [41] C. Yang, H. Wu, Z. Li, W. He, N. Wang, and C.-Y. Su, "Mind control of a robotic arm with visual fusion technology," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 3822–3830, 2018.
- [42] H. Lin, T. Zhang, Z. Chen, H. Song, and C. Yang, "Adaptive fuzzy Gaussian mixture models for shape approximation in robot grasping," *International Journal of Fuzzy Systems*, vol. 21, no. 4, pp. 1026–1037, 2019.

Research Article

Stochastic Parameter Identification Method for Driving Trajectory Simulation Processes Based on Mobile Edge Computing and Self-Organizing Feature Mapping

Jingfeng Yang,^{1,2,3} Zhiyong Luo ,³ Nanfeng Zhang,⁴ Jinchao Xiao,^{1,2} Honggang Wang,⁵ Shengpei Zhou,^{1,2} Xiaosong Liu,⁶ and Ming Li ^{7,8}

¹Shenyang Institute of Automation Chinese (Guangzhou) Academy of Sciences, Guangzhou 511458, China

²Shenyang Institute of Automation Chinese Academy of Sciences, Shenyang 110016, China

³School of Electronics and Communication Engineering, Sun Yat-Sen University, Guangzhou 510006, China

⁴Huangpu Customs District Technical Center, Guangzhou 510730, China

⁵School of Communications and Information Engineering, Xi'an University of Posts and Telecommunications, Xi'an 710061, China

⁶Guangdong Zhongke Zhenheng Information Technology Co., Ltd, Foshan 528225, China

⁷South China Agricultural University, Guangzhou 510642, China

⁸Yaz Technology Co., Ltd., Guangzhou 510630, China

Correspondence should be addressed to Zhiyong Luo; 13925121031@139.com and Ming Li; liming@yazcn.com

Received 16 August 2020; Revised 4 November 2020; Accepted 14 December 2020; Published 7 January 2021

Academic Editor: Rongxin Cui

Copyright © 2021 Jingfeng Yang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the rapid development of sensor technology for automated driving applications, the fusion, analysis, and application of multimodal data have become the main focus of different scenarios, especially in the development of mobile edge computing technology that provides more efficient algorithms for realizing the various application scenarios. In the present paper, the vehicle status and operation data were acquired by vehicle-borne and roadside units of electronic registration identification of motor vehicles. In addition, a motion model and an identification system for the single-vehicle lane-change process were established by mobile edge computing and self-organizing feature mapping. Two scenarios were modeled and tested: lane change with no vehicles in the target lane and lane change with vehicles in the target lane. It was found that the proposed method successfully identified the stochastic parameters in the process of driving trajectory simulation, and the standard deviation between simulation and the measured results obeyed a normal distribution. The proposed methods can provide significant practical information for improving the data processing efficiency in automated driving applications, for solving single-vehicle lane-change applications, and for promoting the formation of a closed loop from sensing to service.

1. Introduction

With the development of automated driving technologies, especially the onset of 5G technology, the demand for traffic data collection in the field of intelligent networking has increased significantly. Conventional driving environment detection technologies use road monitoring equipment with different traffic sensing technologies including video, GPS, geomagnetism, and radar. These devices are installed on roads by local public security agencies to ensure a safe and

smooth traffic flow, thus strengthening the law enforcement against traffic violations and the intelligent control of city-level transportation. However, these driving environment detection technologies cannot satisfy automated driving communication requirements of short time delay, high reliability, wide coverage, and vehicle-to-vehicle communication. Under this backdrop, the application of electronic registration identification of motor vehicles has been proposed and promoted on a national level through such measures as establishing national standards. Electronic

registration identification of motor vehicles have been promoted in a few cities and applied to several areas such as smart parking [1, 2], intelligent signal control [3, 4], intelligent connected vehicles [5], special vehicle traffic management and special vehicle identification monitoring [6], and traffic operation supervision and environmental protection restrictions [7]. However, as a means of law enforcement, the application of vehicle interactions is still lacking due to different technical problems; thus, it is difficult to achieve the required accuracy for vehicle driving trajectory identification. Hence, researchers have proposed different identification methods to solve this problem; for example, grid-based methods have been successfully used in multiobjective optimization algorithms [8–10], such as the positioning of driverless cars [11–13]. Grewe et al. [14] explored a series of MEC-enabled, high-quality, and reliable vehicle-borne services (such as electronic horizon, which assist vehicle movement), summarized the challenging problems encountered in the application of MEC technology to the network of vehicles, and proposed some potential solutions. Truong et al. [15] advocated the use of a combination of software-defined networking and MEC technology to address the problems of existing vehicle-borne self-organizing networks, such as inadequate coverage of communication range, unscalable network communication capacity, and a management that lacks intelligence and flexibility. They designed an edge computing network framework (dubbed FSDN) for the Internet of vehicles and MEC server that acts as a distributed zone controller to provide different local services, such as streaming of media content distribution and vehicle lane-change prompts, aimed at providing excellent decision-making capabilities. Real-time and accurate traffic information is the prerequisite for an improved urban traffic efficiency; Dikaiakos et al. [16] proposed the vehicle information transmission protocol (VITP) and implemented dynamic route planning using the forward traffic status and roadside points of interest obtained from query requests to downstream vehicles through the geographic routing protocol. Dombush and Joshi [17] studied the automatic discovery mechanism of traffic congestion and the distributed clustering of abnormal traffic flow and achieved the aggregation of perceived data through interactive sensing between vehicles. Zhang and Zhao [18] constructed a mathematical model to automatically collect, aggregate, process, and transmit traffic information and dynamically update the traffic coverage of the entire road network. Gramaglia et al. [19] used external data sources such as vehicle beacon messages and weather conditions to detect the degree of traffic congestion through complex event processing. Terroso-Saenz et al. [20] and Li et al. [21] detected traffic congestion from intervehicle collaboration where each vehicle used the collected beacon messages to estimate the surrounding traffic congestion using fuzzy logic and then corrected the individual estimation error through consultation. Bauza et al. [22] conducted traffic condition sensing through VANET collaboration and made short-term traffic condition forecasts using a linear least squares method. However, these studies have only solved the problems of cooperative traffic condition sensing and

neither consider implementation issues in automated driving scenario nor analyze important driving trajectories in vehicle-to-vehicle communication. In the present paper, based on electronic registration identification of motor vehicles and readers, a closed loop from perception to service was formed using mobile edge computing and self-organizing feature mapping to identify stochastic parameters for driving trajectory simulation processes and solve different application problem of lane-change scenarios in automated driving.

2. Driving Behavior, Speed, and Trajectory Calculation Based on Electronic Registration Identification of Motor Vehicles

The vehicle data collection method integrated with electronic registration identification of motor vehicles reads the vehicle information of the vehicle from the on-board unit with the electronic registration identification of motor vehicles reader installed on the gantry frame. The collected information is then uploaded to the comprehensive sensing base station on the test section of the road for edge computing. The overall deployment of the comprehensive sensing base station can complete data collection, verification, transmission, and processing in actual applications. The overall layout and actual installation are shown in Figure 1:

The main method of acquiring the driving speed of a vehicle with an electronic registration identification of motor vehicles reader uses the UHF radio frequency identification technology. The directional horizontally-polarized UHF antenna of the registration reader interacts with the unit installed on the front windshield of the vehicle and calculates the speed of the vehicle by measuring the time difference for the vehicle to pass through a fixed distance in the identification zone. Specifically, the positions of the vehicle at different times are calculated from RSSI values returned by the electronic registration identification of motor vehicles reader at different times, and the travel speed can then be calculated from the time differences. By analyzing RSSI values returned by the vehicle after entering the identification cross-section, it can be determined whether the location of the vehicle is at the boundary between the direct illumination zone and the blind zone or at the farthest point in the reflection zone; subsequently, the vehicle speed is calculated by the formula $V = S/(T_o - T_i)$.

3. Mobile Edge Computing and Self-Organizing Feature Mapping

3.1. Mobile Edge Computing. The fundamental concept of mobile edge computing is to move the cloud computing platform to the edge of the mobile access network in an attempt to deeply integrate a cellular network of conventional telecommunication with Internet services and also to reduce the end-to-end delay of mobile service delivery. It changes the state of a separate network in conventional wireless communication systems by exploiting the inherent capabilities of wireless networks and improves the user

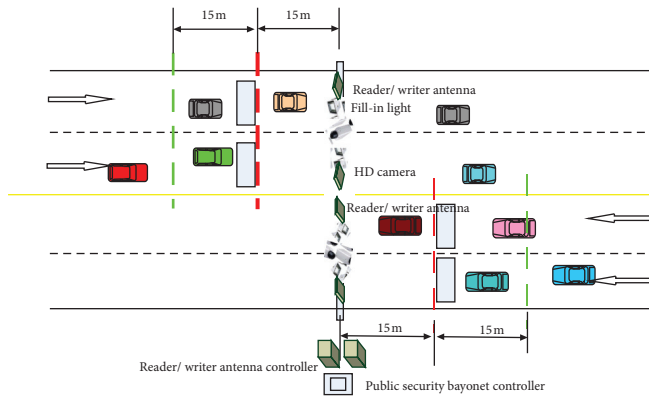


FIGURE 1: Comprehensive layout diagram and actual installation figure.



experience. By adding mobile edge computing platform network elements to a conventional wireless network, a business platform including contents, services, and applications, can be lowered to the edge of a mobile network to provide computing and data storage services to users. The basic characteristics of mobile edge computing include service localization, short-distance, low time-delay service delivery, user location awareness, and other network service capabilities. It can, therefore, bring new changes to the mode of operation for telecom operators and establish a new ecosystem of industrial chains and networks.

The edge computing architecture relies on the communication infrastructure and services provided by Edge Cloud Collaboration and LET/5G. The edge side mainly includes vehicle-borne edge computing units and RSU or MEC servers. Vehicle units depend on the cooperation of RSU or MEC servers. On the vehicle side, the control systems of intelligent networked vehicles are becoming increasingly complex with the use of AI and V2X applications. Load integration and simplified control systems are used to integrate different systems into HMI, and complex vehicle-borne information services including ADAD, IVI, digital instruments, and HUD are made to run on the same hardware platform through simulation technology. Moreover, load integration based on virtualization and the hardware abstraction layer HAL are more amenable to flexible business orchestration, deep learning model update, and software and firmware upgrade for a cloud-based vehicle driving system.

Due to several hardware constraints, the generation of a large amount of V2X data, and the imbalance of task processing, real-time scheduling is necessary for computing tasks. In the present work, a hybrid critical computing task scheduling technique was employed. The proposed method is linearly complex but is low cost and highly schedulable and can lower the time consumption of a system by reducing the number of repeated calculations in the passive delay priority updating. By establishing a more accurate upper bound of the peak period based on the quantitative relationship between the task number and the peak period, the spatial overhead of the system can be reduced and the scheduling performance can be improved.

3.2. Stochastic Parameter Identification Method Based on Self-Organizing Feature Mapping. According to the current practice of data optimization through neural networks, a self-organizing feature map (SOFM) was used in the current work to perform data optimization. Self-organizing feature map (SOFM) is a neural network with a clustering function [23]. Self-organizing mapping is a multidimensional scaling method, which can map any dimensional data in the input space to an output space of lower dimension. A self-organizing map (SOM) is an array of neurons at a regular location. Neurons can be placed on grid nodes in one, two, or higher dimensions, however usually in a one-dimensional or two-dimensional grid.

The operation of a SOFM network can be divided into two stages: training and work [24, 25]. In the training stage, samples from the training set are randomly fed into the network. For a specific input mode, the output layer generally has a neuron that produces the largest response to win. At the beginning of the training phase, it is uncertain which neuron in the output layer will respond most to the input mode. When the input mode changes to a different category, the winning neuron in the two-dimensional plane also changes. Neurons surrounding the winner neuron have a greater response due to the lateral mutual excitation. Therefore, weight vectors connected to the winner neuron and its surrounding neurons adjust in different degrees to the direction of the input vector. The degree of adjustment gradually diminishes as the distance from the winner neuron to its surrounding neurons increases. The network adjusts weights in a self-organizing manner by a large number of training samples. Finally, each neuron in the output layer becomes a sensitive neuron for a specific mode, and the corresponding inner weight vector becomes the center vector of each input mode. When the features of two input modes approach each other, neurons representing these two mode categories also approach each other, resulting in the formation of an ordered feature map reflecting the distribution of sample modes in the output layer.

After the completion of SOFM training, the specific relationship between each neuron in the output layer and each input mode category is completely determined, to use

the network as a mode classifier. At each input mode, the specific neuron in the output layer of the network generates the maximum response and automatically classifies the input. It should be noted that when the input mode of the network does not belong to any of the modes of the training set, SOFM can only assign it as belonging to the closest mode category. The classification process is performed by the following steps [24, 25].

In the first step, the winner neuron is found, the dot product of the input mode and the weight vector are calculated, and the winner neuron with the largest dot product is selected.

In the second step, the winning neighborhood is defined and the weight adjustment domain at time t centered around the winner neuron is determined. Generally, the initial neighborhood N is larger and gradually shrinks with the increase of the training time during the training process.

In the third step, the weights of all neurons in the winning neighborhood are adjusted by the following formula:

$$w_{i,j}(t+1) = w_{i,j}(t) + \eta(t, N) [x_i^p - w_{i,j}(t)], \quad (1)$$

$$i = 1, 2, \dots, n \quad j \in N_j \cdot (t),$$

where $\eta(t, N)$ is a function of the training time t and the topographical distance N between neuron j and the winner neuron j^* in the neighborhood.

Many functions can meet the above rules, for example, the following function can be constructed:

$$\eta(t, N) = \eta(t) e^{-N}. \quad (2)$$

The fourth step is the termination of the inspection. Unlike a BP network, the concept of output errors does not exist during the training of a SOFM network. As it is an unsupervised training, the training ends when the rate of learning $\eta(t)$ decays to zero or a certain preset positive small number. If the condition is not met, the process returns to step 1.

Kalman filter, based on the model of a state space, estimates the recursive relationship of an algorithm for a given process of filtering the state vector and constructs a measurement function (structural parameter or state) as an independent variable for an unknown quantity. It therefore derives an estimate of the unknown quantity from the measurement data. Kalman filter is widely applied to system identification. The conventional linear Kalman filter generally suffers from several shortcomings, such as low accuracy, poor stability, and slow response to target maneuver. In the application, the nonlinear motion equation of motion of the estimated dynamic system can be linearized and applied to the convergence of Kalman filter estimation. Furthermore, considering the nonlinear characteristics of the dynamic system near the reference trajectory and the comparison between strategically estimated algorithms, the real system can be described by a linearized equation.

Kalman filter can be divided into two parts: state prediction equation (state prediction) and state correction equation (observation update). The state prediction equation

is responsible to predict current state variables and estimate the error covariance in a timely manner to construct a priori estimate for the next time state. The state correction equation is responsible for giving feedback and combines the a priori estimate and the new observed variable to construct an improved a posteriori estimate.

Let the following equations describe an n -dimensional linear dynamic system and an m -dimensional linear observation system [26–28]:

$$\begin{aligned} x_{i+1} &= Ax_i + w_i, \\ z_i &= Hx_{i+1} + v_i, \end{aligned} \quad (3)$$

where x_{i+1} is an n -dimensional vector that represents the state of the system at the time instance $i+1$ and A is a $n \times n$ ordered matrix, which becomes the state transition matrix of the system, and reflects the state transition of the system from the i th sampling time to the $i+1$ st sampling time. w_i is an n -dimensional vector and represents the random interference acting on the system at the time instance $i+1$, and it is assumed that w_i is a Gaussian white noise sequence with a known zero mean and a covariance matrix Q_i , z_i is an m -dimensional observation vector, and H is a $m \times n$ ordered observation matrix and represents the transition from the state x_{i+1} to the observation vector z_i . For the m -dimensional observation noise v_i , it is also assumed that v_i is a Gaussian white noise sequence with a known zero mean and a covariance matrix R_i .

Therefore, the following recursive formula of the filter can be obtained:

$$\begin{aligned} K_i &= \hat{P}_i H^T (H \hat{P}_i H^T + R_i)^{-1}, \\ \hat{P}_i &= A P_{i-1} A^T + Q_i, \\ P_i &= (I - K_i H) \hat{P}_i, \\ \hat{x}_{i+1} &= x_{i+1} + K_i (z_i - H x_{i+1}), \end{aligned} \quad (4)$$

where Q_i is an $n \times n$ -ordered covariance matrix of the model noise w_i , R_i is the $m \times m$ -ordered covariance matrix of the observed noise v_i , K_i is the $n \times m$ -ordered gain matrix, \hat{x}_{i+1} is an n -dimensional vector and represents the estimated value after filtering at the time instance $i+1$, and P_i is the estimated error covariance matrix of order $n \times n$.

According to the calculation method described above, one can start with x_1 and a given P_0 and then recursively calculate the estimate of each time state by the known matrices Q_i , R_i , H , and A and the observation value z_i at the time i . If the linear system is stationary, then A and H are both constant matrices. If the model noise w_i and the observation noise v_i are stationary random sequences, then Q_i and R_i are constant matrices. Under such conditions, the constant gain discrete Kalman filter becomes asymptotically stable.

4. Tests and Results

In the present work, a single-vehicle changing lane was used as the main application scenario to establish a motion model and perform system identification. The single-

vehicle lane-change process was further divided into two scenarios: (i) lane change with no vehicle in the target lane and (ii) lane change with vehicles in the target lane. The trajectory of the lane-change execution phase began as the vehicle started to move from the initial lane to the target lane, and it proceeds to cross the lane line and reaches the target lane. In the entire process of crossing the lane line, any turning movement of the vehicle was considered as a failure of the execution of the lane-change operation. As the lane line information and the motion trajectory in the environment were acquired by vehicle-borne units and roadside units and calculated by edge computing, the distance between the vehicle and the lane line was calculated directly at any time.

The vehicle trajectory was expressed by the mathematical expression $\eta(t) = (X(t), Y(t), dX/dt(t), dY/dt(t))$ to include the horizontal and vertical positions and speed information of the vehicle in the observation geodetic coordinate system, where X, Y , and t represent, respectively, the horizontal coordinate, vertical coordinate, and sampling time. The schematic diagram of a lane-change process is shown in Figure 2.

Let t_0 be the running speed of the vehicle at time v_0 ; then, its lateral displacement and longitudinal position are, respectively, $L_h = v_0 t_0 \cos\theta$ and $L_z = v_0 t_0 \sin\theta$. Starting from the initial moment of the trajectory, a search could find two points with the smallest time intervals on both sides of the lane line. Now, setting them to t_{c1} and t_{c2} , respectively, the time for lane change can be obtained by the following formula:

$$\Delta_{t_{c2}-t_{c1}} = f_{t_{c1}}(X(t_{c1}), Y(t_{c1})) \cdot f_{t_{c2}}(X(t_{c2}), Y(t_{c2})). \quad (5)$$

According to the above definition, >0 indicates a left lane change and <0 indicates a right lane change.

Similarly, when there are other vehicles in the target lane, according to the principle of safe lane change, the speed and distance of the lane-change vehicle must exceed those of the vehicle in the target lane. Moreover, the lateral speed of the lane-change vehicle must be greater than that of the vehicle in the target lane, and the lateral distance difference of the lane-change vehicle must be greater than twice the length of the vehicle.

On this basis, lane-change experiments were carried out for two different scenarios where vehicles were present and absent in the target lane, and the lane-change states during a left lane change and a right lane change were compared. Figures 3 and 4, respectively, present the results of left lane change and right lane change with no vehicles in the target lane.

The curves in Figure 3 represent the simulated trajectories obtained through system recognition, for three left lane-change tests. The data labeled by o , x , and $*$ indicate the actual values of these three tests based on edge computing of results acquired and obtained from vehicle-borne unit and roadside units, and their corresponding relationships to simulated trajectory curves are presented in blue, black, and

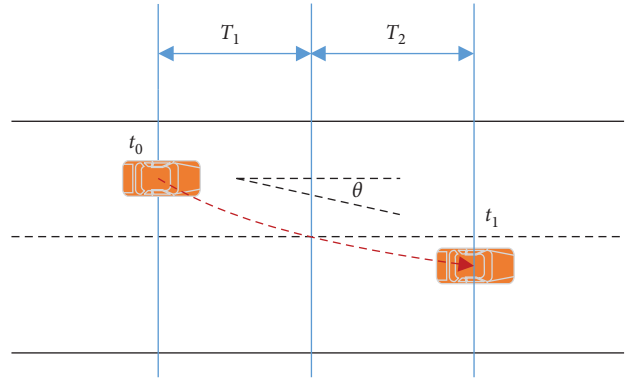


FIGURE 2: Schematic diagram of a lane-change process.

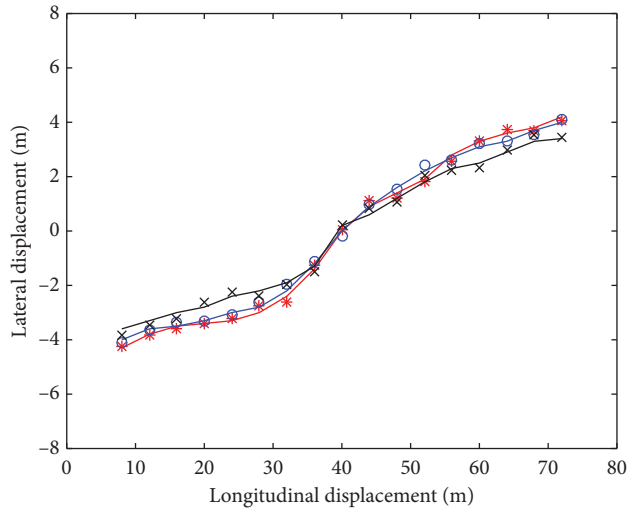


FIGURE 3: Actual test data and simulation data of left lane change.

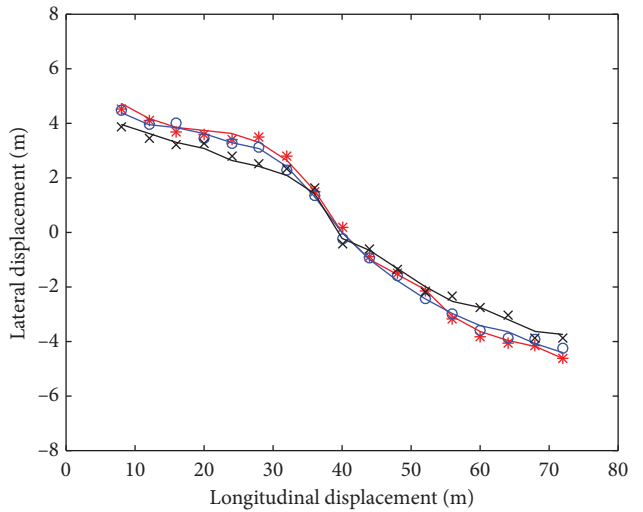


FIGURE 4: Actual test data and simulation data of right lane change.

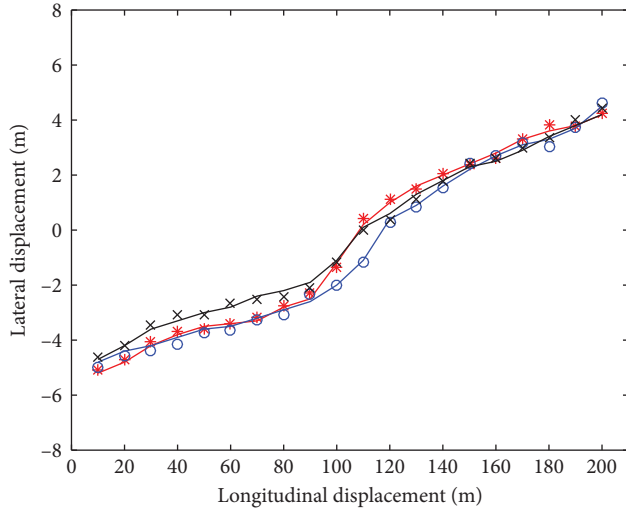


FIGURE 5: Actual test data and simulation data of left lane change.

red, respectively. Figure 4 displays the results for three right lane-change tests. Under standard operating conditions, the standard deviations between the actual value and the simulated trajectory for the three left lane-change operations were 0.78, 0.82, and 0.88, respectively, thus obeying a normal distribution. The lateral displacements in right lane-change operations were greater those in left lane-change operations, and it can be attributed to the seat location of the driver. The standard deviations for the three right lane-change tests were 0.92, 0.93, and 0.89 (slightly large than those of left lane-change operations), thereby still obeying a normal distribution despite slightly larger.

When there were other vehicles in the target lane, tests were carried out in the same way. Figures 5 and 6, respectively, present the results of left lane change and right lane change with vehicles in the target lane and with trajectory recognition based on actual data acquired from the vehicle-borne unit and roadside units. The results are compared with the actual data, as shown in Figure 5.

In comparison to the lane-change test results with no vehicles in the target lane, the longitudinal distance for completing the lane-change operation slightly increased in this scenario. The average longitudinal distance for completing a left lane change increased by 1.42 meters, whereas the average lateral distance increased significantly (nearly 93.89 meters). It happened because the obtained results were affected by an increased lane changing distance, the speed of vehicles in the target lane, and a longer distance to ensure safety.

Under standard operating conditions, the standard deviations between the actual value and the simulated trajectory for the three left lane-change tests were 1.12, 1.21, and 1.19, respectively, thus demonstrating a normal distribution. The lateral distances in right lane-change operations were greater than those in left lane-change operations, and it can be ascribed to the seat position of the driver. The standard deviations, for the three right lane-change tests were 1.36, 1.53, and 1.47, respectively (slightly larger than those of left lane-change operations), thereby still conforming to a normal distribution.

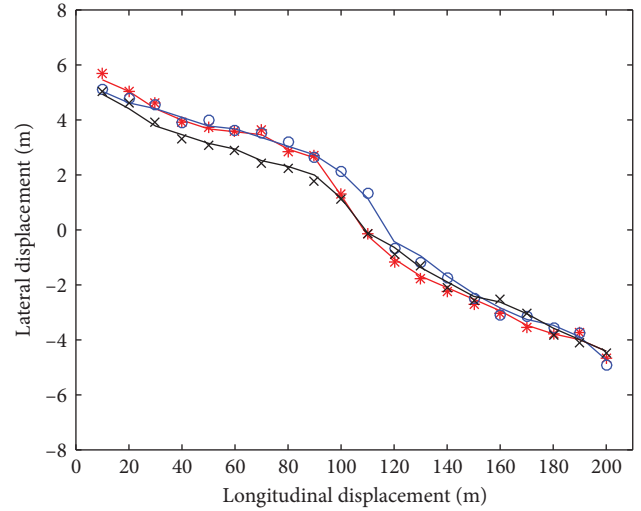


FIGURE 6: Actual test data and simulation data of right lane change.

5. Conclusion

The conventional automobile motion trajectory is mainly constrained by the kinematic and dynamic characteristics and real-time issues of a vehicle. In recent years, the research on smart car movement trajectories has paid more attention to the application of V2X communication among conventional automobiles. Transportation vehicles generally face a highly complex and stochastic driving environment. In order to strengthen the driving safety management, the present paper considered the single-vehicle lane-change process as a research topic. Based on electronic registration identification of motor vehicles and the associated reader/writer, mobile edge computing, and self-organizing feature mapping algorithms, a stochastic parameter recognition method was proposed for the driving trajectory simulation process, and its feasibility was verified through testing. A vehicle trajectory data acquisition method was first proposed based on electronic registration identification of motor vehicles, and the detailed description from the wiring and installation of the readers were then provided to the acquisition, calibration, and application of data. The lane-change process was mathematically modeled by mobile edge computing and self-organizing feature mapping, and the obtained simulation results were compared with actual test data. It was found that the standard deviation between the actual value of the lane-change operation and the simulated trajectory conformed to a normal distribution. Therefore, the proposed method can effectively improve the accuracy of intelligent vehicle trajectory planning and the driving safety during lane-change operation.

Data Availability

The original data used to support the findings of this study are restricted by the relevant law-enforcement departments in order to protect vehicle information privacy and law enforcement basis. Data are available from relevant law-enforcement departments for researchers who meet the criteria for access to confidential data.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Authors' Contributions

Jingfeng Yang conceptualized the study, developed methodology, carried out formal analysis, and wrote the original draft. Zhiyong Luo developed methodology and supervised the study. Nanfeng Zhang validated the study and carried out data curation. Jinchao Xiao carried out data curation, formal analysis, and visualization. Honggang Wang validated the study and contributed to resources. Shengpei Zhou wrote, reviewed, and edited the article. Xiaosong Liu contributed to software and resources. Ming Li investigated the study, contributed to funding acquisition, and administered project.

Acknowledgments

This research was funded by the National Key Research and Development Program (no. 2017YFD0700602), 2018 Industrial Internet Innovation and Development Project-Basic Standards and Experimental Verification of Industrial Internet Edge Computing, Key Research and Development Plan of Shaanxi Province (2018ZDXM-GY-041), and Foshan Entrepreneurship and Innovation Team Project (2017IT100032).

References

- [1] J. Yang, Z. Zhang, N. Zhang et al., "Vehicle text data compression and transmission method based on maximum entropy neural network and optimized Huffman encoding algorithms," *Complexity*, vol. 2019, Article ID 8616215, 9 pages, 2019.
- [2] H. Jiang, W. Fang, and J. Huang, "Design of parking management system based on the electronic registration identification," *China Science and Technology Information*, vol. 28, no. 23, pp. 78-79, 2016.
- [3] R. Wu and H. Yang, "Application and research of the electronic registration identification in traffic management field," *Police Technology*, vol. 33, no. 3, pp. 24-26, 2017.
- [4] Y. Tao, X. Yan, T. Wang et al., "Intelligent transportation system development strategy for future intelligent society," *Science & Technology Review*, vol. 37, no. 34, pp. 48-53, 2016.
- [5] W. Lang, S. Tian, Schwarzc et al., "A Review of application of driving simulation technology in R&D of vehicular intelligence technology," *Journal of Highway and Transportation Research and Development*, vol. 34, no. 12, pp. 140-150, 2017.
- [6] C. Zuo, Z. You, and X. Niu, "The application of the electronic registration identification in public security traffic management," *Transportation Enterprise Management*, vol. 32, no. 1, pp. 91-93, 2017.
- [7] Z. L. Sun, "The application and practice of the electronic registration identification," *China Information Security*, vol. 7, no. 10, pp. 68-70, 2016.
- [8] J. Knowles and D. Corne, "Properties of an adaptive archiving algorithm for storing nondominated vectors," *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 2, pp. 100-116, 2003.
- [9] K. Deb, M. Mohan, and S. Mishra, "Evaluating the ϵ -domination based multi-objective evolutionary algorithm for a quick computation of Pareto-optimal solutions," *Evolutionary Computation*, vol. 13, no. 4, pp. 501-525, 2005.
- [10] M. Laumanns, L. Thiele, K. Deb, and E. Zitzler, "Combining convergence and diversity in evolutionary multiobjective optimization," *Evolutionary Computation*, vol. 10, no. 3, pp. 263-282, 2002.
- [11] M. Wu, S.-L. Dai, and C. Yang, "Mixed reality enhanced user interactive path planning for omnidirectional mobile robot," *Applied Sciences*, vol. 10, no. 3, p. 1135, 2020.
- [12] J. Luo, Z. Lin, Y. Li, and C. Yang, "A teleoperation framework for mobile robots based on shared control," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, p. 377, 2020.
- [13] H. Kong, C. Yang, G. Li, and S.-L. Dai, "A sEMG-based shared control system with No-target obstacle avoidance for omnidirectional mobile robots," *IEEE Access*, vol. 8, no. 1, pp. 26030-26040, 2020.
- [14] D. Grewe, M. Wagner, M. Arumaithurai et al., "Information-centric mobile edge computing for connected vehicle environments: challenges and research directions," in *Proceedings of the Workshop on Mobile Edge Communications*, ACM, New York, NY, USA, August 2017.
- [15] N. B. Truong, G. M. Lee, and Y. Ghamri-Doudan, "Software defined networking-based vehicular ad-hoc network with fog computing," in *Proceedings of the 2015 IFIP/IEEE International Symposium on Integrated Network Management (IM)*, pp. 1202-1207, IEEE, Ottawa, Canada, May 2015.
- [16] M. Dikaiakos, A. Florides, T. Nadeem, and L. Iftode, "Location-aware services over vehicular ad-hoc networks using car-to-car communication," *IEEE Journal on Selected Areas in Communications*, vol. 25, no. 8, pp. 1590-1602, 2007.
- [17] S. Dombush and A. Joshi, "Street smart traffic: discovering and disseminating automobile congestion using VANETs," in *Proceedings of the 2007 IEEE 65th Vehicular Technology Conference-VTC2007-Spring*, April 2007.
- [18] Q. Zhang and J. H. Zhao, "A model for automatic collection and dynamic transmission of traffic information based on VANET," in *Proceedings of the 2012 15th International IEEE Conference on Intelligent Transportation Systems*, pp. 373-378, IEEE, Anchorage, AK, USA, September 2012.
- [19] M. Gramaglia, M. Calderon, and C. J. Bernados, "ABEONA monitored traffic: VANET-assisted cooperative traffic congestion forecasting," *IEEE Vehicular Technology Magazine*, vol. 9, no. 2, pp. 50-57, 2014.
- [20] F. Terroso-Saenz, M. Valdes-Vela, C. Sotomayor-Martinez, R. Toledo-Moreo, and A. F. Gomez-Skarmeta, "A cooperative approach to traffic congestion detection with complex event processing and VANET," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 2, pp. 914-929, 2012.
- [21] W. Li, C. Yang, Y. Jiang, X. Liu, and C. Su, "Motion planning for omni-directional wheeled mobile robot by potential field method," *Journal of Advanced Transportation*, vol. 2017, Article ID 4961383, 11 pages, 2017.
- [22] R. Bauza and J. Gozalvez, "Traffic congestion detection in large-scale scenarios using vehicle-to-vehicle communications," *Journal of Network and Computer Applications*, vol. 36, no. 5, pp. 1295-1307, 2013.
- [23] S. Alexander, "Cluster analysis for large scale gene expression studies," pp. 28-31, Alexander Sturn, Rockville, MD, USA, 2000, M.S. thesis.
- [24] C. Xu, Y. Yang, and W. Chen, "Application and analysis of self-organizing feature map," *Journal of ChangChun Teachers College (Natural Science)*, vol. 24, no. 4, pp. 55-58, 2005.

- [25] C. Li, L. Ning, and P. Shi, "The principle and application of self-organizing mapping network in the cluster of cultivated and use pressure in China," *Journal of Beijing Normal University(Natural Science)*, vol. 51, no. 5, pp. 543–547, 2006.
- [26] F. Haitao, "*The estimation method of parameter identification for aircraft stochastic control system*," M.S. thesis, Shenyang Ligong University, Shenyang, China, 2015.
- [27] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, "Robot learning system based on adaptive neural control and dynamic movement primitives," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 777–787, 2019.
- [28] H. Wang, S. Wang, J. Yao et al., "Effective anti-collision algorithms for RFID robots system," *Assembly Automation*, vol. 40, no. 1, pp. 55–64, 2019.

Research Article

A Gradient-Based Recurrent Neural Network for Visual Servoing of Robot Manipulators with Acceleration Command

Zhiguan Huang ¹, Zhengtai Xie ², Long Jin ^{1,2} and Yuhe Li ¹

¹Guangdong Provincial Engineering Technology Research Center for Sports Assistive Devices, Guangzhou Sport University, Guangzhou, China

²School of Information Science and Engineering, Lanzhou University, Lanzhou, China

Correspondence should be addressed to Long Jin; jinlongsysu@foxmail.com and Yuhe Li; 13922738963@139.com

Received 31 July 2020; Revised 11 November 2020; Accepted 15 December 2020; Published 30 December 2020

Academic Editor: Ning Wang

Copyright © 2020 Zhiguan Huang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Recent decades have witnessed the rapid evolution of robotic applications and their expansion into a variety of spheres with remarkable achievements. This article researches a crucial technique of robot manipulators referred to as visual servoing, which relies on the visual feedback to respond to the external information. In this regard, the visual servoing issue is tactfully transformed into a quadratic programming problem with equality and inequality constraints. Differing from the traditional methods, a gradient-based recurrent neural network (GRNN) for solving the visual servoing issue is newly proposed in this article in the light of the gradient descent method. Then, the stability proof is presented in theory with the pixel error convergent exponentially to zero. Specifically speaking, the proposed method is able to impel the manipulator to approach the desired static point while maintaining physical constraints considered. After that, the feasibility and superiority of the proposed GRNN are verified by simulative experiments. Significantly, the proposed visual servo method can be leveraged to medical robots and rehabilitation robots to further assist doctors in treating patients remotely.

1. Introduction

As one of the greatest human inventions in the 20th century, robot technology has undoubtedly made great progress in the past decades with brilliant research achievements [1–4]. After the birth, growth, and maturity of robots, they have become the indispensable core equipment in the manufacturing industry due to their high automation and efficiency. Especially as the rising star of the family of robots, redundant robots, which possess more degrees of freedom (DOFs) than the task requires, are capable of performing complicated tasks efficiently with the great property and versatility. In detail, the redundancy characteristic assists the redundant robots in fulfilling additional task demands, for example, repetitive motion planning [5], physical constraint avoidance [6], and manipulability optimization [7, 8]. In combination with medical technology, various medical robots have been developed and explored for patient

rehabilitation and surgical execution as an important application prospect. Relying on high reliability and flexibility, medical robots are able to perform complex medical tasks, thus reducing the burden on doctors and improving treatment. The learning and control ability of various robots is also valued and explored by many scholars [9–11]. A novel learning framework for the robot learning and generalizing human-like variable impedance skills is developed in [9] with great research and practical value. Further, some adaptive control methods are presented for estimating the unknown model of manipulator dynamic, which achieves great parameter estimation and tracking effects [10, 11].

In current years, the kinematic control of redundant robots has become a research hotspot, thus drawing the attention of abundant scholars to expand their applications [12–16]. Zhang and Zhang present a minimum-velocity-norm (MVN) scheme for redundancy resolution of the redundant manipulators, which retains the robot joints within safe bounds [17]. A

modified neural network approach in [18] is well designed for precise control of the robot manipulator, which can eliminate the error accumulation with accurate results. Moreover, the authors in [19] research an ingenious transformation method to deal with the acceleration limitation problem from the velocity level, and the experimental results illustrate the superiority of the method. It is deserved to notice that the above investigations [15–19] all transform the kinematic control issue of redundant robots into quadratic programming and then exploit the Karush–Kuhn–Tucker (KKT) conditions [20] or Lagrange multiplier method to solve the optimization schemes. In addition, the mentioned schemes in [15–19] are all velocity-level solutions such that they cannot interfere directly with the acceleration level.

With the continuous development of sensors and Internet of Things technology, robot applications have become very rich owing to information acquisition and processing. The sensor can transmit the external information directly to the control center of the robot and give appropriate feedback to the information through specific intelligent algorithms. As a greatly important robot application, the vision servoing technology drives the robot to accurately feedback the external vision in real time through the visual information collected by the vision sensor [21–23]. This technology is already being used in industrial production and robotic surgery [24, 25]. However, it is worth pointing out that the existing techniques [26–28] for solving the vision servoing problem often rely on the implementation of the pseudoinverse method to converge errors, which has achieved great results in both the acceleration-level schemes and the velocity-level schemes. By means of proportional-differential control, acceleration command for the visual servoing control is generated with excellent stability [26]. Moreover, an effective method to detect and compensate for faults in visual servoing systems is presented in literature [27], which is verified by simulation and experimental results. Based on the pseudoinverse operation of the Jacobian matrix, the robotic ball catching task is implemented [28]. This method takes advantage of the eye-in-hand construction to establish the motion capture system for locating fast-moving objects. However, a large number of investigations do not consider the existence of joint constraints and have potential for damage to the robot manipulators [21–24, 26–28]. Due to the physical limitations of the robot motor and robot structure, the control signals need to be kept within a reasonable range to maintain the normal operation of the robot manipulators. To this end, this paper formulates the visual servoing problem as a quadratic programming scheme with equality and inequality constraints in consideration of physical constraints.

The rise of intelligent algorithms in recent years has solved many difficult problems in electronic and engineering fields [29–31]. Numerous intelligent algorithms have been designed for powerful performance, such as noise suppression [32], simplified computation [33, 34], and predictive learning [35, 36]. Among the intelligent algorithms to solve the visual servoing of the manipulator, the neural network method stands out due to its fast parallel processing performance and learning ability [37–41]. In [42], a

recurrent neural network is constructed for the visual servoing issue to force the feature point of the manipulator to approach the designed target point. Then, the extended research [43] eliminates the pseudoinversion operation and equips the neural network with powerful robustness. In addition, as a common optimization method, the gradient descent method has made some progress in the design of robot control algorithms in recent years [44, 45]. It can be used to accurately locate and control the robot by minimizing the position error [46]. Based on the above research, we establish the visual servoing issue based on acceleration commands and transform it into a quadratic programming scheme solved by the neural network method. Besides, the contributions of this paper are summarized below:

- (1) The proposed method regards the visual servoing problem as a constrained quadratic programming scheme with acceleration command and meanwhile considers the joint constraints to ensure the safety of the manipulator
- (2) This paper proposes a gradient-based recurrent neural network (GRNN) for dealing with the research on the robot visual servoing via the gradient descent method and exploiting compensation item
- (3) The simulation example and illustrative experiment illustrate the feasibility and superiority of the proposed method

The remainder of this paper is summarized as follows. Section 2 covers the preliminaries and the visual servoing kinematics. In Section 3, the visual servoing problem is transformed into a constrained quadratic programming scheme at the acceleration level with the corresponding GRNN deduced. The theoretical analyses of the proposed method are presented by using the Lyapunov method in Section 4. Section 5 carries out a simulation example to demonstrate the feasibility of the proposed method. In the end, we summarily conclude the whole paper in Section 7.

2. Preliminaries

In this section, the visual servoing kinematics is introduced, which records the conversion relationship between the joint space and the image space.

Primarily, in consideration of an eye-in-hand vision system [28], i.e., an n -DOF manipulator with a camera attached to the end effector, the forward kinematics of the manipulator is given as follows:

$$\mathcal{P}(\vartheta(t)) = p(t), \quad (1)$$

where $\mathcal{P}(\cdot)$ describes the transformation relationship between the joint space and Cartesian space; $\vartheta(t) \in \mathbb{R}^n$ represents the joint angle of the manipulator; and $p(t) \in \mathbb{R}^m$ denotes the Cartesian coordinates of the end effector. The investigation of visual servoing issue always takes both the position and posture of the end effector, and thus $p(t)$ is set as a six-dimensional vector hereinafter ($m = 6$). Taking the derivative of time with respect to formula (1) leads to

$$J(\vartheta(t))\dot{\vartheta}(t) = \dot{p}(t), \quad (2)$$

where $J(\vartheta(t)) \in \mathbb{R}^{m \times n}$ stands for the robot Jacobian matrix, which is determined by the manipulator structure; $\vartheta(t)$ signifies the joint velocity of the manipulator; and $\dot{p}(t)$ is the end effector velocity containing angular velocity and translational velocity. In addition, the physical constraints, involving joint velocity $\dot{\vartheta}$ and joint acceleration $\ddot{\vartheta}$, to maintain the safe operation of the manipulator system are provided as below:

$$\dot{\vartheta}^- \leq \dot{\vartheta} \leq \dot{\vartheta}^+, \quad (3a)$$

$$\ddot{\vartheta}^- \leq \ddot{\vartheta} \leq \ddot{\vartheta}^+, \quad (3b)$$

with $\dot{\vartheta}^-$ and $\dot{\vartheta}^+$ being the upper and lower bounds of joint velocity and $\ddot{\vartheta}^-$ and $\ddot{\vartheta}^+$ denoting the upper and lower bounds of joint acceleration. As for the camera frame and image frame, the corresponding relationship is deduced by means of similar triangle and given as follows [27, 42]:

$$\begin{bmatrix} a_i \\ b_i \end{bmatrix} = \frac{l}{d} \begin{bmatrix} a_c \\ b_c \end{bmatrix}, \quad (4)$$

of which $[a_i, b_i]^T$ is a point coordinate in the image frame with the superscript T denoting the transpose of a matrix or a vector; $[a_c, b_c, d]^T$ stands for the coordinate in the camera frame; and l denotes the focal length of the camera. Besides, in the image frame, point coordinates can be converted to pixel coordinates $v = [u, v]^T$ by the following formula [43]:

$$u = u_p + \kappa_a a_i, \quad (5a)$$

$$v = v_p + \kappa_b b_i, \quad (5b)$$

where $[u_p, v_p]^T$ stands for the designed original point and κ_a and κ_b are the pixel standard size. Furthermore, the relationship between the camera velocity, i.e., the end effector velocity \dot{p} , and pixel coordinate velocity \dot{v} can be introduced as

$$H(v, d)\dot{p} = \dot{v}, \quad (6)$$

where $H(v, d) \in \mathbb{R}^{2 \times 6}$ denotes the image Jacobian matrix [47, 48] with its expression being

$$H(v, d) = M \begin{bmatrix} -\frac{l}{d} & 0 & \frac{la_i}{d} & \frac{a_i b_i}{l} & -\frac{a_i^2 + l^2}{l} & b_i \\ 0 & -\frac{l}{d} & \frac{lb_i}{d} & -\frac{b_i^2 + l^2}{l} & -\frac{a_i b_i}{l} & -a_i \end{bmatrix}, \quad (7)$$

with

$$\begin{aligned} a_i &= \frac{u - u_p}{\kappa_a}, \\ b_i &= \frac{v - v_p}{\kappa_b}, \end{aligned} \quad (8)$$

$$M = \begin{bmatrix} \kappa_a & 0 \\ 0 & \kappa_b \end{bmatrix}.$$

Based on the above instructions, especially formula (2) and formula (6), it can be readily obtained that $H(v, d)J\dot{\vartheta} = \dot{v}$, which involves the relationship between the joint space and the image space. To simplify the presentation, one designs

$$\mathcal{F}\dot{\vartheta} = \dot{v} \text{ with } \mathcal{F} = H(v, d)J. \quad (9)$$

Furthermore, the kinematic relationship at the acceleration level is derived by taking time derivative as

$$\ddot{v} = \mathcal{F}\ddot{\vartheta} + \dot{\mathcal{F}}\dot{\vartheta}, \quad (10)$$

where \ddot{v} represents the acceleration of feature point in the image frame and $\dot{\mathcal{F}}$ denotes the time derivative of \mathcal{F} .

3. Acceleration-Level IBVS Scheme and Its Solution

The robot vision servoing controls the robot manipulator to interact with circumstances according to the visual information. This issue can be simplified to find the static point in the image frame by feeding back the image information. To this end, we turn this visual servoing problem into a constrained quadratic programming problem and design a neural network-based solver.

3.1. Quadratic Programming Scheme with Constraints. Above all, the visual servoing problem is formulated at the acceleration level into the following quadratic programming scheme:

$$\text{minimize } \frac{1}{2} \ddot{\vartheta}^T \ddot{\vartheta}, \quad (11)$$

$$\text{subject to } \ddot{v} = \mathcal{F}\ddot{\vartheta} + \dot{\mathcal{F}}\dot{\vartheta}, \quad (12)$$

$$v = v^*, \quad (13)$$

$$\ddot{\vartheta} \in \rho, \quad (14)$$

where v^* denotes the desired feature point, which is a designed constant vector and, $\rho = \{\ddot{\vartheta} \in \mathbb{R}^n, \rho^- \leq \ddot{\vartheta} \leq \rho^+\}$ is an inequality constraint corresponding to the physical limit (3) with ρ^- and ρ^+ devised as

$$\begin{aligned}\rho_i^- &= \max\{\alpha(\dot{\vartheta}_i^- - \dot{\vartheta}_i), \ddot{\vartheta}_i^-\}, \\ \rho_i^+ &= \min\{\alpha(\dot{\vartheta}_i^+ - \dot{\vartheta}_i), \ddot{\vartheta}_i^+\},\end{aligned}\quad (15)$$

where $\alpha > 0$ stands for the design parameter. Via (15), the physical constraints of joint acceleration and joint velocity could be considered and controlled within bounds simultaneously [15]. In this regard, take the upper limit of the physical constraint $\rho_i^+ = \min\{\alpha(\dot{\vartheta}_i^+ - \dot{\vartheta}_i), \ddot{\vartheta}_i^+\}$ as an example. For the joint velocity, when the joint velocity $\dot{\vartheta}_i$ approaches the upper bound of velocity-level joint constraint $\dot{\vartheta}_i^+$, $\alpha(\dot{\vartheta}_i^+ - \dot{\vartheta}_i)$ gets small and even close to zero. Afterwards, ρ_i^+ becomes tiny or even zero, so that the joint velocity stops growing and stays in joint constraints. Simultaneously, the upper bound of acceleration-level joint constraint $\ddot{\vartheta}_i^+$ is activated to realize acceleration-level joint constraint. Similarly, ρ_i^- is able to realize the velocity-level joint constraint and the acceleration-level joint constraint simultaneously.

3.2. Neural Network Solution. Differing from the traditional method to deal with equality constraints and inequality constraints, the gradient descent method [49] is exploited to derive the solution to the quadratic programming scheme (11)–(14). Design an error function $e = v - v^*$ to start the derivation. Utilizing neural dynamic formula $\dot{e} = -\delta e$ ($\delta > 0$) [50] and $\dot{v}^* = \ddot{v}^* = 0$, one can get

$$\dot{v} = -\delta(v - v^*), \quad (16)$$

which can be arranged and rewritten into the form of two norms as follows:

$$e = \frac{\|\dot{v} + \delta(v - v^*)\|_2^2}{2}. \quad (17)$$

Given the gradient descent formula [51],

$$\ddot{\vartheta} = -\nabla e = -\gamma \frac{\partial(e)}{\partial \dot{\vartheta}}, \quad (18)$$

with $\gamma > 0$, it would be readily deduced that

$$\ddot{\vartheta} = \gamma \mathcal{F}^T(-\dot{v} - \delta(v - v^*)). \quad (19)$$

Then, a compensation item ω is presented to make up for the lagging error in equation (19) as below:

$$\ddot{\vartheta} = \gamma \mathcal{F}^T(-\dot{v} - \delta(v - v^*)) + \omega. \quad (20)$$

Via deliberating the final desired stable state, i.e., $v - v^* = \dot{v} = 0$, one can simply get the expression of ω referring to the derivation below. Multiplying both sides of equation (20) by \mathcal{F} one gains

$$\mathcal{F}\ddot{\vartheta} = \gamma \mathcal{F}\mathcal{F}^T(-\dot{v} - \delta(v - v^*)) + \mathcal{F}\omega. \quad (21)$$

Set $v - v^* = \dot{v} = 0$, and it can be obtained that

$$\mathcal{F}\ddot{\vartheta} = \mathcal{F}\omega. \quad (22)$$

Then, taking the time derivative of $\dot{v} = \mathcal{F}\dot{\vartheta} = 0$ as

$$\mathcal{F}\ddot{\vartheta} = -\dot{\mathcal{F}}\dot{\vartheta}. \quad (23)$$

Comparing the two formulas above, one has

$$\mathcal{F}\omega = -\dot{\mathcal{F}}\dot{\vartheta}. \quad (24)$$

Hence, it can be easily got that

$$\omega = -\mathcal{F}^\dagger \dot{\mathcal{F}}\dot{\vartheta}, \quad (25)$$

with superscript † being the pseudoinverse operator of a matrix and $J^\dagger = J^T(JJ^T)^{-1}$. Consequently, the GRNN solver is structured for solving the quadratic programming scheme (11)–(14) as follows:

$$\ddot{\vartheta} = \mathcal{F}_\rho(\gamma \mathcal{F}^T(-\dot{v} - \delta(v - v^*))) - \mathcal{F}^\dagger \dot{\mathcal{F}}\dot{\vartheta}, \quad (26)$$

where $\mathcal{F}_\rho(x) = \arg \min_{y \in \rho} \|y - x\|$ can be regarded as a bounded activation function and the usage of $\arg \min$ can be referred to [52, 53], which is equivalent to the inequality constraint (14). As Figure 1 depicts, visual servoing scheme (11)–(14) aided with GRNN solver (26) integrates the robot frame and image frame and can be regarded as a restricted online acceleration controller. For GRNN (26) and scheme (11)–(14), the following corresponding relation is given. Owing to the derivative process that GRNN (26) originates from the error function (17), the gradient descent formula is designed to reduce the image error, thus ultimately achieving equality constraint (13). In the next place, the output control command is established at the acceleration level, which corresponds to the acceleration-level kinematics formula (12). Note that compensation item ω is the pseudoinverse solution of the system function in a stable state, i.e., the minimization of joint acceleration, which is equivalent to minimizing objective function (11). As for joint constraint (14), introducing $\mathcal{F}_\rho(\cdot)$ is able to impose restrictions on joint velocity and joint acceleration. In short, the proposed GRNN solver (26) corresponds to the quadratic programming scheme (11)–(14).

Remarks. Compared with the existing visual servo technologies, the innovations of this paper are worth emphasizing as follows: regarding the scheme (11)–(14) construction level, most of the previous strategies on visual servoing are controlled at the joint velocity level, few of which are controlled and driven by joint acceleration. In addition, none of the existing acceleration-level visual servo schemes takes joint limits into account, which is considered in the quadratic programming scheme (11)–(14). From the perspective of the intelligent algorithm, a majority of the existing techniques apply the pseudoinverse method to directly deal with the errors, which incurs additional computational overhead. However, GRNN (26) is deduced according to the gradient descent method and compensation term, which provides a novel approach to dealing with the visual servoing problem.

4. Stability Proof

In this section, the stability proof is provided to prove the feasibility and effectiveness of the proposed method (26) to

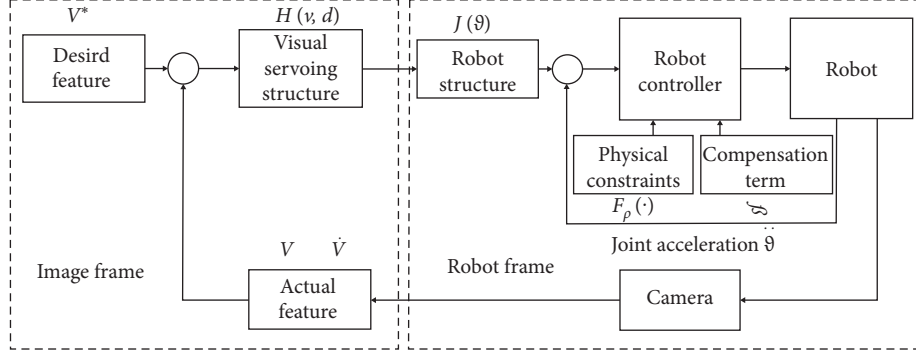


FIGURE 1: Control flowchart of the visual servoing scheme (11)–(14).

dispose of the visual servoing issue. The relevant theorem is given as follows.

Theorem 1. *The error $\varepsilon = v - v^*$ synthesized by GRNN (26) can approach zero globally, provided that $-\mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta} \in \rho$.*

Proof. Declare that the setting of precondition $-\mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta} \in \rho$ has two core functions. The first is to determine the minimum joint constraints, thus ensuring the safe operation of the manipulator. It is easy to image that forcing the joint to remain within the constraints may lead to the increase of error as reported in [54]. The second point is worth mentioning that precondition $-\mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta} \in \rho$ is of necessities for the proper derivation of the theorem. According to (10), one can get

$$\mathcal{F} \ddot{\vartheta} + \dot{\mathcal{F}} \dot{\vartheta} = \mathcal{F} \mathcal{F}_\rho (\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}) + \dot{\mathcal{F}} \dot{\vartheta}. \quad (27)$$

In the light of $\varepsilon = v - v^*$, $\dot{\varepsilon} = \dot{v}$, and $\ddot{\varepsilon} = \ddot{v}$, equation (27) can be rearranged as

$$\ddot{\varepsilon} = \mathcal{F} (\mathcal{F}_\rho (\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}) + \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}). \quad (28)$$

Let $V = \dot{\varepsilon}^T \dot{\varepsilon} / 2$ stand for a Lyapunov candidate. Therefore, calculating its time derivative $\dot{V} = \dot{\varepsilon}^T \dot{\varepsilon}$ results in

$$\begin{aligned} \dot{V} &= \dot{\varepsilon}^T \mathcal{F} (\mathcal{F}_\rho (\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}) + \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}) \\ &= -\frac{1}{\gamma} (\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta} + \delta \mathcal{F}^T \mathcal{E} + \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}) \\ &\quad \times (\mathcal{F}_\rho (\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}) + \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}). \end{aligned} \quad (29)$$

Consider the inequality relation $\|\mathcal{F}_\rho(x) - x\|^2 \leq \|x - y\|^2$, $\forall y \in \rho$. We simply devise $x = \gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}$ and $y = -\gamma \delta \mathcal{F}^T \mathcal{E} - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}$ and get

$$\begin{aligned} &\|\mathcal{F}_\rho (\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}) - \gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) + \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}\|^2 \\ &\leq \|\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta} + \gamma \delta \mathcal{F}^T \mathcal{E} + \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}\|^2. \end{aligned} \quad (30)$$

Expanding the left side of the above equation generates

$$\begin{aligned} &\|\mathcal{F}_\rho (\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}) + \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta} + \gamma \delta \mathcal{F}^T \mathcal{E}\|^2 \\ &\quad + \|\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta} + \delta \mathcal{F}^T \mathcal{E} + \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}\|^2 \\ &\quad - 2(\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta} + \delta \mathcal{F}^T \mathcal{E} + \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta})^T \\ &\quad \times (\mathcal{F}_\rho (\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}) + \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta} + \gamma \delta \mathcal{F}^T \mathcal{E}). \end{aligned} \quad (31)$$

Observe the two formulas above, and it can be easily gained that

$$\begin{aligned} &\|\mathcal{F}_\rho (\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}) + \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta} + \gamma \delta \mathcal{F}^T \mathcal{E}\|^2 \\ &\leq 2(\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta} + \delta \mathcal{F}^T \mathcal{E} + \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta})^T \\ &\quad (\mathcal{F}_\rho (\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}) + \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta} + \gamma \delta \mathcal{F}^T \mathcal{E}). \end{aligned} \quad (32)$$

Substituting equation (29) into equation (32) deduces

$$\begin{aligned} \dot{V} - \gamma \delta \dot{\varepsilon}^T \mathcal{F} \mathcal{F}^T \mathcal{E} &\leq \\ -\frac{1}{2\gamma} \|\mathcal{F}_\rho (\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}) + \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta} + \gamma \delta \mathcal{F}^T \mathcal{E}\|^2 &\leq 0. \end{aligned} \quad (33)$$

Evidently, one has

$$\dot{V} \leq \gamma \delta \dot{\varepsilon}^T \mathcal{F} \mathcal{F}^T \mathcal{E}. \quad (34)$$

Recalling the neural dynamic formula $\dot{\varepsilon} = -\delta \mathcal{E}$, it is evident that

$$\gamma \delta \dot{\varepsilon}^T \mathcal{F} \mathcal{F}^T \mathcal{E} = -\gamma \delta^2 \dot{\varepsilon}^T \mathcal{F} \mathcal{F}^T \mathcal{E} \leq -\gamma \delta^2 \sigma \dot{\varepsilon}^T \mathcal{E} \leq 0, \quad (35)$$

with design parameter $\gamma > 0$, $\delta > 0$, and $\sigma > 0$ denoting the minimum eigenvalue of positive definite matrix $\mathcal{F} \mathcal{F}^T$. Therefore, it can be naturally concluded that $\dot{\varepsilon}$ is of great convergence with $\dot{V} \leq 0$. Referring to the Lasalle invariance principle [55], we let $\dot{V} = 0$ to derive the stable state and get the following two conditions:

$$\dot{\varepsilon} = 0 \text{ or } \dot{\varepsilon} = \mathcal{F}_\rho (\gamma \mathcal{F}^T (-\dot{\varepsilon} - \delta \mathcal{E}) - \mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}) = -\mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta}. \quad (36)$$

Given that $-\mathcal{F}^\dagger \dot{\mathcal{F}} \dot{\vartheta} \in \rho$, the solutions to the above two conditions can be gained:

$$\dot{\varepsilon} = \varepsilon = 0. \quad (37)$$

In this regard, a conclusion can be readily drawn that ε is convergent to zero globally. The proof is thus completed (Figure 2). \square

5. Simulation Example

This section provides a simulation example to demonstrate the performance of GRNN (26) when confronted with the robot visual servoing issue. Specifically speaking, the PUMA 560 manipulator (6-DOF) is modeled with a camera attached to its end effector to track the desired static point in the image frame. In addition, the structure information of the PUMA 560 manipulator can be referred to the existing literature [43] with the photo of PUMA 560 shown in Figure 2. It is worth pointing out that when considering only one desired feature point, the kinematic control of the PUMA 560 manipulator can be regarded as utilizing the 6-dimensional joint space to control 2-dimensional image space, which can approximately treat the PUMA 560 manipulator as a redundant manipulator.

In the first place, the simulation setting and the neural network parameters are introduced. Simply put, the parameters of the neural network and camera system are set as $u_p = v_p = 256$ pixel, $\kappa_a = \kappa_b = 8 \times 10^4$ pixel/m, $l = 8 \times 10^{-3}$ m, $d = 2$ m, $v^* = [256, 256]^T$ pixel, $\delta = 10$, $\gamma = 10^3$, and $\alpha = 20$. As to the state and physical constraints of the PUMA 560 manipulator, the states are chosen as $\vartheta_0 = [0.3, -0.9, 0.4, 0.3, -1, -0.2]^T$ rad, the initial coordinate of feature point $v_0 = [169, 104]^T$ pixel, $\dot{\vartheta}^+ = -\dot{\vartheta}^- = [0.4]_{6 \times 1}$ rad/s, and $\ddot{\vartheta}^+ = -\ddot{\vartheta}^- = [3]_{6 \times 1}$ rad/s².

The simulation results are provided in Figure 3. As depicted in Figure 3, the PUMA 560 manipulator successfully achieves the desired feature point driven by GRNN (26). The error ε in Figure 3(b) and $\dot{\varepsilon}$ in Figure 3(c) converge to zero in 1 s. With regard to joint information, Figure 3(d) through Figure 3(f) record the joint acceleration, joint velocity, and joint angle during the simulation, respectively. It is worth emphasizing that joint acceleration and joint velocity are maintained within the designed physical constraints, which ensures the safe execution of the task. Overall, the above results indicate the feasibility and efficiency of the proposed GRNN (26) when handling the visual servoing issue.

To demonstrate the superiority of the proposed method, the traditional pseudoinverse method is employed to deal with the visual servoing problem with results provided in Figure 4. The control law adopted by the traditional pseudoinverse method is generalized as

$$\ddot{\vartheta} = \mathcal{J}^\dagger(-\beta\dot{v} - \varepsilon(v - v^*) - \dot{\mathcal{J}}\dot{\vartheta}), \quad (38)$$

with $\beta = 25$ and $\varepsilon = 100$. It is worth pointing out that the investigations of visual servoing based on the pseudoinversion operation of the Jacobian matrix are common and effective in the existing method [21, 26, 28]. Nevertheless, the pseudoinversion operation of a matrix brings more computational complexity, and the conventional pseudoinverse methods do not take joint limits into account,

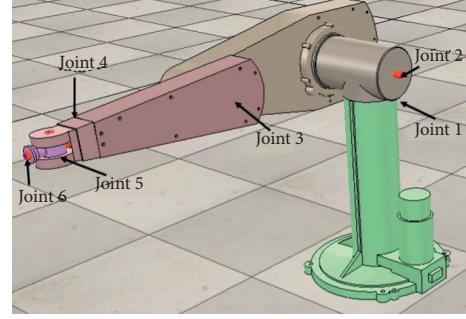


FIGURE 2: Photo of PUMA 560 with six joints.

which are regarded as the deficiencies of existing methods [21, 26, 28]. As depicted in Figure 4(a), the error ε quickly converges to zero in 1.5 s, i.e., the manipulator successfully tracks the desired feature point. However, Figure 4(b) indicates that due to the large value of the initial error ε , the generated initial accelerations are even more than 10 rad/s², which would damage the PUMA 560 manipulator. On the contrary, the proposed method (26) limits the acceleration in the physical constraints, which emphasizes the superiority of the proposed method (26).

Beyond that, an illustrative experiment is conducted on a UR5 manipulator (6-DOF) [25] with a visual sensor installed on its end effector, which is assisted by Virtual Robot Experimentation Platform (V-rep). The experiment results plotted in Figure 5 are synthesized by the proposed GRNN (26). Note that in Figure 5(a), the measured object is regarded as the desired point v^* , which can be captured by the visual sensor, and that the center of the sensor view is the feature point v of the robot visual system. By constantly transmitting the error information $\varepsilon = v - v^*$ to GRNN (26), the visual servoing issue can be solved with v approaching v^* as described in Figures 5(b) and 5(c), which implies the validity of the proposed GRNN (26).

6. Comparisons

In this section, some existing visual servoing approaches [21, 25–27, 42, 43, 48] are assembled in Table 1 to highlight the superiority of the proposed quadratic programming scheme (11)–(14). The following points can be determined. A majority of the existing techniques [21, 26, 27, 48] utilize the pseudoinverse method to carry out research. These approaches often take no account of joint physical constraints, which may lead to a large generated control signal and even cause damage to the manipulator. On the other hand, it is well known that the pseudoinverse operations involved are computationally onerous. Thirdly, the present research on visual servoing at acceleration level is relatively lacking [21, 26]. Therefore, in terms of joint acceleration, the quadratic programming scheme (11)–(14) avoids the pseudoinverse operation by utilizing the matrix transpose operation and meanwhile takes the joint constraints into account. This demonstrates the superiority of the proposed quadratic programming scheme (11)–(14) (Table 1).

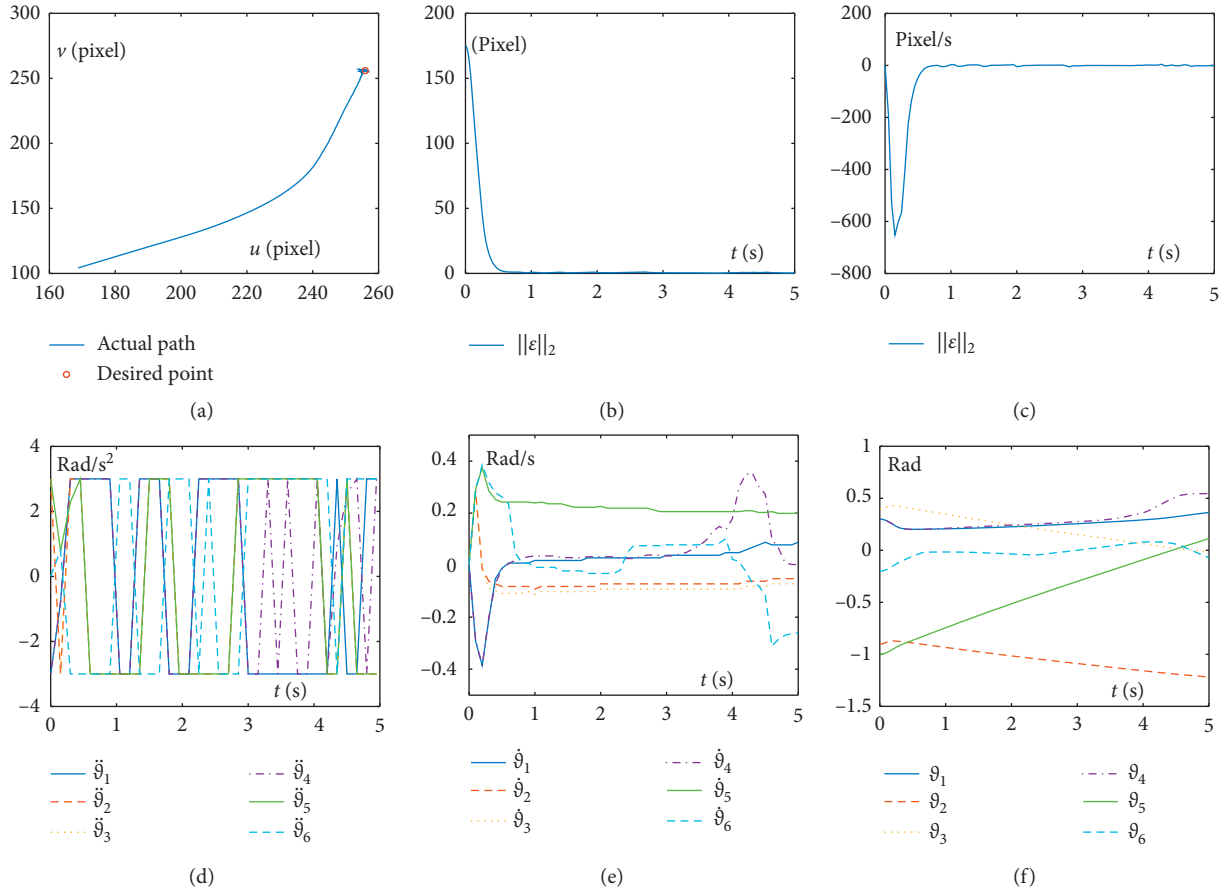


FIGURE 3: Simulation results on the PUMA 560 manipulator driven by the quadratic programming scheme (11)–(14) aided with GRNN (26) for the visual servoing task with execution time $T = 5$ s. (a) Actual path and desired point in image space. (b) Time history of 2-norm of error ϵ . (c) Time history of 2-norm of error $\dot{\epsilon}$. (d) Time history of joint acceleration. (e) Time history of joint velocity. (f) Time history of joint angle.

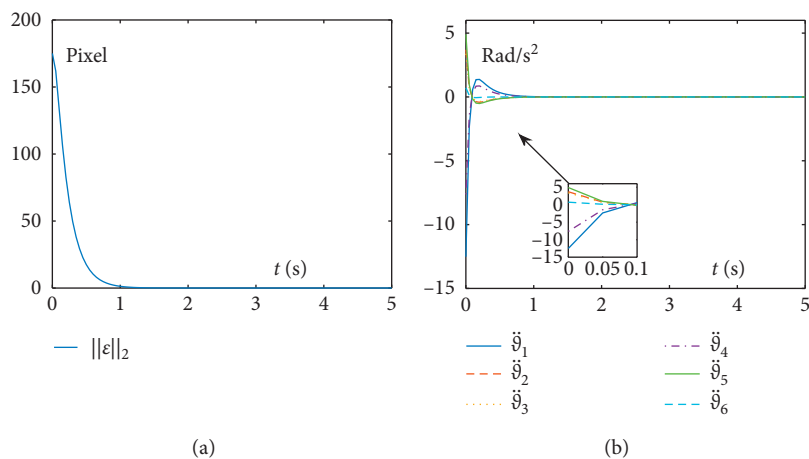


FIGURE 4: Simulation results on the PUMA 560 manipulator driven by the pseudoinverse method (38) for the visual servoing task with execution time $T = 5$ s. (a) Time history of 2-norm of error ϵ . (b) Time history of joint acceleration.

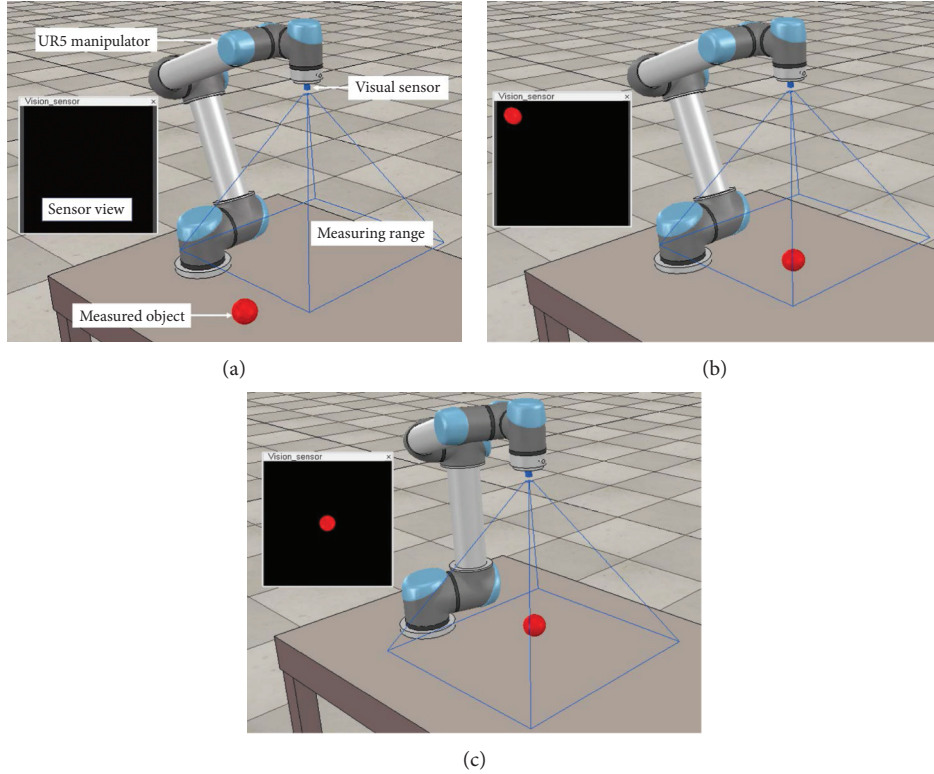


FIGURE 5: V-rep experiment results on the UR5 manipulator driven by the quadratic programming scheme (11)–(14) aided with GRNN (26) for the visual servoing task. (a) UR5 manipulator installed with a visual sensor. (b) Initial state of the robot visual system. (c) Final state of the robot visual system.

TABLE 1: Comparisons among different approaches for visual servoing of redundant manipulators.

	Command level	Gradient descent method	Joint velocity constraints	Joint acceleration constraints	Pseudoinverse operation	Number of neurons
Scheme (26)	Acceleration	Yes	Yes	Yes	No	n
Scheme in [21]	Acceleration	No	No	No	Yes	n
Scheme in [25]	Velocity	No	Yes	No	No	$n + 2$
Scheme in [26]	Acceleration	No	No	No	Yes	n
Scheme in [27]	Velocity	No	No	No	Yes	n
Scheme in [42]	Velocity	No	Yes	No	No	$n + 2$
Scheme in [43]	Velocity	No	Yes	No	No	$n + 2$
Scheme in [48]	Velocity	No	No	No	Yes	n

7. Conclusion

In this paper, the vision servoing issue has been formulated as a constrained quadratic programming scheme at the acceleration level with physical constraints considered. Then, a GRNN has been proposed via the gradient descent method and compensation term with the stability analyses provided. After that, simulation examples have been carried out to demonstrate the correctness of

theoretical analyses and the validity of the proposed method. Note that the proposed method has resolved the visual servoing issue at the acceleration level and also has considered the joint constraints of the manipulator to guarantee the safe operation of the manipulator. For the further research direction, the authors are going to investigate the uncertain conditions and optimization in the visual system, such as noise suppression [56] or Jacobian estimation [57] and manipulability optimization [58].

Data Availability

The data in the paper are not made public online.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the Guangzhou Sport University Innovation and Strengthen Project under Grant 5200080589, in part by the Ministry of Education Industry-Academic Cooperation Collaborative Education Program of China under Grant 201901007048, in part by the Research and Development Foundation of Nanchong (China) under Grant 20YFZJ0018, and in part by the Fundamental Research Funds for the Central Universities under Grant lzujbky-2019-89.

References

- [1] Z. Shen, A. Elibol, and N. Y. Chong, "Understanding non-verbal communication cues of human personality traits in human-robot interaction," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 6, pp. 1465–1477, 2020.
- [2] T. Li, H. Zhao, and Y. Chang, "Delay-dependent stability in uncalibrated image-based dynamic visual servoing robotic system," *Complexity*, vol. 2018, Article ID 1360874, 14 pages, 2018.
- [3] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, "Robot learning system based on adaptive neural control and dynamic movement primitives," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 777–787, 2019.
- [4] L. Jin, S. Li, L. Xiao, R. Lu, and B. Liao, "Cooperative motion generation in a distributed network of redundant robot manipulators with noises," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 48, no. 10, pp. 1715–1724, 2018.
- [5] Z. Xie, L. Jin, X. Du, X. Xiao, H. Li, and S. Li, "On generalized RMP scheme for redundant robot manipulators aided with dynamic neural networks and nonconvex bound constraints," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 9, pp. 5172–5181, 2019.
- [6] Y. Zhang, S. Li, S. Kadry, and B. Liao, "Recurrent neural network for kinematic control of redundant manipulators with periodic input disturbance and physical constraints," *IEEE Transactions on Cybernetics*, vol. 49, no. 12, pp. 4194–4205, 2019.
- [7] J. Zhang, L. Jin, and L. Cheng, "RNN for perturbed manipulability optimization of manipulators based on a distributed scheme: a game-theoretic perspective," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 12, p. 5116, 2020.
- [8] L. Jin, S. Li, H. M. La, and X. Luo, "Manipulability optimization of redundant manipulators using dynamic neural networks," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 4710–4720, 2017.
- [9] C. Yang, C. Zeng, Y. Cong, N. Wang, and M. Wang, "A learning framework of adaptive manipulative skills from human to robot," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 2, pp. 1153–1161, 2019.
- [10] H. Huang, T. Zhang, C. Yang, and C. L. P. Chen, "Motor learning and generalization using broad learning adaptive neural control," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 10, pp. 8608–8617, 2020.
- [11] D. Huang, C. Yang, Y. Pan, and L. Cheng, "Composite learning enhanced neural control for robot manipulator with output error constraints," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 1, pp. 209–218, 2021.
- [12] Y. Kong, R. Zhang, Y. Jiang, and X. Xia, "A repeatable optimization for kinematic energy system with its mobile manipulator application," *Complexity*, vol. 2019, Article ID 8642027, 16 pages, 2019.
- [13] H. Zhang, H. Fang, D. Zhang, X. Luo, and Q. Zou, "Adaptive fuzzy sliding mode control for a 3-DOF parallel manipulator with parameters uncertainties," *Complexity*, vol. 2020, Article ID 2565316, 16 pages, 2020.
- [14] S. K. Pradhan and B. Subudhi, "Position control of a flexible manipulator using a new nonlinear self-tuning PID controller," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 1, pp. 82–95, 2020.
- [15] L. Jin and Y. Zhang, "G2-type SRMPC scheme for synchronous manipulation of two redundant robot arms," *IEEE Transactions on Cybernetics*, vol. 45, no. 2, pp. 153–164, 2015.
- [16] C. Yang, G. Peng, Y. Li, R. Cui, L. Cheng, and Z. Li, "Neural networks enhanced adaptive admittance control of optimized robot-environment interaction," *IEEE Transactions on Cybernetics*, vol. 49, no. 7, pp. 2568–2579, 2019.
- [17] Z. Zhang and Y. Zhang, "Variable joint-velocity limits of redundant robot manipulators handled by quadratic programming," *IEEE/ASME Transactions on Mechatronics*, vol. 18, no. 2, pp. 674–686, 2013.
- [18] S. Li, Y. Zhang, and L. Jin, "Kinematic control of redundant manipulators using neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2243–2254, 2017.
- [19] Y. Zhang, S. Li, and X. Zhou, "Recurrent-neural-network-based velocity-level redundancy resolution for manipulators subject to a joint acceleration limit," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 5, pp. 3573–3582, 2019.
- [20] S. Boyd and L. Vandenberghe, *Convex Optimization*, Cambridge University Press, Cambridge, UK, 2004.
- [21] A. Anwar, W. Lin, X. Deng, J. Qiu, and H. Gao, "Quality inspection of remote radio units using depth-free image-based visual servo with acceleration command," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 10, pp. 8214–8223, 2019.
- [22] F. Ke, Z. Li, and C. Yang, "Robust tube-based predictive control for visual servoing of constrained differential-drive mobile robots," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 4, pp. 3437–3446, 2018.
- [23] D. Tsai, D. G. Dansereau, T. Peynot, and P. Corke, "Image-based visual servoing with light field cameras," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 912–919, 2017.
- [24] T. Osa, C. Staub, and A. Knoll, "Framework of automatic robot surgery system using visual servoing," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Taipei, Taiwan, October 2010.
- [25] W. Li, P. W. Y. Chiu, and Z. Li, "An accelerated finite-time convergent neural network for visual servoing of a flexible surgical endoscope with physical and RCM constraints," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 12, p. 5272, 2020.
- [26] M. Keshmiri, W. Wen-Fang Xie, and A. Mohebbi, "Augmented image-based visual servoing of a manipulator using

- acceleration command," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 10, pp. 5444–5452, 2014.
- [27] M. Van, S. S. Ge, and D. Ceglarek, "Fault estimation and accommodation for virtual sensor bias fault in image-based visual servoing using particle filter," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 4, pp. 1312–1322, 2018.
- [28] P. Cigliano, V. Lippiello, F. Ruggiero, and B. Siciliano, "Robotic ball catching with an eye-in-hand single-camera system," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 5, pp. 1657–1671, 2015.
- [29] P. M. Kebria, A. Khosravi, S. M. Salaken, and S. Nahavandi, "Deep imitation learning for autonomous vehicles based on convolutional neural networks," *IEEE/CAA Journal of Automatica Sinica*, vol. 7, no. 1, pp. 136–149, 2020.
- [30] Y. Huang, J. Na, X. Wu, and G. Gao, "Approximation-free control for vehicle active suspensions with hydraulic actuator," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 9, pp. 7258–7267, 2018.
- [31] Y. Li, S. Li, and B. Hannaford, "A model-based recurrent neural network with randomness for efficient control with applications," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 4, pp. 2054–2063, 2019.
- [32] Y. Qi, L. Jin, Y. Wang, L. Xiao, and J. Zhang, "Complex-valued discrete-time neural dynamics for perturbed time-dependent complex quadratic programming with applications," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 9, pp. 3555–3569, 2020.
- [33] P. S. Stanimirović, ivković and Y. Wei, "Recurrent neural network for computing the drazin inverse," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 11, pp. 2830–2843, 2015.
- [34] P. S. Stanimirović, M. Ciri, I. Stojanovic, and D. Gerontitis, "Conditions for existence, representations, and computation of matrix generalized inverses," *Complexity*, vol. 2017, Article ID 6429725, 27 pages, 2017.
- [35] X. Luo, M. Zhou, Y. Xia, Q. Zhu, A. C. Ammari, and A. Alabdulwahab, "Generating highly accurate predictions for missing QoS data via aggregating nonnegative latent factor models," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 3, pp. 524–537, 2016.
- [36] X. Luo, H. Wu, H. Yuan, and M. Zhou, "Temporal pattern-aware QoS prediction via biased non-negative latent factorization of tensors," *IEEE Transactions on Cybernetics*, vol. 50, no. 5, pp. 1798–1809, 2020.
- [37] L. Cheng, W. Liu, T. Huang, Z.-G. Hou, and M. Tan, "A neural-network-based controller for piezoelectric-actuated stick-slip devices," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 3, pp. 2598–2607, 2018.
- [38] S. A. A. Shah, I. Uddin, F. Aziz, S. Ahmad, M. A. Al-Khasawneh, and M. Sharaf, "An enhanced deep neural network for predicting workplace absenteeism," *Complexity*, vol. 2020, Article ID 5843932, 12 pages, 2020.
- [39] L. Jin, S. Li, B. Hu, M. Liu, and J. Yu, "A noise-suppressing neural algorithm for solving the time-varying system of linear equations: a control-based approach," *IEEE Transactions on Industrial Informatics*, vol. 15, no. 1, pp. 236–246, 2019.
- [40] D. Guo, F. Xu, Z. Li, Z. Nie, and H. Shao, "Design, verification, and application of new discrete-time recurrent neural network for dynamic nonlinear equations solving," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 3936–3945, 2018.
- [41] D. Guo, S. Li, and P. S. Stanimirović, "Analysis and application of modified ZNN design with robustness against harmonic noise," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 7, pp. 4627–4638, 2020.
- [42] Y. Zhang, S. Li, B. Liao, L. Jin, and L. Zheng, "A recurrent neural network approach for visual servoing of manipulators," in *Proceedings of the International Conference on Intelligent Manufacturing and Automation*, pp. 614–619, Melbourne, Australia, August 2017.
- [43] Y. Zhang and S. Li, "A neural controller for image-based visual servoing of manipulators with physical constraints," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 11, pp. 5419–5429, 2018.
- [44] Q. Song, Y. Wu, and Y. C. Soh, "Robust adaptive gradient-descent training algorithm for recurrent neural networks in discrete time domain," *IEEE Transactions on Neural Networks*, vol. 19, no. 11, pp. 1841–1853, 2008.
- [45] N. Zhang, W. Wu, and G. Zheng, "Convergence of gradient method with momentum for two-layer feedforward neural networks," *IEEE Transactions on Neural Networks*, vol. 17, no. 2, pp. 522–525, 2006.
- [46] L. Jin, Z. Xie, M. Liu, C. Ke, C. Li, and C. Yang, "Novel joint-drift-free scheme at acceleration level for robotic redundancy resolution with tracking error theoretically eliminated," *IEEE/ASME Transactions on Mechatronics*, p. 1, 2020, In press.
- [47] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 5, pp. 651–670, 1996.
- [48] K. Hashimoto, T. Kimoto, T. Ebine, and H. Kimura, "Manipulator control with image-based visual servo," in *IEEE International Conference on Robotics and Automation*, Sacramento, CA, USA, April 1991.
- [49] P. S. Stanimirović, M. D. Petković, "Gradient neural dynamics for solving matrix equations and their applications," *Neurocomputing*, vol. 306, pp. 200–212, 2018.
- [50] Y. Zhang and S. S. Ge, "Design and analysis of a general recurrent neural network model for time-varying matrix inversion," *IEEE Transactions on Neural Networks*, vol. 16, no. 6, pp. 1477–1490, 2005.
- [51] X. Luo, D. Wang, M. Zhou, and H. Yuan, "Latent factor-based recommenders relying on extended stochastic gradient descent algorithms," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, p. 1, 2019, In press.
- [52] X. Luo, M. Zhou, S. Li et al., "Incorporation of efficient second-order solvers into latent factor models for accurate prediction of missing QoS data," *IEEE Transactions on Cybernetics*, vol. 48, no. 4, pp. 1216–1228, 2018.
- [53] X. Luo, M. Zhou, S. Li, and M. Shang, "An inherently nonnegative latent factor model for high-dimensional and sparse matrices from industrial applications," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 5, pp. 2011–2022, 2018.
- [54] Z. Xie, L. Jin, X. Luo, Z. Sun, and M. Liu, "RNN for repetitive motion generation of redundant robot manipulators: an orthogonal projection-based scheme," *IEEE Transactions on Neural Networks and Learning Systems*, p. 1, 2020, In press.
- [55] H. K. Khalil, *Nonlinear Systems*, Prentice-Hall, Englewood Cliffs, NJ, USA, 3rd edition, 2001.
- [56] L. Xiao, S. Li, K. Li, L. Jin, and B. Liao, "Co-design of finite-time convergence and noise suppression: a unified neural model for time varying linear equations with robotic applications," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 12, p. 5233, 2020.

- [57] Z. Xie, L. Jin, X. Luo, S. Li, and X. Xiao, "A data-driven cyclic-motion generation scheme for kinematic control of redundant manipulators," *IEEE Transactions on Control Systems Technology*, vol. 29, no. 1, p. 53, 2021.
- [58] L. Jin, Z. Zhang, M. Liu, S. Li, L. Xiao, and Z. Yang, "Perturbed manipulability optimization in a distributed network of redundant robots," *IEEE Transactions on Industrial Electronics*, p. 1, 2020, In press.

Research Article

Prediction of the RFID Identification Rate Based on the Neighborhood Rough Set and Random Forest for Robot Application Scenarios

Hong-Gang Wang,¹ Shan-Shan Wang ,¹ Ruo-Yu Pan,¹ Sheng-Li Pang,¹ Xiao-Song Liu,² Zhi-Yong Luo,³ and Sheng-Pei Zhou⁴

¹School of Communication and Information Engineering & School of Artificial Intelligence, Xi'an University of Posts and Telecommunications, Xi'an, Shaanxi 710121, China

²Guangdong Zhongke Zhenheng Information Technology Co. Ltd., Foshan, Guangdong 528225, China

³School of Electronics and Communication Engineering, Sun Yat-Sen University, Guangzhou 510006, China

⁴Shenyang Institute of Automation (Guangzhou) Chinese Academy of Sciences, Guangzhou 511458, China

Correspondence should be addressed to Shan-Shan Wang; 15029070967@163.com

Received 3 September 2020; Revised 12 November 2020; Accepted 3 December 2020; Published 29 December 2020

Academic Editor: Rongxin Cui

Copyright © 2020 Hong-Gang Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

With the rapid development of Internet of Things technology, RFID technology has been widely used in various fields. In order to optimize the RFID system hardware deployment strategy and improve the deployment efficiency, the prediction of the RFID system identification rate has become a new challenge. In this paper, a neighborhood rough set and random forest (NRS-RF) combination model is proposed to predict the identification rate of an RFID system. Firstly, the initial influencing factors of the RFID system identification rate are reduced using neighborhood rough set theory combined with the principle of heuristic attribute reduction of neighborhood weighted dependency, thus obtaining a kernel factor subset. Secondly, a random forest prediction model is established based on the kernel factor subset, and a confusion matrix is established using out-of-bag (OOB) data to evaluate the prediction results. The test is conducted under the constructed RFID experimental environment, whose results showed that the model can predict the identification rate of the RFID system in a fast and efficient way, and the classification accuracy can reach 90.5%. It can effectively guide the hardware deployment and communication parameter protocol setting of the system and improve the system performance. Compared with BP neural network (BPNN) and other prediction models, NRS-RF has shorter prediction time and faster calculation speed. Finally, the validity of the proposed model was verified by the RFID intelligent archives management platform.

1. Introduction

In recent years, ultra-high-frequency (UHF) passive RFID technology has been widely applied in applications of unmanned warehouse, industrial site, new retail store management, and other scenarios due to its excellent ability in long-distance and multitag reading [1]. With respect to the conventional quasistatic RFID system that is usually deployed in a fixed way in particular areas, its system architecture and parameter configuration are unadjustable to some extent, making it difficult to be applied to certain

practical situations. In response to the abovementioned circumstances, a novel mobile RFID system is, therefore, proposed. RFID robots are not only to simply assemble the robot with the RFID system but also to combine the RFID system with the mobile robot to form a unified system. The optimization and control of the RFID system need to fully consider the factors such as tag environment, space, moving speed, and other factors. Compared with the existing conventional quasistatic RFID system, the RFID system on the robot is a typical dynamic system. In the conventional quasistatic RFID system, the system deployment and reading

strategy are relatively fixed, while in mobile RFID robots, the RFID system needs to constantly adjust the parameters and control the robot moving to maximize the adaptation of the environment to obtain the best application performance. When the mobile RFID robot is in task areas, it can adaptively adjust protocol parameters and hardware deployment strategies to accomplish tag reading tasks reliably under dynamic scenarios, thereby improving identification efficiency of the system. In domains of project planning and designing, the RFID system identification rate is the key technical index measuring system quality. In order to improve efficiency of the system's architecture and engineering deployment, predicting the identification rate of the novel mobile RFID system is of critical priority.

Among the existing prediction models of the RFID system identification rate, Liu et al. successively proposed a logistic regression analysis model, learning vector quantization neural network, and other intelligent algorithms to predict the rate of RFID system identification, which achieved good prediction effect. Despite the good effect achieved through adopting the abovementioned proposals, certain shortcomings still occur in practical applications involving huge amount of computation and overfitting [2, 3]. By introducing neighborhood rough set theory, Qiao et al. conducted optimization works with respect to influencing factors of the RFID system identification rate to improve identification efficiency. However, those factors being selected from actual test scenario are rather subjective, which may impose potential disadvantageous impacts on the identification rate of the actual system [4].

All the aforementioned algorithms are used to predict the system identification rate in the conventional quasi-static RFID system, yet few research studies focus on the system in dynamic scenes so far. In response to various tag numbers, complex multipath channel interference, and other factors encountered in designing and deploying the RFID system in dynamic scenes currently, tremendous efforts have been made on validation tests and on reducing miss rate of tags to avoid adjusting parameter deployment to do a lot of testing. Through applying RFID technology to mobile robot while extending its corresponding application schemes to dynamic scenes, a novel mobile RFID system is, therefore, established, in which a new intelligent learning algorithm being referred to as the neighborhood rough set and random forest (NRS-RF) combination model is introduced to predict the system's identification rate.

From the perspective of RFID system hardware deployment, we comprehensively select initial influencing factors to explore the relationship between the system identification rate and each influencing factor. This approach avoids doing a lot of verification tests in order to obtain the optimal deployment strategy. The NRS-RF model predicts the system identification rate in a fast and efficient way, so as to reverse guide the hardware deployment and communication protocol parameters of the RFID system, improve the performance of the RFID system, and meet the engineering needs. This novel mobile RFID system breaks the conventional quasistatic RFID system design and

provides more in-depth scene perception and real-time read-write strategy optimization for practical engineering needs.

The specific steps of the NRS-RF model are as follows:

Firstly, multiple initial influencing factors that affect the identification rate of RFID system are identified comprehensively. In specific, NRS theory is adopted to reduce influencing factors and data redundancy in between these factors, by which the kernel factor subset is selected. Secondly, the bootstrap method is used to resample the training set to support training the random forest prediction model [5–7]. The NRS-RF model is compared with other prediction models such as the backpropagation neural network (BPNN) to verify its advantages in predicting the identification rate of the RFID system. The test results show that the NRS-RF model can accurately and quickly complete the prediction of the RFID system identification rate, and the classification accuracy can reach 90.5%. It effectively guides the project deployment and improves the performance of the RFID system.

Compared with other models such as the BPNN, the NRS-RF model has obvious advantages in terms of classification accuracy and training time. Last but not least, the model is applied to the project of the RFID intelligent archives management platform, thus validating and verifying effectiveness of the proposed model.

The remainder of this paper is organized as follows: Section 2 presents an overall review of related work. Section 3 highlights relevant theoretical methods. Section 4 outlines experimental testing and analysis. Section 5 analyzes simulation results in detail and engineering application. Section 6 summarizes conclusions.

2. Related Work

Generally speaking, the deployment environment of the conventional RFID system is usually located in fixed scenes such as the entrance and exit of a corridor or passageway, for which reason it is inconvenient to apply tag identification in these areas. Under the circumstance of the scene with large identification area, a multireader mechanism is generally adopted [8]. However, the expenditure for improving the mechanism will be huge and unaffordable, not to mention potential collision between readers. Currently, as diverse algorithms emerge, protocol algorithms of reader anticollision has attracted much attention, among which heartbeat algorithm [9] and color wave algorithm, as well as the improved version of corresponding algorithms, prosper [10, 11]. In this paper, as our study mainly concentrates on single mobile reader mechanism, the RFID system with multiple readers, therefore, does not need to be particularly introduced in detail.

The state-of-the-art mobile RFID system is suitable for tag identification in small and medium areas. This mobile RFID system that works at UHF does not require any power supply (passive), featuring with characteristics of long identification distance, small size, strong directionality, and outstanding robustness against environmental changes [12]. The system ensures that not only all tags are covered within

the identification range of readers' signal and can be read successfully but also corresponding information processing belongs to the extended applications of the conventional fixed RFID system. However, the aforementioned dynamic RFID system cannot achieve its optimal status due to the vulnerable identification rate being seriously affected by external interference that may be induced by subjectivity, experience, and real-time hardware deployment of the system. Therefore, the dynamic RFID application scene needs to guarantee more in-depth scene perception and real-time read-write strategy optimization, which poses new challenges for RFID read-write technology.

Under new technological strategies and formats such as artificial intelligence, big data, new-generation robots, intelligent manufacturing, and new retail, RFID technology is facilitating robots to complete automation and dehumanization of warehouse management, industrial site, and new retail management [13]. Minh et al. introduced relations between the RFID system identification rate and influencing factors in a mobile system, established a support vector machine (SVM) model, and predicted the RFID system identification rate. However, certain errors still existed between the prediction results and the actual identification rate. In the abovementioned methodology, factors that affect the system identification rate were selected inadequately, and the actual scene was neither validated nor verified [14].

In January 2016, *Keonn* technology company of United States presented an RFID robot being referred to as *advanrobot* and applied it to a clothing retail scene to achieve fast and accurate mobile reading [15]. In April 2016, a well-known United States manufacturer named *Thingmagic* introduced adaptive duty cycle technology in the reader to minimize the reader's working time with respect to tag numbers, thus reducing power consumption [16]. Wang et al. proposed an efficient energy detection and calculation method for the RFID system in a dynamic scene, which is different from conventional anticollision algorithm. The tag helps the reader to judge whether the tags in the identification area collide with each other or not by sending a PBD burst time. If collision occurs, the collision problem will be solved by recursive polling, thereby improving the tag identification rate of the system [17]. In August 2016, *Impinj* proposed a scheme of the *Speedway Revolution* RFID reader, to which automatic performance setting is introduced based on environmental noise detection and on automatic dynamic antenna switching technology to optimize read-write time and efficiency [18].

In this paper, aiming at promoting sustainable development of RFID technology, research studies on the RFID robot and on developing new generation of adaptive read-write technology are, therefore, conducted, satisfying specific demand of niche market while accelerating technological progress of the industry. In order to improve the system identification rate from a physical perspective of hardware deployment, an intelligent learning algorithm named RFID system identification rate prediction is proposed based on the NRS-RF model. By using our proposed method, not only is the relation mined between diverse influencing factors and the system identification rate but

also intelligent scene perception is realized through model matching instead of using conventional methods, thereby improving the prediction accuracy. Moreover, by combining the novel mobile RFID system, the optimal combination of hardware deployment configuration scheme is obtained to improve the RFID system's identification rate, thus maximizing effectiveness and efficiency of the hardware deployment while ensuring cost efficiency in terms of labor force and resources.

The philosophy of using the NRS-RF model can be summarized as follows:

- (1) Influencing factors of the RFID system identification rate are selected as the sample data, in which the NRS theory is used to reduce the attribute of these factors, to select the kernel factor subset that affects the identification rate, and to reduce the input dimension of nonlinear mapping.
- (2) Based on the kernel factor set, the prediction model is constructed featuring with the 2-classification random forests RFID system identification rate, upon which a novel mobile RFID experimental test platform is established accordingly. Due comparison analysis is performed between the NRS-RF model and the BPNN and other prediction models in terms of OA, Kappa coefficient, RMSE, MAE, training time and prediction time, and correlation. The test results show the superiority of the NRS-RF model.
- (3) The prediction model is applied to the intelligent archives management platform, and the importance distribution of influencing factors to RFID system identification rate classification prediction is analyzed, verifying effectiveness and efficiency of the proposed model.

3. Methods

3.1. Reduction Feature Factors. Emerging as an innovation in classical rough set theory, neighborhood rough set (NRS) theory was put forward by Lin in 1988 [19, 20]. The idea of NRS algorithm is that, in the real space, each data point will form a neighborhood $\delta_B(x_i)$ and the data in the neighborhood family will constitute the basic information particles [21–23]. NRS solves the problem of numerical data set that is not easy to be processed in classical rough set theory, removes redundant data features, and selects the key factors that affect the identification rate of the RFID system [24, 25].

In the RFID system, the information system W is composed of quad-tuple $W = (U, Y, V, f)$, where U is the sample number set of the identification rate, Y is the sample set of the identification rate ($Y = C \cup D$), C is the influencing factors of the identification rate functioning as the attribute set, and D is the classification level of the identification rate functioning as the decision attribute. This quad-tuple information system W is called the decision table, within which V denotes the value field of attribute and f represents the mapping relation used to specify the property value of sample x , that is, $f = U \times Y \rightarrow V$.

If the sample $x_i \in U$, the neighborhood condition of x_i needs to satisfy $\delta_B(x_i) = \{x_j | x_j \in U, \Delta_B(x_i, x_j) \leq \delta\}$, where Δ denotes distance function, for any $x_1, x_2, x_3 \in U$, and Δ satisfies

$$\begin{cases} \Delta(x_1, x_2) = \Delta(x_2, x_1), \\ \Delta(x_1, x_2) \geq 0, \Delta(x_1, x_2) = 0; \quad \text{at this time } x_1 = x_2, \\ \Delta(x_1, x_3) \leq \Delta(x_1, x_2) + \Delta(x_2, x_3). \end{cases} \quad (1)$$

For any attribute set L , when it is classified, the indiscernible data will be grouped into one class. They belong to the indiscernible relation, which can be given by the following equation:

$$\text{IND}(L) = \{(x, y) \in U \times U: f(x, a) = f(y, a), a \in P\}. \quad (2)$$

For the indistinguishable relationship group $H \subseteq Y$, $a \in H$, if the relationship $\text{IND}(L) = \text{IND}(H - \{h\})$ is satisfied, it is considered that $\{h\}$ is the redundant data on H , which can be reduced. It can be defined that the value of theory field is V , R is the equivalent relation on V , to which the upper approximation, the lower approximation of neighborhood rough set, and the boundary field of subset x satisfy the following equations, respectively:

$$\bar{R}(X) = \cup \{Y \in U | R: Y \cap X \neq \emptyset\}, \quad (3)$$

$$R(X) = \cup \{Y \in U | R: Y \subseteq X\}, \quad (4)$$

$$BN_R(X) = \bar{R}(X) - R(X), \quad (5)$$

where $\text{Pos}_R(X) = R(X)$ is the positive domain of subset X and $\text{Neg}_R(X) = U - \bar{R}(X)$ is the negative domain. For any $c \in C$, the dependence degree of decision attribute D on condition attribute c is defined as

$$\gamma_a = \frac{\text{Pos}_a(D)}{U}. \quad (6)$$

If two random variables are defined, the correlation degree calculated using mutual information measure can be satisfied by the following equation:

$$I(X, Y) = - \sum_{x \in V_x} \sum_{y \in V_y} p(x, y) \log_2 \frac{p(x, y)}{p(x)p(y)}. \quad (7)$$

3.2. Random Forest Prediction. Random forest (RF) algorithm is based on the decision tree as a learning machine to build bagging integration [26, 27], thus further introducing the selection of random attributes. Specifically, conventional decision tree algorithm selects an optimal attribute in the current attribute set when selecting the partition attribute, whereas the RF algorithm randomly selects a subset containing K -th attributes in the attribute set, through which an optimal attribute is selected from the subset selection for classification. Using this kind of random selection, the random forest can avoid disadvantages of overfitting,

exhibiting excellent antinoise performance. Outperforming other intelligent algorithms, it only requires simple computation while maintaining cost efficiency. The principle of RF algorithm is given below, as shown in Figure 1.

The random forest algorithm adopts an integrated algorithm; the classification accuracy of the algorithm itself is much higher than other single algorithms, so the accuracy is higher. The random forest algorithm can handle high-dimensional data without any feature selection. When bootstrap sampling is performed on training samples, out-of-bag data will be generated. Unbiased estimates of true errors can be obtained in the process of model generation without loss of training data. Due to the simple implementation, high accuracy, and strong antioverfitting ability of the algorithm, when faced with nonlinear data such as the identification rate of the RFID system, the model shows high classification accuracy and is also suitable as a benchmark model.

Due to the change of tag numbers and the complex multipath channel interference in the architecture and deployment of the RFID system, the prediction value of the RFID system identification rate is, therefore, discretized. In order to avoid adjusting the parameters to obtain optimal deployment strategy and to do a lot of testing and verification, it is necessary to comprehensively select the influencing factors, thus mining the nonlinear relationship between the RFID system identification rate and its influencing factors. From the perspective of hardware deployment, the RFID hardware deployment and communication protocol parameters should be optimized and adjusted with the purpose to reduce the missed rate of tags and to improve the system performance. The relation between the influencing factors and the identification rate is obtained using RF classification prediction algorithm, to which its mathematical model can be expressed by the following equations:

$$\text{RFP} = \{Tn, M\}, \quad (8)$$

$$d = f(x_1, x_2, \dots, x_n, \text{RFP}), \quad (i = 1, 2, \dots, n). \quad (9)$$

In equation (8), RFP is the parameter set of the random forest prediction model, Tn is the number of regression trees in the model, and M is the number of influencing factors. In equation (9), the prediction method of the RFID system identification rate is defined, where f is the uncertainty function relation of random forest classification algorithm, d is the system identification rate, and x_i is the i -th index factor affecting the identification rate, including number of tags, number of antennas, reading distance, and other parameters. n is the number of influencing factors of the system identification rate. Another advantage of the random forest algorithm is that the influencing factors participating in the algorithm can measure the importance degree of the classification. The contribution value of the influencing factors can be determined by calculating the information gain rate of the dataset. The information gain rate is positively correlated with certainty of the influencing factor, indicating that the higher the information gain rate, the stronger the certainty of the influencing factor. Calculation of the information gain is given by the following equations:

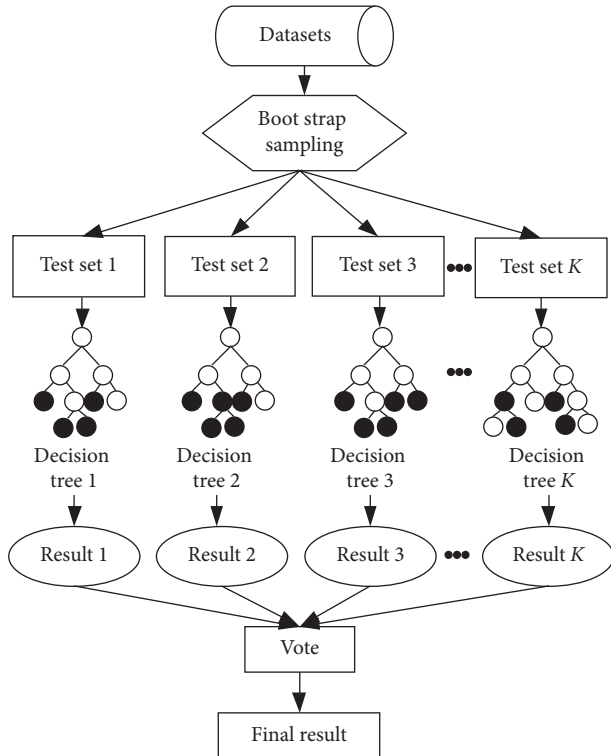


FIGURE 1: Schematic diagram of the random forest algorithm.

$$g(D, X_i) = H(D) - H(D|X_i), \quad (10)$$

where $H(D)$ is the information entropy of dataset D , namely,

$$H(D) = - \sum_{i=1}^n D_i \log_2 D_i, \quad (11)$$

where $H(D|X_i)$ is the entropy change brought by X_i variables under the condition of dataset D .

$$H(D|X_i) = H(D, X_i) - H(X_i). \quad (12)$$

4. Experiments

4.1. Selection of Influencing Factors. We use the UHF passive RFID system as the experimental background to build a novel mobile UHF passive RFID test platform. This type of RFID system has many advantages that users cannot refuse. This technology can achieve no human intervention and without-contact identification. This technology can realize nonhuman intervention and noncontact identification. The system can be used in many harsh environments, such as automatic container terminal yard operation system identification, high-speed moving object identification, multitag identification, and other scenarios at low cost.

For the novel mobile RFID system, because of its mobility, the tags in the identification area have the opportunity to be identified and processed, which can linearly expand the identification range of the reader. Under the same test conditions, the novel mobile RFID system can identify more tags and obtain the system identification rate faster than the

conventional quasistatic RFID system. However, when the system is put into actual engineering deployment, it will encounter factors such as changes in the number of tags and complex multipath channel interference. When we adjust the hardware deployment of the system, there will be subjectivity and timeliness, resulting in the identification range of the reader. In the blind area, we can reduce the error of the RFID system identification rate by optimizing the hardware configuration reasonably, so as to maximize the performance of the system.

In RFID system project planning and designing, the system identification rate that represents the number of tags being successfully read in the inventory process accounts for the total number of tags. It is an important indicator in measuring the system performance. Generally, the system identification rate correlates with controllable factors involving moving speed of the reader, number of tags, and other factors and with uncontrollable factors including multipath channel interference, Doppler effect, and other factors, as shown in Figure 2. During the experiment, eleven controllable factors are selected as conditions, including height of the antenna $P1$, number of tags $P2$, horizontal distance between the tag and shelf $P3$, number of reader polling cycles $P4$, moving speed of the antenna $P5$, horizontal angle of the antenna $P6$, vertical angle of the antenna $P7$, number of antennas $P8$, light intensity (the sunlight and the dark room without light are selected under same experimental conditions) $P9$, shelf height $P10$, and reader transmitting power $P11$. The level of the RFID system identification rate is regarded as the decision attribute.

In order to verify prediction accuracy of the proposed intelligent prediction method for the RFID system, firstly, orthogonal experiments are conducted using a variable-controlling approach. The antenna height is set to 0.6 m, 0.9 m, and 1.2 m, respectively; the number of tags is set to 90, 130, 150, and 180, respectively; the distance between tags and antennas is set to 0.8 m, 1.5 m, and 1.9 m, respectively; the number of polling turns of the reader is set to 1, 2, and 3, respectively; the moving speed of the antenna is set to 0.3 m/s, 0.6 m/s, and 0.9 m/s, respectively; the horizontal angle of antenna is set to 0° and 30° , respectively; the vertical angle of antenna is set to 0° and 30° , respectively; the number of antennas of the same specification is set to 1 and 2, respectively; the light intensity is set to 0 (sunlight) and 1 (darkroom), respectively; the shelf height is set to 0.6 m and 1.2 m, respectively; and the reader transmitting power is set to 18 dBm, 23 dBm, and 28 dBm, respectively.

We need to traverse 10368 sets of cross experiments and record the identification rate of the RFID system, in which each group of influencing factors is used to conduct 3 experiments, and both the average value of the RFID system identification rate and the influence factors can be taken as the sample data.

In practical engineering applications, the new type of portable RFID system should be expected to reach 100% identification rate on tagged goods inventory. In practical applications, however, due to complex constraining factors such as multipath effect, it is not ideal to rely on the current hardware. Therefore, the threshold of the identification rate

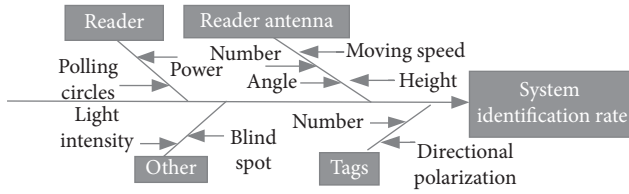


FIGURE 2: Influencing factors and the system identification rate.

can only reach 95%, which needs to be adjusted with respect to differing projects according to actual situations. We believe that whether the identification rate is qualified or not lies in satisfying the requirements of actual situations. If the identification rate of the system exceeds 95%, it is regarded as high rate and can be categorized into type 1; otherwise, it is labelled as unqualified type 2 if the identification rate is less than the threshold value of 95%.

4.2. Comprehensive Analysis of the Test Platform. According to the proposed prediction model of the RFID system identification rate, a novel mobile UHF passive RFID system is constructed combining a mobile robotic car reader antenna and other devices as an RFID experimental test platform. Featuring 9 m length, 4 m width, and 3.75 m height, the platform is built in the corridor of an open classroom, as shown in Figure 3(a). The product of water drop robot of *Beijing Yunji technology company* is selected as the robotic vehicle used in the experiment, which maneuvers with abilities of highly sensitive intelligent perception and positioning navigation.

As shown in Figure 3(b), the robotic vehicle that is enabled with a path scanning function can adjust its moving speed following the experimental requirements. The vehicle scans the test site, where the white color represents the area it walks within, the gray color represents the unexplored area, and the black solid line represents the obstacle information established in the map. The scanning results are shown in Figure 4. The mobile robotic vehicle is equipped with an adjustable tripod, reader, and antenna. The detailed experimental specifications are introduced as follows.

An RFID reader (MODEL Mercury6, *ThingMagic*) is adopted, which has stable read-write performance supporting reading ISO 18000-6C protocol standard in wired mode with a 9 dBi circular polarization reader antenna. The four-layer book shelf on the left side is 1.5 m * 2 m, with a certain number of books evenly distributed on each layer and with archive tags pasted on the side of each book. The tag used is UHF passive and its working frequency mainly ranges from 860 MHz to 920 MHz, exhibiting excellent directionality and satisfactory read distance.

The novel mobile RFID experimental system has the following characteristics:

- (1) The novel mobile RFID system overcomes the traditional quasistatic RFID system because the identification area is static, there is always the problem of blind spots in identification, and it avoids the disadvantages of manual handheld readers that cause

large errors in the identification rate of the RFID system

- (2) In order to identify all tags in the area, the novel mobile RFID system requires a mobile robot to poll the tags in the reading area, and the tags are covered by the reader signal area, at least, once
- (3) Some tags will repeatedly enter and leave the signal area of the reader
- (4) Some tags will be identified multiple times by the reader

The experiment builds a novel mobile UHF passive RFID test platform. Because only a movable single reader mechanism can be used to realize multitag identification, the cost is much less than the RFID system with multireader mechanism, which also saves multiple readings. It also saves equipment installation and wiring costs. However, the system still has the following limitations.

Due to the sector-shaped antenna radiation field of RFID mobile robotic vehicle, the echo signal of tags on both sides of the sector is weak and vulnerable to interference. With the range of sector coverage, the echo signal of far end tag A is weak, whereas that of near end tag B is strong. During the process of identifying tags, the tags on both ends of the sector may block the reader antenna if the vehicle moves too fast, thus resulting in information loss.

In order to facilitate simple and direct identification, different moving speeds of the vehicle are selected as initial influencing factors for the RFID system. As shown in Figure 5, it is required that the vehicle must move along the linear path with the signal radiation radius R at a reasonable and constant speed under specified conditions to obtain the optimal identification rate of the RFID system. Our future research will be focusing on the optimal disk point path to obtain the maximum identification rate of the system.

Throughout the process of identifying tags, only by adjusting the parameters from the physical perspective of hardware deployment can the optimal deployment scheme of combination be obtained. In the abovementioned experiment, however, the ideal identification rate of 100% cannot be achieved merely through adjusting hardware configurations. Admittedly, there are still other detrimental encumbrances that may potentially induce blind area problems, to which limitation involving antenna polarization mismatching and multipath fading can be attributed largely.

In response to the situation that the number of tags on the shelf is positively correlated with the size of storage area, only small-size tags are selected to be uniformly placed in the identification area without considering the serious tag collision problem in the scene with dense multiple UHF tags. With respect to other initial factors selected in Section 4.1, such as antenna height, polling circle of the reader, and horizontal distance between the antenna and shelf, a reasonable test range should be conscientiously selected. Furthermore, there are still many important tasks to be accomplished, which include, but not limited to, conducting orthogonal combination of controllable variables and testing



FIGURE 3: Test environment, (a) experimental test platform, and (b) water droplet robot.

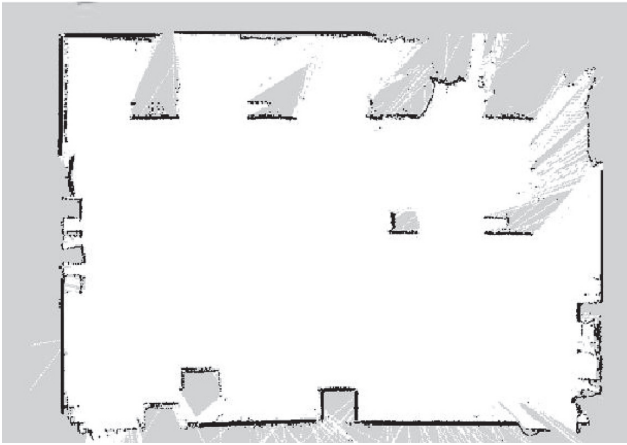


FIGURE 4: Two-dimensional scanning image of the test environment.

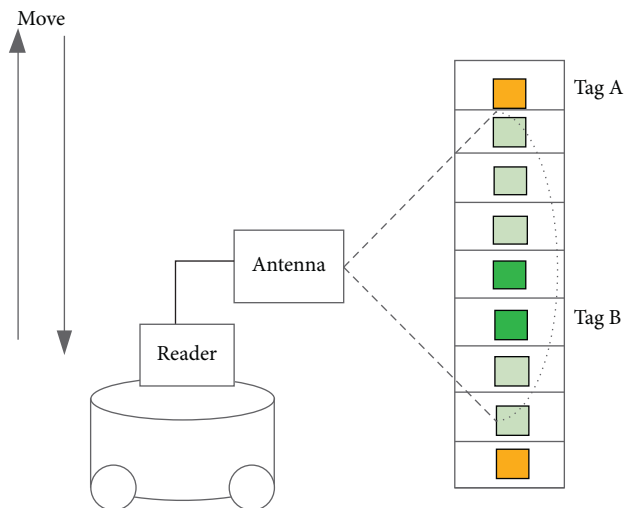


FIGURE 5: Schematic diagram of the novel mobile RFID system with the robotic vehicle.

diverse combinations of influencing factors, as well as recording the RFID system identification rate.

4.3. Proposed Algorithm

4.3.1. Reducing Influencing Factors. In Section 4.1, 11 controllable factors (see Section 4.1) are selected as condition attributes, and the level of the RFID system identification rate is taken as a decision attribute.

In order to reduce the redundancy in between the aforementioned 11 groups of influencing factors and to improve the prediction accuracy of system identification, heuristic reduction algorithm of neighborhood weighted dependency is, therefore, adopted. The purpose of using this algorithm aims on reducing the attribute reduction of influencing factors and on obtaining the kernel factor subset R [28, 29]. The algorithm is a forward greedy attribute reduction algorithm based on the attribute importance of weighted dependence (Algorithm 1). Detailed explanation of the algorithm is described as follows [30, 31].

In a neighbourhood decision system $W = (U, CUD, V, f, \varepsilon)$, for $Z \subset C$, we define attribute $x \in Z - B$ and define w as adjustment parameter and $w\gamma_B^\delta$ as weighted dependency, so the importance degree of Z and D based on the weighted dependence degree satisfies the following equation:

$$\text{sig}(x, Z, D) = w\gamma_{B \cup x}^\varepsilon(D) - w\gamma_B^\varepsilon(D). \quad (13)$$

The neighborhood radius $\varepsilon = 0.28$ was selected, the attribute subset was selected from the empty set, and the reduction subset was selected in turn to build an ordered reduction attribute subset. The dependence of the 11 groups of influencing factors increased as the important attributes of the reduction subset increase, thereby finally obtaining the decision table M1, as shown in Table 1. From Table 1, it is obvious that there are 500 sets of data after reducing sample set U which is composed of $P1, P2, P3, P4, P5, P8$, and $P11$. Compared with the initial influencing factors, after performing heuristic reduction algorithm of neighborhood

Input: Neighborhood decision system $W=(U, C \cup D, V, f, \varepsilon)$, ε is neighborhood threshold, T is temporary subset.

Output: Reduction subset R .

Steps:

- (1) $\phi \longrightarrow R; T \longrightarrow C - R;$
- (2) $\forall x_i \in T$, Calculate the attribute importance sig of the sample $\text{sig}(x_i);$
- (3) $x_k = \arg(\max(\text{sig}(x_i)))$, Find the attribute with the most important attribute of attribute reduction attribute subset $x_i;$
- (4) **If** ($\text{sig}(x_i) > 0$)
- (5) $R \longleftarrow R \longleftarrow \{x_k\}, T \longleftarrow T - \{x_k\};$
- (6) **Else**
- (7) $\text{return } R;$
- (8) **End for**

ALGORITHM 1: Forward greedy attribute reduction.

rough set weighted dependency, some redundant factors are removed, and the input features are changed from the original 11 sets to 7 sets.

4.3.2. Random Forest Prediction. The prediction model in this paper is based on the random forest toolbox developed by the University of Colorado [32], and the corresponding codes were written in the operating environment of MATLAB7.1. The RFID identification rate prediction model based on the NRS-RF includes 6 steps. Specific prediction steps of the NRS-RF are demonstrated as follows:

- (1) *Normalized Input Data.* The kernel factor subset {antenna height $P1$, number of tags $P2$, distance between tags and shelf $P3$, number of polling cycles $P4$, moving speed of the antenna $P5$, number of antennas $P8$, and transmission power of the reader $P11$ } obtained by attribute reduction of neighborhood rough set theory is used to construct $500 * 7$ -dimensional sample data as input variables of the random forest model. The sample data are processed by the following equation:

$$x'_i = \frac{x_i - \min(x)}{\max(x) - \min(x)}. \quad (14)$$

The predicted value of the identification rate of the system is transformed by

$$x = x'(x_{\max} - x_{\min}) + x_{\min}, \quad (15)$$

where x is the initial RFID system identification rate value and x_{\max} and x_{\min} are the maximum and minimum value of the system identification rate, respectively.

- (2) *Bootstrap Sampling Training Subsets and Decision Tree.* The bootstrap method was used to perform n times of resampling from the sample set S and to randomly generate n training subsets S_i with the same number of samples. During bootstrap sampling of training samples, 1/3 out-of-bag (OOB) data will be left behind. The OOB precision estimation of each decision tree can be obtained through out-of-pocket

TABLE 1: Decision table M1.

W	1	2	...	251	252	...	351	...	500
$P1$	1	3	...	2	3	...	2	...	6
$P2$	3	7	...	5	2	...	2	...	4
$P3$	5	6	...	2	1	...	7	...	2
$P4$	3	5	...	4	7	...	4	...	2
$P5$	5	2	...	5	2	...	1	...	6
$P8$	3	6	...	5	3	...	6	...	2
$P11$	2	3	...	1	4	...	5	...	3
V	1	1	...	1	1	...	1	...	1

samples. The OOB precision estimation of left and right decision trees in the forest can be averaged to obtain the generalization precision estimation of random forest. For all the sample subsets S_i , the CART algorithm is performed to construct decision trees, thus combining these trees to form a random forest, which is expressed as $\{C_1, C_2, \dots, C_i\}$.

- (3) *Node Split Growth.* When the nodes of the decision tree are splitting and growing, the input parameters in the Mtry block prediction model are randomly taken as the split subset of the current node. The value of Mtry represents the disturbance degree of the model attributes because the value in the model is sensitive which directly affects the prediction accuracy of the model. The value can be given according to the empirical equations:

$$M_{\text{try}} = \left\lceil \frac{M}{3} \right\rceil, \quad (16)$$

$$M_{\text{try}} = \lceil \log_2 M \rceil, \quad (17)$$

where M is the number of input variables which is 6 in this study. Hence, according to equations (16) and (17), the M_{try} value is 2. When the nodes are divided in the subset, the Gini index in CART algorithm is taken as the minimum principle to select the optimal split influencing factor and optimal split value. During the splitting process, none of pruning operations is performed, and the M_{try} block degree remains constant. The Gini system is defined by the following equation:

$$\text{Gini}(t_i) = 1 - \sum_{K=1}^K P_k^2, \quad (18)$$

where t_i represents the current influencing factor, K represents the number of groups of the influencing factors t_i , and P_k represents the probability that the sample point belongs to the K class. After determining the optimal splitting influencing factor t_i , if a subset S_i is split into two subsets S_{i1} and S_{i2} with respect to t_i , the optimal splitting value “ a ” can be calculated by the following equation:

$$\min \text{Gini}(t_i, a) = \frac{|S_{i1}|}{|S_i|} \text{Gini}(S_{i1}) + \frac{|S_{i2}|}{|S_i|} \text{Gini}(S_{i2}), \quad (19)$$

where $|S_i|$, $|S_{i1}|$, and $|S_{i2}|$, are the samples of S_i , S_{i1} , and S_{i2} , respectively.

- (4) *Prediction Sample Category.* After each decision tree is constructed from the bottom to the top, the integrity of the tree is preserved without performing pruning operations, and all the decision trees are tested with test set X to obtain the test sample of the prediction category of the RFID system identification rate, which can be expressed as $\{C_1(X), C_2(X), \dots, C_n(X)\}$.
- (5) *Final Prediction Classification.* After training, the sample data x are input from test set data X into the model to obtain the prediction classification results and to select the final classification results of test set by voting mechanism. The principle of voting mechanism can be expressed by the following equation:

$$f_{\text{RF}}(x) = \arg \max_{i=1,2,\dots,c} \{I(f_i^{\text{tree}}(x) = i)\}. \quad (20)$$

- (6) *Evaluation Model.* The confusion matrix is established by the validation set of OOB data, and the classification results are evaluated. Based on confusion matrix, four evaluation indexes are selected, including overall accuracy (OA), Kappa coefficient, root mean square error (RMSE), and mean absolute error (MAE). The final prediction results of the RFID system identification rate are compared with the threshold condition of the system identification rate in the actual project. Therefore, it can be judged whether the hardware deployment scheme of the RFID system can meet the application requirements. The OA, Kappa coefficient, RMSE, and MAE are expressed by the following equations, respectively:

$$\text{Overall accuracy} = \frac{\sum_{i=1}^n X_{ii}}{N}, \quad (21)$$

$$\text{Kappa} = \frac{N \sum_{i=1}^n X_{ii} - \sum_{i=1}^n (X_{i+} \times X_{+i})}{N^2 - \sum_{i=1}^n (X_{i+} \times X_{+i})}, \quad (22)$$

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^n (Y_i - \hat{Y}_i)^2}{n}}, \quad (23)$$

$$\text{MAE} = \frac{1}{n} \sum_{i=1}^n |\hat{Y}_i - Y_i|. \quad (24)$$

4.3.3. Time Complexity Analysis. This paper proposes a neighborhood rough sets and random forest combination model of the identification rate of the RFID system prediction model. The essence is to reduce the dimension of X samples and Y initial influencing factors. Firstly, the initial influencing factors are reduced using neighborhood rough set theory combined with the principle of heuristic attribute reduction of neighborhood weighted dependence, thus obtaining a kernel factor subset. So, the dimension Y is reduced to V . At this time, the time complexity of calculating the kernel factor set of the neighborhood rough set is $O(V^2 X \log X)$. Then, the selected kernel factor subset is taken as the input of the random forest model to establish the RFID system identification rate prediction model. At this time, the time complexity of the model is $O(KVs (\log s)^2)$, where K represents the number of basic classifier CART and s represents the number of training sets in the random forest algorithm. It is obvious that the time complexity at this time is lower than that of the random forest directly dealing with the initial influencing factor. After all, the sample has been dimensionally reduced, $V \leq Y$.

5. Results and Discussion

5.1. Optimizing N-Tree and Constructing the Decision Tree. Before performing the random forest algorithm, it is necessary to optimize the super parameter N-Tree which is the number of decision trees. The 500 * 7-dimensional sample data are input variables of the random forest model. By changing the N-Tree value, the OOB precision corresponding to different N-Tree values can be calculated. The number of decision trees can be estimated by OOB precision, as shown in Figure 6.

From Figure 6, it can be seen that the value of N-Tree increases as the progress of model classification proceeds. When the value of N-Tree is greater than 500, the accuracy accordingly increases, to which the increasing tendency is not obvious but declines instead. Therefore, taking the model's identification classification accuracy and the classification time as reference standards, the final value of N-Tree is 500. Once the number of decision trees is determined, each tree is divided by the Gini coefficient expression and optimal splitting value given in Section 4.3.2 from the root node until each tree accomplishes growing. Here, we select a decision tree to observe its splitting and growing process. The optimal splitting influencing factor and optimal splitting value of each node in the splitting process of the decision tree are shown in Table 2. The complete construction process of the decision tree is drawn from the root node to the bottom, as shown in Figure 7.

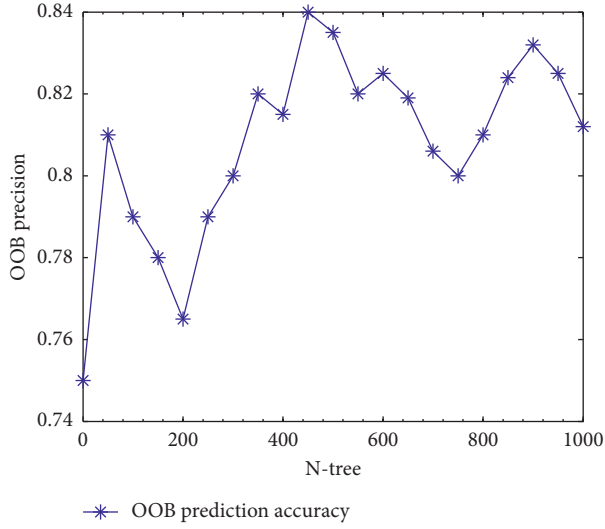


FIGURE 6: Relationship between N-tree and OOB.

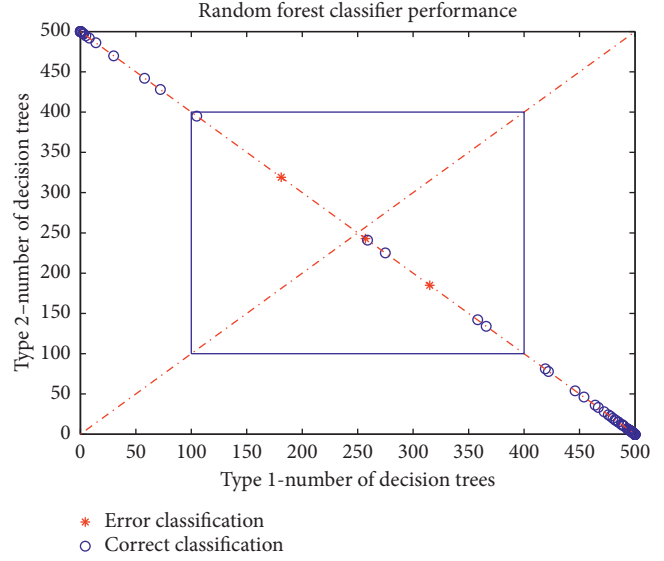


FIGURE 8: Predicting classification chart.

TABLE 2: Optimal splitting influencing factor and the optimal splitting value of nodes.

Optimal splitting influencing factor	Optimal splitting value
P_1	0.364
P_2	0.854
P_3	0.648
P_4	0.985
P_5	0.751
P_8	0.651
P_{11}	0.528

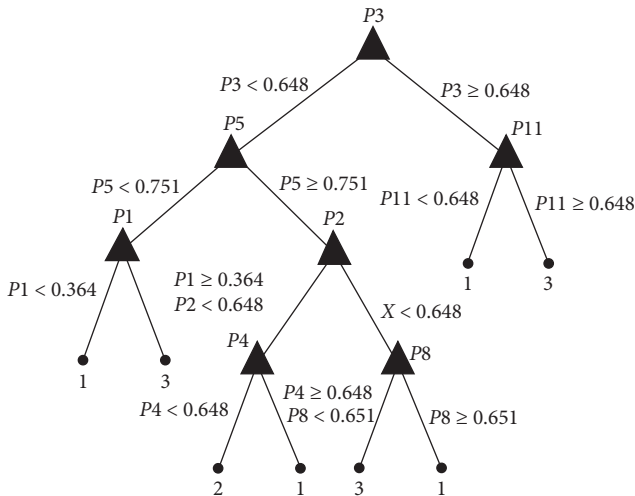


FIGURE 7: Decision tree structure.

5.2. Test Results. The test set is used to verify the classification accuracy of the constructed random forest model. The identification rate of the RFID system is obtained through simulation, as shown in Figure 8. According to the prediction results in Figure 8, the classification accuracy is

90.5%. The horizontal and vertical axis represent 500 groups of data, the red asterisk shape represents the error classification sample, and the blue circle denotes the correct classification sample. The data in the blue square represent the error-prone range, the closer the sample approaches 250 decision trees, the more difficult it is to make decisions, whereas the easier it is to make classification errors.

From the perspective of sample data, among the 500 sets of sample data, there are 357 sets of qualified samples that meet the threshold condition, which means the RFID system identification rate is higher than 95%. There are 143 sets of unqualified samples that do not meet the threshold condition; that said, the RFID system identification rate is less than 95%. Among 100 sets of data in the test set, there are 80 groups of qualified identification rate and the accurate prediction rate is about 96.25%, with an average misjudgment of 3 data groups. Among the unqualified identification rate, there are 20 groups with an accuracy rate of 90% and an average misjudgment of 2 data groups. At this time, the NRS-RF model exhibits excellent performance on predicting the RFID system identification rate.

In order to verify that the prediction accuracy can be improved through using a neighborhood rough set to reduce the initial influencing factor set, the relation between the prediction accuracy and the number of influencing factors is, therefore, analyzed adopting variable-controlling comparison experiment. First, accuracy verification was added to the test samples one by one according to the importance of influencing factors, as shown in Figure 9. It can be seen from Figure 9 that, under the condition of ensuring consistency of other parameters, the classification prediction accuracy was significantly improved as the number of influencing factors increases.

When the number of influencing factors reached 5, the overall prediction accuracy of the test sample increased slowly. When the number increased to 7, the accuracy reached 90.5% and then stabilized following with a small

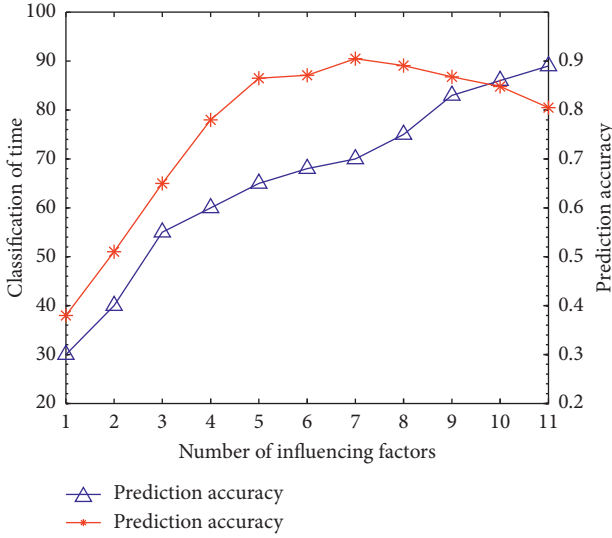


FIGURE 9: Influencing factors reduction verification.

decline. The classification time was on an upward trend as the number of influencing factors increased. When the influencing factors reached 11, the accuracy increased more significantly. Considering the combination of prediction accuracy and classification time, the 7 influencing factors obtained by attribute reduction are the optimal feature combination which effectively improve the classification accuracy while reducing the classification time.

In order to further verify the advantages of the NRS-RF model in predicting the RFID system identification rate, two new prediction algorithms are selected to construct the prediction network model and make a comparative analysis with the proposed NRS-RF model, including the K -nearest neighbor-naive Bayesian (KNN-NB) and backpropagation neural network (BPNN) [33–35]. These three models predict 300 groups of RFID system identification rate sample data, being compared in terms of OA, Kappa coefficient, RMSE, MAE, training time and prediction time, and correlation. The prediction results are shown in Table 3 and Figures 10–12.

The Kappa statistic is a measurement value for evaluating consistency, which indicates whether there is consistency between the predicted results of the model and the actual results. When the Kappa coefficient is greater than 0.75, it indicates that the model is better and has a certain value. It can be seen from Table 3 that the NRS-RF model with the highest classification accuracy has OA and Kappa coefficients of 88.5% and 0.875, which are much higher than those of other two models. It shows that the NRS-RF method can effectively screen out the nuclear factor set of system identification rate and improve the classification accuracy of the model.

Admittedly, featuring with excellent performance on applicability, the NRS-RF method can effectively eliminate the influencing factors affecting the RFID system identification rate. In Figure 10, the RMSE and MAE values of the NRS-RF model are relatively small, the prediction error is small, and the classification accuracy is higher. Comparing

the training set of the three models with the prediction set in terms of operating time, the NRS-RF model requires shorter time while ensuring lower computational complexity and higher calculation speed, thereby better satisfying engineering applications, as shown in Figure 11.

Different from KNN-NB and BPNN models, the random forest model is a kernel subset composed of 500 sample sets as the input of the model. However, due to its simple bifurcation structure of the base learner decision tree, its learning time is less than 6 s.

At the same time, when constructing the decision tree, it randomly selects part of the features as the classification basis of the tree growth, and when constructing the internal base learning device, it adopts the random sampling to put back the training samples, which ensures the generalization ability of the final model. Therefore, the random forest algorithm model has higher OA.

The correlation between the predicted value of the identification rate of the three models and the actual value is analyzed, as shown in Figure 12. The correlation coefficient R of the three models is $R_{\text{NRS-RF}} = 0.891$, $R_{\text{KNN-NB}} = 0.824$, and $R_{\text{BPNN}} = 0.798$, respectively. The R value of the NRS-RF model is closer to 1, which indicates that the prediction value of the model is closer to the actual measurement value, exhibiting better prediction effect.

The NRS-RF model compared with the other two kinds of prediction model shows great advantages, mainly because the initial influencing factors of redundant attributes is more, and they not only increase the classifier identification time but also make the classification accuracy of the RFID system significantly decreased. Through the reduction of NRS algorithm, the kernel factor subset is obtained. The classification of kernel factors set contains stronger characteristic sensitivity, improves the prediction precision, and reduces the computational complexity of the model.

The KNN-NB combination algorithm uses the KNN algorithm to calculate the distance between the sample data to be tested and the sample set. The selected sample data are used as the training sample of the NB algorithm, and then, the NB model is used for prediction and classification, where $K=3$. Because the NB model needs to know the prior probability, the prior probability often depends on the hypothetical model. However, there are many kinds of hypothetical models, so in some cases, the prediction effect will be poor due to the hypothetical prior model, so there is a certain error rate in classification decisions.

We use a three-layer BPNN model. The number of hidden layer nodes is set to 6, and the number of output layer nodes is 1. Since the BPNN model is essentially a gradient descent method, the objective function to be optimized is more complicated and prone to the “sawtooth phenomenon,” which makes the convergence speed of the BPNN model slow and affects the final prediction classification accuracy of the test set.

5.3. Engineering Application. As intelligent archives management inventory technology rapidly develops [36, 37], the UHF passive RFID technology liberates the archives

TABLE 3: Prediction accuracy comparison.

Prediction model	Sample data	Correct classification	Error classification	OA (%)	Kappa coefficient
NRS-RF	300	265	35	88.5	0.875
KNN-NB	300	254	46	84.7	0.805
BPNN	300	245	35	82.3	0.784

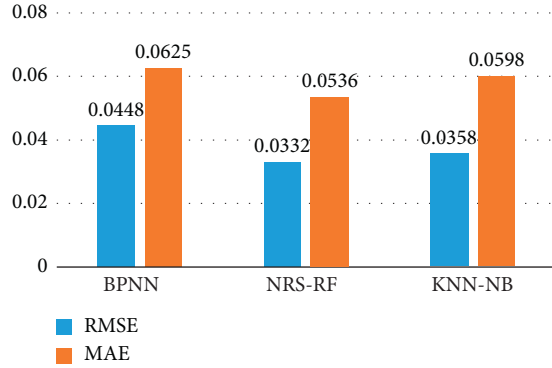


FIGURE 10: Prediction error.

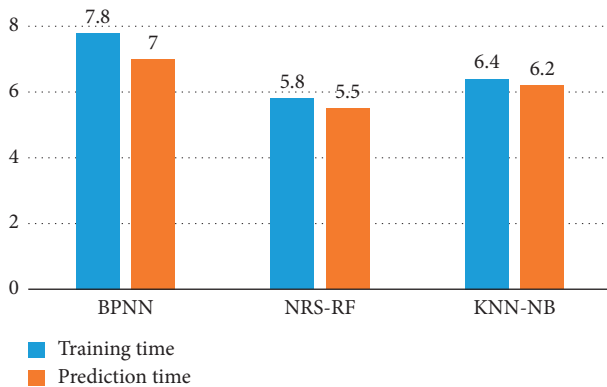


FIGURE 11: Comparison of training time and prediction time in different models.

management from the relatively “stereotyped” impression brought by barcodes. In order to further improve reading efficiency of the archive inventory, the RFID system is mounted on a mobile robotic vehicle, and the effectiveness of the NRS-RF model is verified in archives management applications. Under diverse hardware deployment conditions, this study selects 300 sets for the RFID system identification rate and for influencing factors as the sample data, on which heuristic attribute reduction of the initial influencing factors is conducted using neighborhood rough set theory. The importance distribution of influencing factors is given in this paper, which is obtained by neighborhood rough set reduction and by OOB error analysis of random forest.

The high importance score indicates that the influencing factor has greater impacts and contributions on classification results, as shown in Figure 13. It can be seen that the importance scores all exceed 5, involving antenna height $P1$, reader transmission power $P2$, distance between the tag and

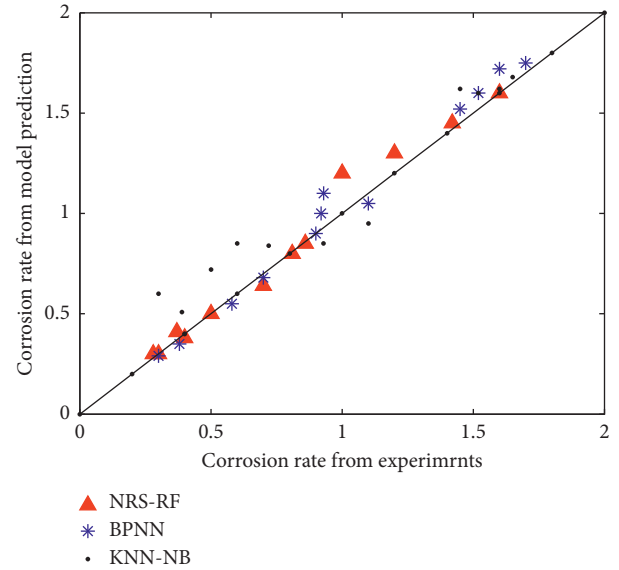


FIGURE 12: Correlation between the predicted value and actual value.

antenna $P3$, and other 4 influencing factors. Finally, the set of kernel factors is selected, which include antenna height $P1$, number of tags $P2$, distance between tags and antennas $P3$, reader polling cycles $P4$, antenna moving speed $P5$, antenna number $P8$, and reader transmitting power $P11$.

The average decline accuracy rate and average decline Gini coefficient of 7 groups of influencing factors are obtained through analyzing the Gini coefficient. Both the abovementioned average decline accuracy rate and average decline Gini coefficient can represent the degree of decline in accuracy when the influencing factor is replaced, both of which are positively correlated with the importance of the influencing factors. As shown in Figure 14, the larger the values of both the abovementioned rate and coefficient, the higher the importance of the influencing factor. In addition, some influencing samples of the RFID system identification rate are shown in Table 4.

From the 300 sample data, 240 groups were selected as training sets to train the model, and the remaining 60 groups were test set data for verification and prediction. Ten groups of test data were randomly selected from the test set samples, and scatter plots were made, as shown in Table 5 and Figure 15. The RMSE is 0.548, and the correlation coefficient R is 0.951, indicating outstanding prediction accuracy of the model.

It can be seen from Table 5 that the predicted classification level of 10 groups of test sample data is basically consistent with the actual classification, satisfying the engineering requirements. By comparing the third and fourth

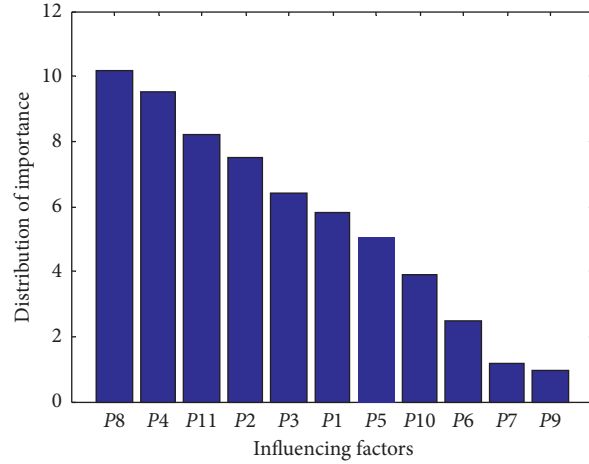


FIGURE 13: Importance distribution of influencing factors.

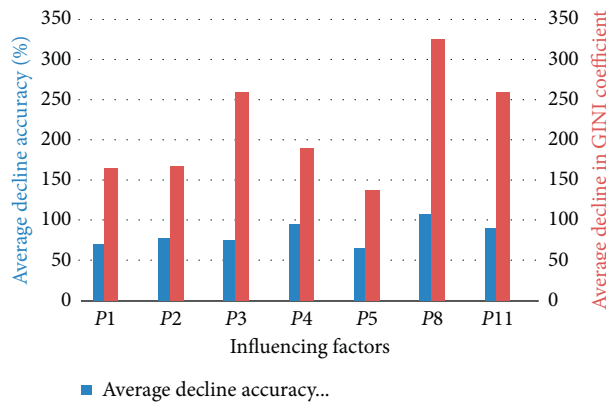


FIGURE 14: Average decline accuracy and average decline Gini coefficient.

TABLE 4: Samples of the RFID system identification rate.

Sample number	P1	P2	P3	P4	P5	P8	P11	RFID system identification rate level
1	0.6	90	0.8	1	0.3	2	18	1
2	0.9	150	1.5	2	0.6	1	23	2
3	0.6	90	1.9	3	0.9	2	28	1
4	1.2	130	0.8	3	0.9	1	18	1
5	0.9	110	1.5	2	0.3	2	23	1
6	1.2	180	1.9	1	0.6	1	28	2

TABLE 5: NRS-RF model RFID system identification rate level prediction.

Sample number	P1	P2	P3	P4	P5	P8	P11	Forecast level	Actual level
1	0.6	90	0.8	1	0.6	1	18	2	2
2	0.9	150	1.5	2	0.3	2	23	2	1
3	1.2	130	0.8	3	0.6	1	18	1	1
4	1.2	130	0.8	2	0.9	1	18	1	2
5	0.9	180	1.5	1	0.9	2	23	2	2
6	1.2	90	1.5	3	0.9	1	28	1	1
7	0.6	180	0.8	1	0.3	1	18	2	2
8	0.9	180	1.5	2	0.6	2	23	2	2
9	1.2	90	1.9	3	0.9	2	28	1	1
10	0.6	150	0.8	1	0.3	2	18	1	2

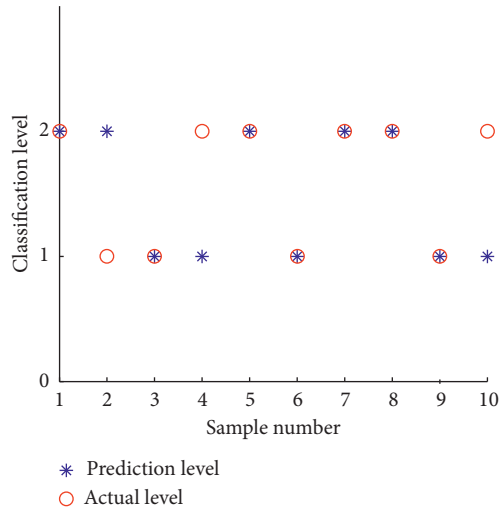


FIGURE 15: Test sample classification.

groups of samples in Table 5, it can be analyzed that the system identification rate can be improved to type 1 by reducing the moving speed of the antenna or increasing the number of reader polling cycles when other influencing factors are consistent. Comparative analysis of the second and fifth groups of samples in Table 5 shows that the system identification rate can be improved to type 1 when the number of tags is reduced or the number of reader polling cycles is increased, thereby accomplishing tag reading to the largest extent. The RFID system identification rate can, therefore, be predicted in a prompt and effective way by mining the relation between the identification rate and the influencing factors. Furthermore, the system identification rate can be improved by purposefully optimizing and adjusting the corresponding hardware deployment, through which the application requirements of more engineering inventories will be satisfied.

6. Conclusions

In order to optimize the hardware deployment of the RFID system and improve the system identification rate, a prediction model of the RFID system identification rate based on the combination model of neighborhood rough set and random forest is proposed through mining the relation between relevant influencing factors and the system identification rate. This study uses neighborhood rough set theory to conduct heuristic attribute reduction of weighted dependence of initial influencing factors and takes the kernel factor set as the input variable of random forest model for model training. The model is validated and verified in the RFID experimental test platform. Simulation results suggest that the fitting accuracy of the NRS-RF model is higher than that of the BPNN and other prediction models. Finally, the proposed model is applied to the RFID intelligent archives management platform, thus proving the excellent performance of the NRS-RF model. The proposed model can reversely configure the parameter setting of RFID hardware

deployment, and the system identification rate is, therefore, improved to satisfy the requirements of engineering applications.

Despite the abovementioned findings, the influence of antenna polarization mismatch, multipath fading, or other possible blind zone restrictions is not fully considered in the process of mobile robotic vehicle inventory tags, which may potentially affect the process of tag inventory. Our future research will be focusing on in-depth exploration of the automatic tag counting technology and on realizing the function of automatic tracking and path planning for mobile robots and robotic vehicles, paving way for the future development of automation in tag reading and writing.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This research was funded by the National Social Science Fund of Education Department of Shaanxi Province (no. 2018JK0704), the Science and Technology Plan Project of Xi'an (nos. 201805040YD18CG24-3 and 2019GX173), the Key Research and Development Plan of Shaanxi Province (nos. 2018ZDXM-GY-041 and 2018GY-150), the Science and Technology Research Plan Project of Xianyang (nos. 2018ZDXM-GY-041 and 2018GY-150), and the Team Project of Foshan Entrepreneurship and Innovation (2017IT100032).

References

- [1] K. Kapucu and C. Dehollain, "A passive UHF RFID system with a low-power capacitive sensor interface," in *Proceedings of the 2014 IEEE RFID Technology and Applications Conference (RFID-TA)*, Tampere, Finland, September 2014.
- [2] Y. L. Liu, S. B. Hu, and F. Y. Yan, "Analysis of key factors of embedded RFID system identification rate," *Journal of Engineering Design*, vol. 12, no. 45, pp. 479–484, 2012.
- [3] Y. Deng, Y. Zhu, Y. F. Yang, X. H. Cheng, and C. Q. Li, "Research on RFID indoor positioning algorithm based on BP neural network," *Miniature Microcomputer System*, vol. 40, no. 8, pp. 1707–1712, 2019.
- [4] L. Y. Qiao, J. G. Yang, H. Zhou, S. S. Dai, and P. C. Fu, "Optimization of key index of reading rate of RFID system based on rough set theory," *Manufacturing Automation*, vol. 36, no. 45, pp. 13–16, 2014.
- [5] M. A. Faraz, S. Yasir, M. Roberto, and C. Noel, "A comparative analysis of machine/deep learning models for parking space availability prediction," *Sensors*, vol. 20, no. 1, pp. 1–17, 2020.
- [6] N. T. Rafia, E. P. Alexander, and M. K. Jong, "Bearing fault diagnosis of induction motors using a genetic algorithm and machine learning classifiers," *Sensors*, vol. 20, no. 1, pp. 1504–1523, 2020.

- [7] K. Ashima, M. Rajni, N. Anand, K. Jeamin, G. K. Byeong, and C. A. Naveen, "Novel deep-learning-based bug severity classification technique using convolutional neural networks and random forest with boosting," *Sensors*, vol. 19, no. 3, pp. 1329–1351, 2019.
- [8] Y. Kang, M. Kim, and H. A. Lee, "A hierarchical structure based reader anti-collision protocol for dense RFID reader networks," in *Proceedings of the 13th International Conference on Advanced Communication Technology*, Seoul, Republic of Korea, December 2011.
- [9] S. Kwangcheol and S. Wonil, "RAC-multi: reader anti-collision algorithm for multichannel mobile RFID networks," *Sensors*, vol. 10, no. 1, pp. 1329–1351, 2010.
- [10] Z. Li, H. Yang, J. Li, C. He, and J. Zhou, "An enhanced neighbor-friendly reader anti-collision algorithm in mobile RFID networks," in *Proceedings of the 2014 4th IEEE International Conference on Information Science and Technology*, Shenzhen, China, April 2014.
- [11] N. Nguyen, B. Liu, and V. A. Pham, "Dynamic-range-based algorithm for reader-tag collision avoidance deployment in RFID networks," in *Proceedings of the 2016 International Conference on Electronics, Information, and Communications (ICEIC)*, Da Nang, Vietnam, January 2016.
- [12] V. Ferro, A. Luz, and A. Lucrécio, "Small long range UHF tag for metal applications," in *Proceedings of the 2013 IEEE International Conference on RFID-Technologies and Applications (RFID-TA)*, Johor Bahru, Malaysia, September 2013.
- [13] G. Cicirelli, A. Milella, and D. Di, "Supervised learning of RFID sensor model using a mobile robot," in *Proceedings of the 2011 IEEE International Conference on RFID-Technologies and Applications*, Sitges, Spain, September 2011.
- [14] J. Minhho, Y. Yonu, S. Cha, and H. Choo, "Mobile RFID tag detection influence factors and predict detectability," *Sensors*, vol. 9, no. 2, pp. 112–119, 2009.
- [15] Keonn, "Shop keonn UHF RFID products," 2020, <https://www.atlasrfidstore.com/keonn-rfid-readers-antennas/>.
- [16] J. Liu, W. Li, Z. Wen, S. Xiao, L. Yang, and F. Tan, "An energy efficiency approach via self-adaptive duty-cycle in social contacts scenario," in *Proceedings of the 25th International Conference on Parallel and Distributed Systems (ICPADS)*, Tianjin, China, December 2019.
- [17] H. Wang, S. Wang, J. Yao et al., "Effective anti-collision algorithms for RFID robots system," *Assembly Automation*, vol. 40, no. 1, pp. 55–64, 2019.
- [18] Impinj speedway connect software for Impinj RFID readers, <http://www.idshi.com/ProductView.aspx?Id=84>, 2020.
- [19] Z. H. Zhou, *Machine Learning*, pp. 121–146, Tsinghua University Press, Beijing, China, 2016.
- [20] Z. Deng, Z. Zheng, D. Deng, T. Wang, Y. He, and D. Zhang, "Feature selection for multi-label learning based on F-neighborhood rough sets," *IEEE Access*, vol. 8, no. 1, pp. 39678–39688, 2020.
- [21] C. Wang, Y. Huang, M. Shao, and X. Fan, "Fuzzy rough set-based attribute reduction using distance measures," *Knowledge-Based Systems*, vol. 164, no. 1, pp. 205–212, 2019.
- [22] C. Wang, Y. Wang, M. Shao, Y. Qian, and D. Chen, "Fuzzy rough attribute reduction for categorical data," *IEEE Transactions on Fuzzy Systems*, vol. 28, no. 5, pp. 818–830, 2019.
- [23] C. Wang, Y. Huang, M. Shao, Q. Hu, and D. Chen, "Feature selection based on neighborhood self-information," *IEEE Transactions on Cybernetics*, vol. 50, no. 9, pp. 4031–4042, 2020.
- [24] J. Zhao and X. Pan, "Remote sensing image feature selection based on rough set theory and multi-agent system," in *Proceedings of the 2015 12th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, Zhangjiajie, China, August 2015.
- [25] M. Ren, Y. Qu, and A. Deng, "Covering rough set-based three-way decision feature selection," in *Proceedings of the 2018 Tenth International Conference on Advanced Computational Intelligence (ICACI)*, Xiamen, China, March 2018.
- [26] V. K. Y. Wu and N. H. Vaidya, "RFID trees: a distributed RFID tag storage infrastructure for forest search and rescue," in *Proceedings of 7th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks (SECON)*, Boston, MA, USA, July 2010.
- [27] P. N. Samarakoon, N. E. P. Promayon, and C. Fouard, "Light random regression forests for automatic multi-organ localization in CT images," in *Proceedings of the 14th International Symposium on Biomedical Imaging (ISBI 2017)*, Melbourne, Australia, June 2017.
- [28] N. Spolaôr and M. C. Monard, "Evaluating ReliefF-based multi-label feature selection algorithm," *Advances in Artificial Intelligence—IBERAMIA 2014*, vol. 40, no. 1, pp. 194–205, 2014.
- [29] Y. Lin, Q. Hu, J. Liu, and J. Duan, "Multi-label feature selection based on max-dependency and min-redundancy," *Neurocomputing*, vol. 168, no. 1, pp. 92–103, 2015.
- [30] M. C. Yao and M. F. Chao, "A greedy algorithm for attribute reduction," *Journal of Jiamusi University (Natural Science Edition)*, vol. 6, no. 12, pp. 125–130, 2013.
- [31] L. I. Mina and C. W. Donga, "Exploration greedy algorithm with attribute reduction," *Computer Engineering*, vol. 38, no. 19, pp. 163–166, 2013.
- [32] Random forest toolbox, https://download.csdn.net/download/u011727648/10502334?utm_source=bbsseo, 2020.
- [33] B. Sanniv, K. Anwesa, B. Rohit, and D. N. Tibarewala, "Classification of lower limb motor imagery using K-nearest neighbor and naïve-bayesian classifier," in *Proceedings of the 3rd International Conference on Recent Advances in Information Technology (RAIT)*, Dhanbad, India, July 2016.
- [34] Y. Yu and L. Zhang, "WSN location method based on BP neural network in NLOS environment," in *Proceedings of the International Conference on Wireless Communication and Sensor Network*, Wuhan, China, December 2014.
- [35] D. Xu, Y. Wang, P. Peng, S. Beilun, Z. Deng, and H. Guo, "Real-time road traffic state prediction based on kernel-KNN," *Transportmetrica A: Transport Science*, vol. 16, no. 1, pp. 104–118, 2020.
- [36] Q. L. Shen, B. Shao, and L. J. Chen, "Design and implementation of book inventory robot based on UHF RFID," *Library Science Research*, vol. 7, no. 1, pp. 24–28, 2016.
- [37] N. Zhang and Y. L. Gou, "Application of new insect-proof RFID tags in smart archive management," *Lantai World*, vol. 7, no. 1, pp. 103–105, 2018.

Research Article

Distributed Integrated Sliding Mode-Based Nonlinear Vehicle Platoon Control with Quadratic Spacing Policy

Lei Zuo , Ye Zhang, Maode Yan, and Wenrui Ma

Department of Electrical and Control Engineering, Chang'an University, Xi'an 710064, China

Correspondence should be addressed to Lei Zuo; L_zuo@chd.edu.cn

Received 22 July 2020; Revised 11 November 2020; Accepted 9 December 2020; Published 28 December 2020

Academic Editor: Ning Wang

Copyright © 2020 Lei Zuo et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper investigates the nonlinear vehicle platoon control problems with external disturbances. The quadratic spacing policy (QSP) is applied into the platoon control, in which the desired intervehicle distance is a quadratic function in terms of the vehicle's velocities. Comparing with the general constant time headway policy (CTHP), the QSP is more suitable to the human driving behaviors (HDB) and can improve the traffic capacity. Then, a novel platoon control scheme is proposed based on the distributed integrated sliding mode (DISM). Since the external disturbances are taken into consideration, the sliding mode method is employed to handle the disturbances. Moreover, the stability and string stability of the proposed platoon control system are strictly analyzed. In final, numerical simulations are provided to verify the proposed approaches.

1. Introduction

In recent years, the vehicle platoon control has received substantially increasing interests in the intelligent transportation system [1–3] due to its significant advantages in reducing traffic jams, improving the traffic safety, and increasing traffic capacity [4–6]. Generally, the objective of the vehicle platoon control is to drive a group of vehicles into a desired platoon from arbitrary initial positions. From this definition, we can find that the dynamics of vehicle, the communication topology, and the spacing policy play important roles in the vehicle platoon control.

To reach the desired platoon, many control theories have been applied to vehicle driving, including the consensus control [7], adaptive control [8–10], model predictive control [11], and the sliding mode control [12–17]. For example, a distributed consensus strategy with second-order dynamics is proposed to achieve the platooning of vehicles in [7], where the actuator saturation and absent velocity measurement are considered. In [8], the vehicle platoon control with velocity constraints, input saturations, and unknown driving resistances are handled by the adaptive neural sliding mode control techniques. By using the model predictive control method, a novel vehicle platoon control

scheme with multiple constraints and communication delays is proposed in [11]. While considering the state of the leader and the braking force, a distributed integrated sliding mode-based platoon control algorithm is given in [12]. In these existing results, the sliding mode control method has attracted increasing interests due to its significant advantages in dealing with the external disturbances. For instance, the neuroadaptive quantized PID sliding mode control method for heterogeneous vehicle platoon is presented with external disturbances and unknown actuator dead-zone in [13]. The Pontryagin's minimum principle (PMP) based set-point optimization and sliding mode control law are proposed for vehicle platoon in [14]. In [15], a distributed adaptive integrated sliding mode control method is developed to show the stability and string stability of the proposed vehicle platoon control system. Although these existing sliding mode control methods have great advantages in vehicle platoon control, the above studies mostly use the linear dynamics, which greatly limits their applications. Hence, it is necessary to further investigate the sliding mode-based platoon control with nonlinear vehicles.

In addition, another key point of vehicle platoon is to select a reasonable intervehicle spacing strategy. It can improve the traffic capacity and reduce the road pressure [18].

Generally, there are two strategies for the desired platoon: the constant-spacing policy (CSP) and the variable-spacing policy (VSP). The differences between these spacing policies are the evolution policies of the desired distance between any adjacent vehicles [19]. Comparing with the CSP, the desired distances in VSP are usually time-varying and related to the vehicle's velocities. In detail, the constant time headway policy (CTHP) and the quadratic spacing policy (QSP) are two classical spacing policies in VSP. Particularly, the desired distance in QSP is a quadratic function in terms of the vehicle's velocities. It can meet the requirement of human driving behaviors (HDB). For example, the vehicle platoon control with QSP is proposed in [20], where the traffic flow stability and energy consumption are considered. In [21], a novel platoon control scheme with QSP is presented for the heavy trucks in mixed traffic situations, in which the fuel economy and pollution are taken into consideration. While considering the vehicle characteristics and road conditions, a vehicle platoon control algorithm with novel QSP is proposed for the longitudinal spacing control in [22]. Although the QSP shows great advantages in vehicle platoon control, there are few results applying the QSP into the vehicle platoon control with nonlinear dynamics.

Motivated by this fact, a distributed integrated sliding mode (DISM) based vehicle platoon control strategy with both QSP and nonlinear dynamics is proposed. The finite-time stability and string stability of the proposed vehicle platoon control system are strictly analyzed. In detail, a third-order nonlinear dynamics with external disturbances is applied to describe the vehicles. Then, the vehicle platoon with QSP is presented to maintain the desired intervehicle spacing. Since the QSP is a quadratic function in terms of the vehicle's velocities, the characteristic of this policy is more practical than the existing results with CTHP and CSP [23]. Moreover, the finite-time stability for each vehicle is shown by using the Lyapunov stability theory, and the string stability of the

proposed vehicle platoon control system is demonstrated based on the coupled sliding mode method. In addition, an approximation function is employed to overcome the chattering in the sliding mode control.

The remainder of this paper is presented as follows. In Section 2, the preliminaries and problem formulation are presented. Then, the DISM-based vehicle platoon control with nonlinear dynamics and QSP is proposed, and the stability of proposed platoon control system is analyzed in Section 3. To illustrate the proposed approaches, numerical simulations are presented in Section 4. In final, Section 5 draws the conclusion and describes the future work.

Notations: throughout this paper, (1) \mathbb{R} and \mathbb{R}^+ represent the set of all real numbers and the set of all nonnegative real numbers, respectively. (2) $\|\cdot\|$ stands for the Euclidean norm of a vector. (3) $|\cdot|$ denotes the absolute value of real numbers. (4) The $\text{sgn}(\cdot)$ denotes the signum function.

2. Preliminaries and Problem Formulation

2.1. Vehicle Dynamics. Consider a group of nonlinear vehicles, which includes one leader and N followers. Index these vehicles as $0, 1, \dots, N$, where 0 denotes the leader, and $1, \dots, N$ are the followers. The dynamic model of each vehicle is shown as

$$\begin{cases} \dot{r}_i(t) = v_i(t), \\ \dot{v}_i(t) = a_i(t), \\ \dot{a}_i(t) = f_i(v_i(t), a_i(t)) + g_i(v_i(t))b_i(t) + \omega_i(t), \end{cases} \quad (1)$$

where $a_i(t)$, $v_i(t)$, and $r_i(t)$ are the acceleration, velocity, and position of the i^{th} vehicle ($i = 0, 1, 2, \dots, N$), respectively; $\omega_i(t)$ is the external disturbance of the i^{th} vehicle, and $|\omega_i(t)| \leq \Omega$; $b_i(t)$ is the engine control input; $f_i(v_i(t), a_i(t))$, and $g_i(v_i(t))$ are the nonlinear terms and are given as

$$\begin{cases} f_i(v_i(t), a_i(t)) = -\frac{1}{\zeta_i} \left(a_i(t) + \frac{\rho C_i d_{ci} v_i^2(t)}{2m_i} + \frac{d_{mi}}{m_i} \right) - \frac{\rho C_i d_{ci} v_i(t) a_i(t)}{m_i}, \\ g_i(v_i(t)) = \frac{1}{m_i \zeta_i}, \end{cases} \quad (2)$$

where m_i is the mass of the i^{th} vehicle; ρ is the mass density of the air; d_{ci} is the coefficient of drag; d_{mi} is the mechanical resistance; C_i is the cross-section area of the i^{th} vehicle; and ζ_i is the engine time.

Then, in order to solve the nonlinear vehicle dynamics of (1), a feedback linearization controller is introduced as follows:

$$b_i(t) = u_i(t)m_i + \frac{\rho C_i d_{ci} v_i^2(t)}{2} + d_{mi} + \zeta_i \rho C_i d_{ci} v_i(t) a_i(t). \quad (3)$$

Substituting (2) and (3) into (1), the model of the i^{th} vehicle is described as

$$\begin{cases} \dot{r}_i(t) = v_i(t), \\ \dot{v}_i(t) = a_i(t), \\ \dot{a}_i(t) = -\frac{a_i(t)}{\zeta_i} + \frac{u_i(t)}{\zeta_i} + \omega_i(t), \end{cases} \quad (4)$$

where $u_i(t)$ denotes the control input for the i^{th} vehicle after linearization.

Remark 1. We use the real nonlinear vehicle dynamic model in this paper, and a feedback linearization method is deployed to transform the vehicle dynamic model into a third-order linear system. Comparing with the existing results, it is more similar to the real dynamics of a vehicle and more practical significance in vehicle platoon control.

For the communication topology, we have the following assumption.

Assumption 1. Suppose the communication topology among the vehicles is bidirectional. That is, each vehicle can only communicate with its adjacent neighbors.

2.2. Intervehicle Spacing Policy. The intervehicle spacing policy plays an important role in vehicle platoon control. For the intervehicle spacing policy, we usually select a proper function to construct the desired distance among the vehicles. The QSP is a classical spacing policy in vehicle platoon control. It is a quadratic function in terms of the vehicle's velocities, which is greatly suitable to the human driving behavior (HDB, $d_{i,i-1} = A + TV + GV^2$). The detailed QSP is shown by

$$d_{i,i-1} = p_0 v_i^2(t) + p_1 v_i(t) + x, \quad (5)$$

where $d_{i,i-1}$ denotes the desired distance between two adjacent vehicles; x is the standstill spacing; and p_0 is the designed positive parameter and is given as $p_0 = (\eta / (2a_{\max}))$, with η being the safety factor depending on the road or weather, and a_{\max} denoting the absolute value of the maximum possible deceleration. p_1 is the time delay in platoon systems, which compensates for the delay in braking or acceleration (about 10 ms–80 ms [23]).

Remark 2. Generally, the intervehicle spacing policy is closely related to the parameters p_0 and p_1 . When $p_0, p_1 > 0$, the intervehicle spacing policy is QSP. In contrast, when $p_0 = 0, p_1 > 0$ and $p_0 = 0, p_1 = 0$, the intervehicle spacing policy reduces to a CTHP and a CSP, respectively, which have been studied extensively in vehicle platoon. Hence, comparing to the CTHP and CSP, the QSP is more general and practical. For the CSP, supposed that the distance of the adjacent vehicles maintained the same constant spacing, which may increase the traffic jams, the CTHP is also required to maintain a constant time headway parameter. Comparing with the general CSP and CTHP, the nonlinear QSP can be regarded as the VTHP to adjust the time headway and increase traffic capacity, which can also guarantee string stability and traffic flow stability. Hence, the vehicle platoon control with QSP is more practical in applications.

2.3. Problem Formulation. Based on the QSP, the spacing error $e_i(t)$ can be written as

$$e_i(t) = \delta_i(t) - x - p_0 v_i^2(t) - p_1 v_i(t), \quad (6)$$

where $\delta_i(t) = r_{i-1}(t) - r_i(t) - l$ denotes the distance between any two adjacent vehicles; l is the length of each vehicle.

On this basis, we intend to design a DISM-based vehicle platoon control scheme with nonlinear dynamics and QSP. This paper has to meet the following requirements:

- (1) Stability: the finite-time stability of each vehicle is guaranteed
- (2) String stability [24]: the transient errors are not enlarging with vehicle index due to any maneuver of the lead vehicle, if the error propagation transfer function $G_i(s)$ satisfies

$$|G_i(s)| = \left| \frac{E_{i+1}(s)}{E_i(s)} \right| \leq 1, \quad \text{for } i = 1, \dots, N, \quad (7)$$

where $E_i(s)$ denotes the Laplace transform of $e_i(t)$

- (3) For each vehicle, it must satisfy that

$$v_i(t) \longrightarrow v_0(t),$$

$$\delta_i(t) \longrightarrow x + p_0 v_0^2(t) + p_1 v_0(t), \quad \text{for } i = 1, \dots, N. \quad (8)$$

3. The DISM-Based Platoon Control Scheme

In this section, a novel DISM-based vehicle platoon control scheme is proposed for a group of nonlinear vehicles with QSP. The finite-time stability of each vehicle and string stability of the proposed platoon control system are strictly analyzed.

First, the integrated sliding mode surface for the i^{th} vehicle is shown as

$$s_i(t) = \dot{e}_i(t) + \alpha_1 e_i(t) + \alpha_2 \int_0^t e_i(\tau) d\tau, \quad (9)$$

where α_1 and α_2 are the positive constants.

According to Hurwitz stability theory, α_1 and α_2 are required to make all the eigenvalues of $p^2 + \alpha_1 p + \alpha_2 = 0$ contain a real-negative part. In this case, let $\alpha_1 = 2\epsilon$ and $\alpha_2 = \epsilon^2$, where ϵ is a positive constant [25].

Since the sliding mode surface in (9) cannot guarantee the string stability of the vehicle platoon, an improved coupled sliding mode surface is provided as follows:

$$S_i(t) = \begin{cases} s_{i+1}(t) - \beta s_i(t), & i = 1, \dots, N-1, \\ -\beta s_i(t), & i = N, \end{cases} \quad (10)$$

where β is the parameter of coupled intensity satisfying $0 < \beta \leq 1$. The relationship between $S_i(t)$ and $s_i(t)$ is shown as

$$S(t) = Bs(t), \quad (11)$$

where $s(t) = [s_1(t) \ s_2(t) \ \dots \ s_N(t)]^T$, $S(t) = [S_1(t) \ S_2(t) \ \dots \ S_N(t)]^T$, and

$$B = \begin{bmatrix} -\beta & 1 & \dots & 0 & 0 \\ 0 & -\beta & 1 & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \dots & -\beta & 1 \\ 0 & 0 & \dots & 0 & -\beta \end{bmatrix}. \quad (12)$$

To further show the relationship between s_i and S_i , we have the following lemmas.

Lemma 1. *Equivalence of the convergence of the CSS and each sliding surface toward zero [26]: S_i becomes zero for all $i = 1, \dots, N$, if and only if s_i becomes zero.*

According to (6), (9), and (10), \dot{S}_i is presented as

$$\begin{aligned} \dot{S}_i(t) &= \dot{s}_{i+1} - \beta \dot{s}_i, \quad i = 1, 2, \dots, N-1, \\ &= [\ddot{e}_{i+1}(t) + \alpha_1 \dot{e}_{i+1} + \alpha_2 e_{i+1}(t)] - \beta [\ddot{e}_i(t) + \alpha_1 \dot{e}_i(t) + \alpha_2 e_i(t)] \\ &= -\beta \left([a_{i-1}(t) - a_i(t)] - \phi_i(t) \left[-\frac{a_i(t)}{\zeta_i} + \frac{u_i(t)}{\zeta_i} + \omega_i(t) \right] - 2p_0 a_i^2(t) \right) + D_i(t) \\ &= \beta \phi_i(t) \left[-\frac{a_i(t)}{\zeta_i} + \frac{u_i(t)}{\zeta_i} + \omega_i(t) \right] - \beta [a_{i-1}(t) - a_i(t)] + 2\beta p_0 a_i^2(t) + D_i(t), \end{aligned} \quad (13)$$

where $\phi_i(t) \triangleq 2p_0 v_i(t) + p_1$, and $D_i(t) = [\ddot{e}_{i+1}(t) + \alpha_1 \dot{e}_{i+1}(t) + \alpha_2 e_{i+1}(t)] - \beta [\alpha_1 \dot{e}_i(t) + \alpha_2 e_i(t)]$.

For $i = N$, we obtain that

$$\begin{aligned} \dot{S}_N(t) &= -\beta \dot{s}_N \\ &= -\beta [\ddot{e}_N(t) + \alpha_1 \dot{e}_N(t) + \alpha_2 e_N(t)] \\ &= -\beta \left([a_{N-1}(t) - a_N(t)] - \phi_N(t) \left[-\frac{a_N(t)}{\zeta_N} + \frac{u_N(t)}{\zeta_N} + \omega_N(t) \right] - 2p_0 a_N^2(t) \right) + D_N(t) \\ &= \beta \phi_N(t) \left[-\frac{a_N(t)}{\zeta_N} + \frac{u_N(t)}{\zeta_N} + \omega_N(t) \right] - \beta [a_{N-1}(t) - a_N(t)] + 2\beta p_0 a_N^2(t) + D_N(t), \end{aligned} \quad (14)$$

where $\phi_N(t) \triangleq 2p_0 v_N(t) + p_1$, and $D_N(t) = -\beta [\alpha_1 \dot{e}_N(t) + \alpha_2 e_N(t)]$.

Then, the following two lemmas are provided to show the finite-time stability and string stability.

Lemma 2. (Barbalat Lemma [27]). *If $\phi(t): \mathbb{R} \rightarrow \mathbb{R}^+$ is a uniformly continuous function for $t \geq 0$ and the limit of the integral $\lim_{t \rightarrow \infty} \int_0^t \phi(\tau) d\tau$ exists and is finite, then*

$$\lim_{t \rightarrow \infty} \phi(t) = 0. \quad (15)$$

Lemma 3. (Finite-Time Stability [28]). *Suppose there is a positive definite Lyapunov function $V(x, t)$ defined on $U \times \mathbb{R}^+$, where U is the neighborhood of the origin. There are positive real constants $c > 0$ and $a \in (0, 1)$, such that $\dot{V}(x, t) + cV^a(x, t)$ is negative semidefinite on U . Then, $V(x, t)$ is locally finite-time convergent. In addition, the settling time T satisfies $T \leq (V^{1-a}(x_0, t))/(c(1-a))$ for any given initial condition $x(t_0)$ in the neighborhood of the origin in U .*

Then, we have the following DISM-based vehicle platoon control schemes.

$$u_i(t) = -\frac{1}{\beta\phi_i(t)} \left\{ -\beta\zeta_i [a_{i-1}(t) - a_i(t)] + \zeta_i D_i(t) + 2\beta\zeta_i p_0 a_i^2(t) + \zeta_i \gamma \operatorname{sgn}(S_i(t)) \right\} + a_i(t), \quad i = 1, 2, \dots, N, \quad (16)$$

with

$$D_i(t) = \begin{cases} [\ddot{e}_{i+1}(t) + \alpha_1 \dot{e}_{i+1}(t) + \alpha_2 e_{i+1}(t)] - \beta[\alpha_1 \dot{e}_i(t) + \alpha_2 e_i(t)], & i = 1, \dots, N-1, \\ -\beta[\alpha_1 \dot{e}_N(t) + \alpha_2 e_N(t)], & i = N, \end{cases} \quad (17)$$

where $\phi_i(t) \triangleq 2p_0 v_i(t) + p_1$; according to the velocity of each vehicle bound, $|\phi_i(t)| \leq \Phi$. γ is the positive design parameter of the controller and satisfies that $\gamma > \beta\Omega\Phi$.

To formally show the finite-time stability of each vehicle and string stability of the whole vehicle platoon, the following theorems are proposed.

Theorem 1. *Consider a vehicle platoon system described by (4). The communication topology among vehicles is bidirectional, and the QSP is shown in (5). Then, by using the proposed platoon control law in (16), the finite-time stability of each vehicle can be guaranteed.*

Proof. Consider the following Lyapunov function candidate:

$$V(t) = \sum_{i=1}^n V(t)_i = \sum_{i=1}^n \frac{1}{2} S_i^2(t). \quad (18)$$

Then, taking the derivation of $V_i(t)$, we obtain that

$$\dot{V}(t) = \sum_{i=1}^n S_i(t) \dot{S}_i(t). \quad (19)$$

Substituting (13)–(16) into (19), it follows

$$\begin{aligned} \dot{V}(t) &= \sum_{i=1}^n \left\{ S_i(t) \left[\beta\phi_i(t) \left(-\frac{a_i(t)}{\zeta_i} + \frac{u_i(t)}{\zeta_i} + \omega_i(t) \right) \right. \right. \\ &\quad \left. \left. - \beta(a_{i-1}(t) - a_i(t)) + 2\beta p_0 a_i^2(t) + D_i(t) \right] \right\} \\ &= \sum_{i=1}^n \{ S_i(t) [-\gamma \operatorname{sgn}(S_i(t)) + \beta\phi_i(t) \omega_i(t)] \} \\ &= \sum_{i=1}^n \{ -\gamma \operatorname{sgn}(S_i(t)) S_i(t) + \beta\phi_i(t) \omega_i(t) S_i(t) \} \\ &\leq \sum_{i=1}^n \{ -\gamma |S_i(t)| + \beta\Phi\Omega |S_i(t)| \} \\ &= \sum_{i=1}^n \{ -[\gamma - \beta\Phi\Omega] |S_i(t)| \} \leq 0. \end{aligned} \quad (20)$$

Denoting $\varphi(t) \triangleq [\gamma - \beta\Phi\Omega] \sum_{i=1}^n |S_i(t)|$ and integrating it from 0 to t , we have that

$$V(0) - V(t) \geq \int_0^t \varphi(\tau) d\tau. \quad (21)$$

Since $\dot{V}(t) \leq 0$, we have that $V(0) - V(t) \geq 0$ is positive and bounded if $V(0)$ is bounded. Then, according to Lemma 2, we obtain that

$$\lim_{t \rightarrow \infty} \varphi(t) = \lim_{t \rightarrow \infty} [\gamma - \beta\Phi\Omega] \sum_{i=1}^n |S_i(t)| = 0. \quad (22)$$

As $\gamma - \beta\Phi\Omega > 0$, it follows that $\lim_{t \rightarrow \infty} S_i(t) = 0$. Then, according to (9), (11), and Lemma 1, $s_i(t)$ and $e_i(t)$ would converge to zeros. Furthermore, as $\gamma - \beta\Phi\Omega \geq \bar{\gamma} > 0$, (20) can also be written as

$$\dot{V}(t) \leq -[\gamma - \beta\Phi\Omega] \sum_{i=1}^n |S_i(t)| \leq -\bar{\gamma} \sqrt{2V(t)}, \quad (23)$$

and equivalently $\dot{V}(t) + \bar{\gamma} \sqrt{2V(t)} \leq 0$, where $\sqrt{2V(t)}$ satisfying $\sqrt{2V(t)} \leq \sqrt{2V(0)} - \bar{\gamma}t$. Then, we can get that $V(t) = 0$ and equivalently $S_i(t) = 0$ when $T \geq (\sqrt{2V(0)}/\bar{\gamma})$ by Lemma 3. It indicates that $s_i(t)$ and spacing error $e_i(t)$ converge to zero in a finite time. On this basis, the finite-time stability of each vehicle is guaranteed.

Theorem 2. *Since $0 < \beta \leq 1$ and $S_i(t) \rightarrow 0$ in finite time, then the string stability of whole vehicle platoon system is guaranteed.*

Proof. Since $S_i(t) \rightarrow 0$ in a finite time, $s_{i+1}(t) - \beta s_i(t) = 0$. Then, we can obtain that

$$\beta \left(\dot{e}_i(t) + \alpha_1 e_i(t) + \alpha_2 \int_0^t e_i(\tau) d\tau \right) = \dot{e}_{i+1}(t) + \alpha_1 e_{i+1}(t) + \alpha_2 \int_0^t e_{i+1}(\tau) d\tau. \quad (24)$$

Since $\dot{e}_i(0) = 0$, $e_i(0) = 0$, and $e_i^{-1} \triangleq \int_{-\infty}^0 e_i(t)dt = 0$, take the Laplace transform of (24) and obtain that

$$\beta \left(s + \alpha_1 + \frac{\alpha_2}{s} \right) E_i(s) = \left(s + \alpha_1 + \frac{\alpha_2}{s} \right) E_{i+1}(s). \quad (25)$$

According to the condition $0 < \beta \leq 1$, we obtain that

$$|G_i(s)| = \left| \frac{E_{i+1}(s)}{E_i(s)} \right| = |\beta| \leq 1. \quad (26)$$

From (26), the string stability of the vehicle platoon can be achieved. \square

$$u_i(t) = \frac{1}{\beta \phi_i(t)} \left\{ -\beta \zeta_i [a_{i-1}(t) - a_i(t)] + \zeta_i D_i(t) + 2\beta p_0 a_i^2(t) \zeta_i \right\} + \frac{\zeta_i \gamma}{\beta \phi_i(t)} \frac{S_i(t)}{|S_i(t)| + \sigma} + a_i(t), \quad i = 1, 2, \dots, N. \quad (27)$$

Based on the above theorems and analyses, the following algorithm is provided to illustrate the proposed vehicle platoon scheme.

In Algorithm 1, it is worth noting that the initial spacing errors and the initial velocity errors of all vehicles are required to be zero, simultaneously. Furthermore, to facilitate the adjustment of parameters p_0 and p_1 , it is better to choose some small initial velocities for all the vehicles. Since the variable range, the time headway is limited [20], the parameters p_0 and p_1 are also limited by $0 < p_0, p_1 \leq 1$. Then, in order to guarantee the string stability, the β must satisfy that $0 < \beta \leq 1$.

4. Numerical Simulation

In this section, the numerical simulations are provided for a platoon with 4 followers and one leader to verify the proposed approaches.

The initial states of the leader are set as $r_0 = 0$ m and $v_0 = 2$ m/s, and the evolution of leader's velocity is shown as

$$v_0(t) = \begin{cases} 2 \text{ m/s}, & 0 \text{ s} < t \leq 3 \text{ s}, \\ 2 + 2(t - 3) \text{ m/s}, & 3 \text{ s} < t \leq 5 \text{ s}, \\ 6 \text{ m/s}, & 5 \text{ s} < t \leq 10 \text{ s}, \\ 6 - 2(t - 10) \text{ m/s}, & 10 \text{ s} < t \leq 12 \text{ s}, \\ 2 \text{ m/s}, & 12 \text{ s} < t \leq 18 \text{ s}, \\ 2 + (t - 18) \text{ m/s}, & 18 \text{ s} < t \leq 20 \text{ s}, \\ 4 \text{ m/s}, & 20 \text{ s} < t \leq 25 \text{ s}, \\ 4 - (t - 25) \text{ m/s}, & 25 \text{ s} < t \leq 27 \text{ s}, \\ 2 \text{ m/s}, & 27 \text{ s} < t \leq 60 \text{ s}. \end{cases} \quad (28)$$

The initial positions and initial velocities of the followers are set as $x(0) = [-24.76, -49.52, -74.28, -99.04]$ and $v(0) = [2, 2, 2, 2]$, respectively. All controller's parameters are set as $\gamma = 1.5$, $\beta = 0.6$, $\alpha_1 = 2$, $\alpha_2 = 1$, $p_0 = 0.155$, $p_1 = 0.07$, $\sigma = 0.02$, and $\zeta = 0.3$. Then, the standstill distance is given as $x = 18$ m, and the length of each vehicle is $l = 6$ m.

Remark 3. Since we have employed the signum function $\text{sgn}(S_i(t))$ in (16), it may cause chattering in practical applications. In this case, a sigmoid-like function $(S_i(t)/(|S_i(t)| + \sigma))$ is employed to eliminate this chattering. In addition, the σ in this sigmoid-like function should be small enough. It is because when a large σ is chosen, it would lead to a less accurate control result. Thus, we should trade-off the relationship between the value of σ and the control accuracy in practical applications.

Invoking the sigmoid-like function $(S_i(t)/(|S_i(t)| + \sigma))$, $u_i(t)$ can be rewritten as

The external disturbance of the i^{th} vehicle is given by $w_i(t) = 0.003 \sin(2\pi t)$.

On this basis, the simulation results of proposed vehicle platoon control scheme are shown as follows.

Figure 1 illustrates the positions and velocities of all vehicles. As shown in Figure 1, there is no collision between any two adjacent vehicles during the steady-state condition and the initial transient. Then, followers track the velocity of lead vehicle, in which the velocities of followers converge to the velocity of leader (2 m/s). The convergence time of the velocities is about 35 s.

The distances and spacing errors of the platoon with QSP are illustrated in Figure 2. As shown in this figure, the distances of all adjacent vehicles converge a constant value (18.5 m). Meanwhile, the spacing errors converge to zero in a finite time, and the string stability of the vehicle platoon is guaranteed. From Figures 1 and 2, it can be obtained that all vehicles satisfy the control objective in (8). The proposed vehicle platoon system is stable and reaches the desired formation under the DISM control scheme with QSP.

The sliding mode surface is shown in Figure 3. From Figure 3, it can be observed that the $S_i(t)$ reaches the sliding mode surface $S_i(t) = 0$ in a finite time and there is no chattering.

The distances and spacing errors of the platoon with CTHP are presented in Figure 4. From Figure 4, it can be seen that the distance between any two adjacent vehicles converges to a constant value (20 m) and the spacing errors converge to zero in a finite time. Then, in order to show the advantage of the vehicle platoon control with QSP, we compared our proposed approaches in Figure 3 with the vehicle platoon with CTHP in Figure 4; the platoon with QSP has smaller intervehicle distance than the platoon with CTHP. Apparently, the platoon with QSP can provide greater traffic capacity. Hence, the platoon with QSP is superior to the platoon with CTHP. The simulation results validate the proposed approaches.

According to the vehicle platoon performance index in [29], Table 1 shows the simulation results of different spacing policies in tracking performance, fuel economy, and ride

Input: the initial position, velocity, acceleration of each vehicle;
The controller parameters;

Output: the spacing error of the any two adjacent vehicles converges to zero;

- (1) Initialization: the initial positions, velocities, accelerations of a lead vehicle and 4 followers: $r_i(0), v_i(0), a_i(0)$; Controller parameters: $p_0, p_1, x, l, \beta, \alpha_1, \alpha_2, \zeta_i, \gamma$;
- (2) **for** $t = t_0: \Delta t: t_f$ **do**
- (3) calculate $e_i(t), \dot{e}_i(t), \ddot{e}_i(t)$ by using (6);
- (4) construct integrated sliding surface $s_i(t)$ by employing (9);
- (5) calculate $S_i(t)$ and D_i based on (10), (13), (14);
- (6) according to (16), calculate $u_i(t)$;
- (7) update the acceleration, velocity, and position information of the i^{th} vehicle respectively by using (4);
- (8) **end for**

ALGORITHM 1: The DISM-based platoon control algorithm with QSP.

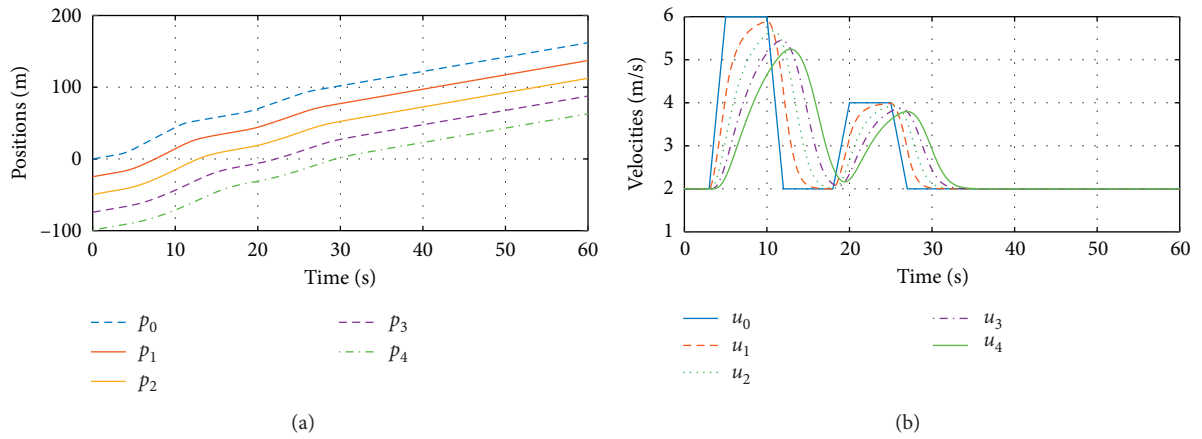


FIGURE 1: Positions and velocities of vehicles.

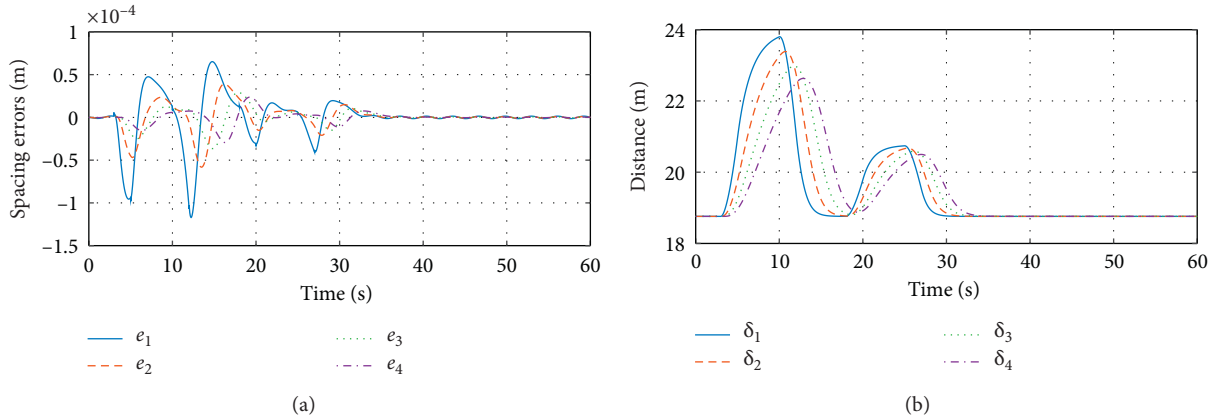


FIGURE 2: Spacing errors and distances of the platoon.

comfort. As shown in Table 1, the fuel economy and ride comfort of the platoon with the QSP are better, and the platoon with the CTHP has better tracking performance. In detail, the energy consumption and comfort ride of the vehicle platoon is mainly caused by unnecessary braking or

acceleration. Comparing with the CTHP, the QSP can be regarded as the variable time headway spacing policy; thus, the energy consumption value of the vehicle platoon with QSP is smaller. Therefore, the QSP can reduce the fuel consumption and enhance ride comfort.

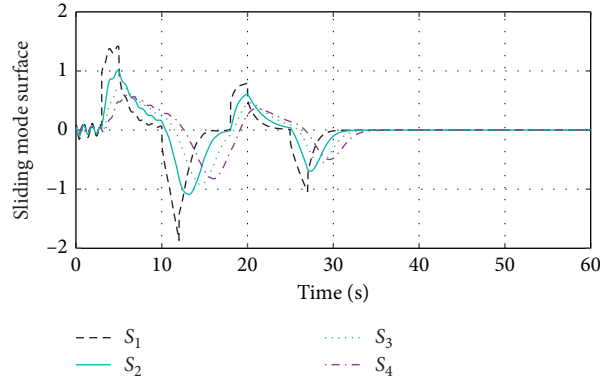


FIGURE 3: Sliding mode surface.

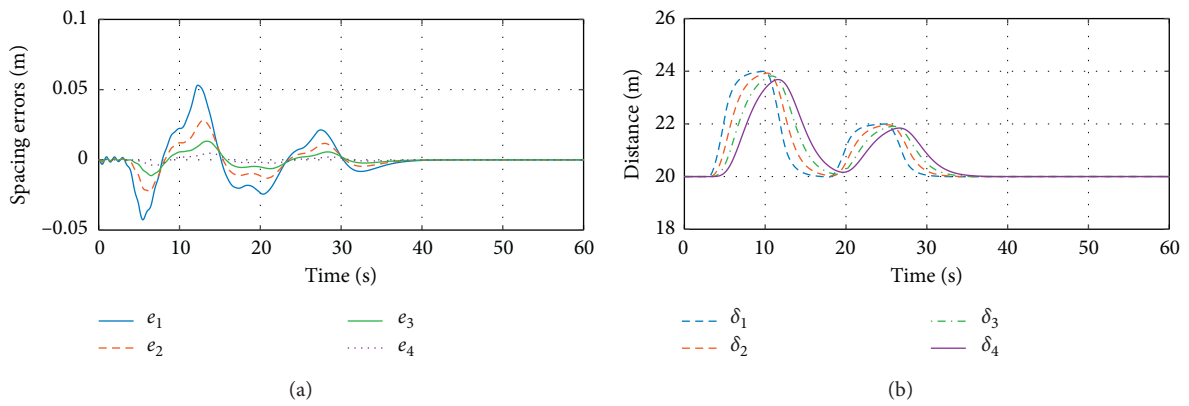
FIGURE 4: Distances and spacing errors of the platoon with CTHP ($h_i = 1$) [25].

TABLE 1: The performance comparison of different spacing policies.

Vehicle index	Tracking performance		Fuel economy		Ride comfort	
	CTH	QSP	CTH	QSP	CTH	QSP
1	11.1820	15.3432	12.2174	10.6983	0.8967	0.7205
2	8.1891	10.1266	8.5112	7.1400	0.3128	0.2319
3	6.5083	7.4740	6.6575	5.4629	0.1689	0.1295
4	5.3897	5.8381	5.4872	4.4124	0.1102	0.0923

5. Conclusion

In this paper, we study the nonlinear vehicle platoon problems with external disturbances, in which the QSP is applied into the platoon control. The desired intervehicle distance in the QSP is a quadratic function in terms of the vehicle's velocities. Comparing with CTHP, the QSP can be regarded as the VTHP to improve traffic capacity and reduce energy consumption. Then, a novel platoon control scheme based on the DISM with QSP and nonlinear dynamics is proposed. Moreover, the finite-time stability of each vehicle is shown by using the Lyapunov theory, and a coupled sliding mode surface is adopted to guarantee the string stability of the vehicle platoon system. In final, the numerical simulations are provided to verify the proposed approaches. In the future, we will continue to investigate the vehicle

platoon with VSP where the unknown external disturbances and mixed traffic situations will be considered.

Data Availability

The data used to support the findings of this study are included within the article and are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Nos.51909008, 61803040, and 62003054) and the Natural Science Basic Research Plan in Shanxi Province of China (Nos.2019GY-218 and 2020GY113).

References

- [1] X. Hu, C. Zou, X. Tang, T. Liu, and L. Hu, "Cost-optimal energy management of hybrid electric vehicles using fuel cell/battery health-aware predictive control," *IEEE Transactions on Power Electronics*, vol. 35, no. 1, pp. 382–392, 2020.

- [2] G. Guo and D. Li, "Adaptive sliding mode control of vehicular platoons with prescribed tracking performance," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 8, pp. 7511–7520, 2019.
- [3] H. Guo, Z. Yin, D. Cao, H. Chen, and C. Lv, "A review of estimation for vehicle tire-road interactions toward automated driving," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 1, pp. 14–30, 2019.
- [4] H. Guo, F. Liu, F. Xu, H. Chen, D. Cao, and Y. Ji, "Nonlinear model predictive lateral stability control of active chassis for intelligent vehicles and its fpga implementation," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 1, pp. 2–13, 2019.
- [5] T. Liu, X. Tang, H. Wang, H. Yu, and X. Hu, "Adaptive hierarchical energy management design for a plug-in hybrid electric vehicle," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 12, pp. 11 513–11 522, 2019.
- [6] X. Hu, T. Liu, X. Qi, and M. Barth, "Reinforcement learning for hybrid and plug-in hybrid electric vehicle energy management: recent advances and prospects," *IEEE Industrial Electronics Magazine*, vol. 13, no. 3, pp. 16–25, 2019.
- [7] P. Yang, Y. Tang, M. Yan, and X. Zhu, "Consensus based control algorithm for nonlinear vehicle platoons in the presence of time delay," *International Journal of Control, Automation and Systems*, vol. 17, no. 3, pp. 752–764, 2019.
- [8] M. Yan, J. Song, P. Yang, and L. Zuo, "Neural adaptive sliding-mode control of a bidirectional vehicle platoon with velocity constraints and input saturation," *Complexity*, vol. 2018, Article ID 1696851, 16 pages, 2018.
- [9] J. Song, M. Yan, Y. Ju, and P. Yang, "Nonlinear gain feedback adaptive dsc for a class of uncertain nonlinear systems with asymptotic output tracking," *Nonlinear Dynamics*, vol. 98, no. 3, pp. 2195–2210, 2019.
- [10] J. Song, M. Yan, and P. Yang, "Neural adaptive dynamic surface asymptotic tracking control for a class of uncertain nonlinear system," *Circuits, Systems, and Signal Processing*, vol. 12, pp. 1–26, 2020.
- [11] M. Yan, W. Ma, L. Zuo, and P. Yang, "Distributed model predictive control for platooning of heterogeneous vehicles with multiple constraints and communication delays," *Journal of Advanced Transportation*, vol. 2020, pp. 1–16, Article ID 4657584, 2020.
- [12] Y. Li, C. Tang, S. Peeta, and Y. Wang, "Integral-sliding-mode braking control for a connected vehicle platoon: theory and application," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 6, pp. 4618–4628, 2019.
- [13] X. Guo, J. Wang, F. Liao, and R. S. H. Teo, "Neuroadaptive quantized pid sliding-mode control for heterogeneous vehicular platoon with unknown actuator deadzone," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 1, pp. 188–208, 2019.
- [14] G. Guo and D. Li, "Pmp-based set-point optimization and sliding-mode control of vehicular platoons," *IEEE Transactions on Computational Social Systems*, vol. 5, no. 2, pp. 553–562, 2018.
- [15] X. Guo, J. Wang, F. Liao, and R. S. H. Teo, "Distributed adaptive integrated-sliding-mode controller synthesis for string stability of vehicle platoons," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 9, pp. 2419–2429, 2016.
- [16] L. Zuo, R. Cui, and W. Yan, "Terminal sliding mode-based cooperative tracking control for non-linear dynamic systems," *Transactions of the Institute of Measurement and Control*, vol. 39, no. 7, pp. 1081–1087, 2017.
- [17] J.-C. Song and Y.-F. Ju, "Distributed adaptive sliding mode control for vehicle platoon with uncertain driving resistance and actuator saturation," *Complexity*, vol. 2020, no. 20, pp. 1–12, Article ID 7581517, 2020.
- [18] G. Rodonyi, "An adaptive spacing policy guaranteeing string stability in multi-brand ad hoc platoons," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, no. 6, pp. 1902–1912, 2018.
- [19] D. Swaroop, J. K. Hedrick, and S. B. Choi, "Direct adaptive longitudinal control of vehicle platoons," *IEEE Transactions on Vehicular Technology*, vol. 50, no. 1, pp. 150–161, 2001.
- [20] B. Bayar, S. A. Sajadi-Alamdari, F. Viti, and H. Voos, "Impact of different spacing policies for adaptive cruise control on traffic and energy consumption of electric vehicles," in *Proceedings of the 2016 24th Mediterranean Conference on Control and Automation (MED)*, pp. 1349–1354, IEEE, Valletta, Malta, July 2016.
- [21] J. Zhang and P. Ioannou, "Longitudinal control of heavy trucks in mixed traffic: environmental and fuel economy considerations," *IEEE Transactions on Intelligent Transportation Systems*, vol. 7, no. 1, pp. 92–104, 2006.
- [22] T. Petrić and I. Petrović, "Longitudinal spacing control of vehicles in a platoon for stable and increased traffic flow," in *Proceedings of the 2012 IEEE International Conference on Control Applications*, pp. 178–183, IEEE, Dubrovnik, Croatia, October 2012.
- [23] H. E. Sungu, M. Inoue, and J. I. Imura, "Nonlinear spacing policy based vehicle platoon control for local string stability and global traffic flow stability," in *Proceedings of the 2015 European Control Conference*, pp. 3396–3401, ECC, Linz, Austria, July 2015.
- [24] G. Guo and W. Yue, "Hierarchical platoon control with heterogeneous information feedback," *IET Control Theory & Applications*, vol. 5, no. 15, pp. 1766–1781, 2011.
- [25] X. Yu, G. Guo, and H. Lei, "Longitudinal cooperative control for a bidirectional platoon of vehicles with constant time headway policy," in *Proceedings of the 2018 Chinese Control and Decision Conference (CCDC)*, pp. 2427–2432, IEEE, Shenyang, China, June 2018.
- [26] J.-W. Kwon and D. Chwa, "Adaptive bidirectional platoon control using a coupled sliding mode control method," *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 2040–2048, 2014.
- [27] H. K. Khalil, "Nonlinear Systems," 2002.
- [28] S. P. Bhat and D. S. Bernstein, "Continuous finite-time stabilization of the translational and rotational double integrators," *IEEE Transactions on Automatic Control*, vol. 43, no. 5, pp. 678–682, 1998.
- [29] S. Li, K. Li, R. Rajamani, and J. Wang, "Model predictive multi-objective vehicular adaptive cruise control," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 3, pp. 556–566, 2011.

Research Article

Extended Kalman Filter-Based Approach for Autonomous Synchronization and Ranging in GPS-Denied Environments

Xiaobo Gu ^{1,2}, Weiqiang Tan ³, Di Zhang ⁴, Yudong Lu ⁵ and Ruidian Zhan ⁶

¹School of Automation, Guangdong University of Technology, Guangzhou 510000, China

²Research Institute of Integrated Circuit Innovation, Guangdong University of Technology, Guangzhou 510006, China

³School of Computer Science and Cyber Engineering, Guangzhou University, Guangzhou 510006, China

⁴The Second Military Agent Office, Equipment Department of China PLA Air Force in Beijing Area, Beijing 100074, China

⁵Guangzhou GRG Metrology & Test Co., Ltd., Guangzhou 510656, China

⁶Chipeye Microelectronics Foshan Ltd., Foshan 528225, China

Correspondence should be addressed to Ruidian Zhan; zhanruidian@chipeye.cn

Received 25 August 2020; Revised 14 November 2020; Accepted 6 December 2020; Published 19 December 2020

Academic Editor: Xin Dong

Copyright © 2020 Xiaobo Gu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Network ranging and clock synchronization based on two-way timing stamps exchange mechanism in complex GPS-denied environments is addressed in this paper. An estimator based on the Extended Kalman filter (EKF) is derived, according to which, the clock skew, clock offset, and ranging information can be jointly estimated. The proposed estimator provides off-line computation by storing the transmitting timing stamps in advance and could be implemented in asymmetrical and asynchronous scenarios. The simulation results show that the proposed estimator achieves a relative good performance than the existed estimators. In addition, a new Bayesian Cramér–Rao Lower Bound (B-CRLB) is derived. Numerous simulation results show that the proposed estimator meets the B-CRLB.

1. Introduction

The Unmanned Autonomous System (UAS) [1–3] has become a hotspot in the past decades with the development of technology. In order to carry out unmanned and autonomous missions, the real-time information of positions, velocities, and timing tags of the system agents are required to be collected. A common approach is to deploy Global Positioning System (GPS) receivers on the network members, which is able to provide precise and accurate Positioning, Velocity, and Timing (PVT) service [4]. However, such a design is no longer available in the complex scenarios where GPS service is not sky visible or of low quality: underwater, valley, indoors, and underground. To solve this problem, the fixed anchor nodes with known positions are implemented to substitute for the PVT service of GPS [5–8]. By broadcasting its timing information, the network agents in the certain area could be able to perform relative positioning, motion estimation and cooperative missions by way of Time

of Arrival (TOA) [9] and Time Difference of Arrival (TDOA) [10] in complex environments. It is noted that achieving time synchronization among the anchor nodes is the first step to carry out the subsequent missions.

Normally, the anchor nodes are of low overall cost, which implies that the on-board oscillators are not highly qualified and the measurement devices are not absolutely accurate. Moreover, the nodes might be affected by the external environmental changes as well. These factors bring a problem that the impacts owing to the relative clock skews and offsets between the clocks could not be ignored [11]. How to correct these errors becomes paramount, especially for the UASs which require all the anchor nodes to provide the synchronized timing tags. Furthermore, power consumption is another important issue for the low-end sensor nodes. Therefore, the message transaction times among the anchor nodes and the message itself should be as few as possible.

In order to estimate both clock skew and clock offset under unknown delay, a Maximum Likelihood-like

Estimator (MLLE) was derived in [12]. A step further, Leng and Wu proposed a Low Complexity Least Square (LCLS) [13] method which outperforms MLLE. Rajan and van der Veen proposed a Global Least Square (GLS) method [14], which was an extension of the LCLS estimator and had the function of jointly estimating the clock offsets, clock skews, and relative distances in an anchor network or even in an anchorless network [15]. An algorithm based on the alternating direction method of multipliers was proposed in [16], and the simulations results show that it outperforms the distributed least squares algorithm. In addition, Luo and Wu proposed an approach based on Kalman Filter (KF) [17], according to which, the accumulated clock offsets and clock skews were estimated once each pair of forward link and corresponding reverse link was collected. A similar scheme was also proposed in [18], which extended the state estimation model to a clock adjustment approach.

To the best of our knowledge, the existed estimators based on two-way timing stamp exchange mechanism and the KF require both the information of the forward and reverse links to difference away the time of flight when performing estimation. This kind of property leads to two problems: one is that there might be a long idle time especially when the time interval between each round transaction is very long and the other problem is that this type of structure could not deal with the circumstance of transaction link loss.

The main contribution of this paper is to propose a novel estimator in the context of complex environments, by which the unknown clock skew, clock offset, and propagation delay can be jointly estimated. The proposed estimator adopts EKF to iteratively update the state estimation on clock parameters and ranging measurements. The two-way timing stamp exchange mechanism is implemented as a basic element to establish the communication network. Moreover, for the sake of decreasing the burden on memory, the transmitting time of each node are recorded in advance so that the EKF estimator can provide explicit off-line computation of the estimation error confidence intervals, since these do not depend on the real-time measurements. Unlike the LS-based estimators, which are less than desirable to collect sufficient observations before doing an update in one big calculation, the proposed estimator could perform the updates once a new observation is collected.

Notation: $(\cdot)_k^+$ stands for (\cdot) at time index $k + 1$, $\text{diag}(\cdot)$ denotes a diagonal matrix formed from its vector argument, the transposition is denoted by $(\cdot)^T$, the matrices are denoted by boldface letters, and \mathbf{I}_N denotes an identity matrix of size N .

2. System Model

We consider a fully asynchronous network consisting of N anchor nodes. Each anchor node deploys an imperfect frequency source, and the clock skews and the clock offsets among the network agents are not identical due to various reasons. The relative distance between each pair of anchor nodes is fixed and unknown.

2.1. Clock Model. Since the free-running clock is not ideal owing to various external and internal reasons, the clock error must be taken into consideration. Let t be the global time and t_i be the local time of node i , then the relationship between t and t_i could be expressed by

$$t_i = \omega_i t + \phi_i \Leftrightarrow C_i(t_i) \triangleq t = \alpha_i t_i + \beta_i, \quad (1)$$

where $\omega_i \in \mathbb{R}_+$ and $\phi_i \in \mathbb{R}$ denote the clock skew and the initial clock offset of node i , respectively, $C_i(t_i)$ denotes the global time of node i at local time t_i , and $\alpha_i \in \mathbb{R}_+$ and $\beta_i \in \mathbb{R}$ are virtual parameters derived from ω_i and ϕ_i . For an autonomous time synchronization system, generally, a real or virtual clock is elected as the reference. Let node 1 be the reference in this paper, that is to say, $\omega_1 = 1$ and $\phi_1 = 0$, which is equivalent to $\alpha_1 = 1$ and $\beta_1 = 0$. Comparing the two equations in (1) and extending to a vector form yields

$$\begin{aligned} \mathbf{a} &= \mathbf{1}_{N-1} \otimes \boldsymbol{\omega}, \\ \boldsymbol{\beta} &= -\boldsymbol{\phi} \otimes \boldsymbol{\omega}, \end{aligned} \quad (2)$$

where $\boldsymbol{\omega} = [\omega_2, \dots, \omega_N]^T \in \mathbb{R}_+^{(N-1) \times 1}$, $\boldsymbol{\phi} = [\phi_2, \dots, \phi_N]^T \in \mathbb{R}^{(N-1) \times 1}$, $\boldsymbol{\alpha} = [\alpha_2, \dots, \alpha_N]^T \in \mathbb{R}_+^{(N-1) \times 1}$, and $\boldsymbol{\beta} = [\beta_2, \dots, \beta_N]^T \in \mathbb{R}^{(N-1) \times 1}$.

It is worth noting that the clock skew might vary versus time and the clock drift term might not be ignored in some cases. However, the proposed model could be reasonable for a small period of time and has been largely adopted in the network clock synchronization literature.

2.2. Ranging Model. The two-way timing stamp exchange mechanism with a centralized scenario, as shown in Figure 1, is implemented to realize time synchronization and ranging in this paper. The two nodes are capable of communicating with each other with a master-slave structure. For the k th forward link between node i and node j , two timing stamps $\{T_{1,ij}^k, T_{2,ij}^k\}$ are collected by this round of communication, which represent the transmitting time instant and reception time instant, respectively. Similarly, the timing stamps $\{T_{3,ij}^k, T_{4,ij}^k\}$ could be collected by the reverse link as well. All the data of two-way communication timing stamps are stored in the center processing unit to preform estimation. The k th forward link and reverse link can be modelled as

$$C_j(T_{2,ij}^k + n_1^k) - C_i(T_{1,ij}^k + n_2^k) = \tau_{i,j} + n_3^k, \quad (3)$$

$$C_i(T_{4,ij}^k + n_4^k) - C_j(T_{3,ij}^k + n_5^k) = \tau_{j,i} + n_6^k, \quad (4)$$

where $\{n_1^k, n_2^k, n_4^k, n_5^k\} \sim \mathcal{N}(0, \sigma_p^2)$ and $\{n_3^k, n_6^k\} \sim \mathcal{N}(0, \sigma_q^2)$ are independent aggregate Gaussian noise variables, which originate from measurements and space disturbances, respectively. $\tau_{i,j}$ denotes the propagation delay between node i and node j . Since the relative positions of the anchor nodes are fixed, the propagation delays of the forward links and reverse links for each pair of master and slave nodes are identical, namely, $\tau_{i,j} = \tau_{j,i}$. It can be seen that the maximum number of direct communication links is $L = (N! / (N - 2)!)$ when there is a full connection topology. The unknown propagation delay is expressed as $\boldsymbol{\tau} = [\tau_{1,2}, \dots, \tau_{1,N},$

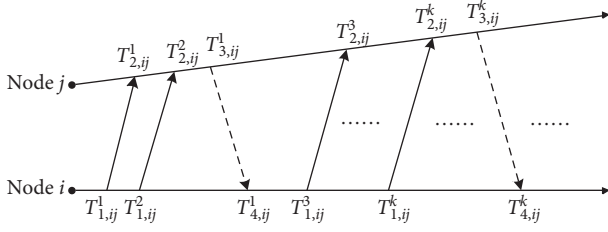


FIGURE 1: Two-way timing stamp exchange mechanism between node i and node j . The solid (dashed) lines denote the forward (reverse) links.

$\tau_{2,3}, \dots, \tau_{2,N}, \tau_{(N-1),N}]^T \in \mathbb{R}^{(L/2) \times 1}$. Substituting (1) into (3) and (4) yields

$$\alpha_i T_{1,ij}^k = \alpha_j T_{2,ij}^k - \beta_i + \beta_j - \tau_{i,j} + \alpha_j n_1^k - \alpha_i n_2^k - n_3^k, \quad (5)$$

$$\alpha_i T_{4,ij}^k = \alpha_j T_{3,ij}^k - \beta_i + \beta_j + \tau_{i,j} - \alpha_i n_4^k + \alpha_j n_5^k + n_6^k. \quad (6)$$

Rearranging (5) and (6) yields

$$T_{2,ij}^k = \frac{\alpha_i T_{1,ij}^k + \beta_i - \beta_j + \tau_{i,j} - n_1^k + \frac{\alpha_i n_2^k + n_3^k}{\alpha_j}}{\alpha_j}, \quad (7)$$

$$T_{4,ij}^k = \frac{\alpha_j T_{3,ij}^k - \beta_i + \beta_j + \tau_{i,j} - n_4^k + \frac{\alpha_j n_5^k + n_6^k}{\alpha_i}}{\alpha_i}. \quad (8)$$

2.3. Proposed Estimator. By combining the clock model and ranging model, a new state estimation model is proposed in this paper. According to this model, the clock error and ranging error could be jointly considered and estimated by the discrete state estimation method.

Substituting (2) into (5) and (6) leads to

$$T_{2,ij}^k = \frac{T_{1,ij}^k \omega_j}{\omega_i} - \frac{\phi_i \omega_j}{\omega_i} + \phi_j + \tau_{i,j} \omega_j - n_1^k + \frac{n_2^k \omega_j}{\omega_i} + n_3^k \omega_j, \quad (9)$$

$$T_{4,ij}^k = \frac{T_{3,ij}^k \omega_i}{\omega_j} - \frac{\phi_j \omega_i}{\omega_j} + \phi_i + \tau_{i,j} \omega_i - n_4^k + \frac{n_5^k \omega_i}{\omega_j} + n_6^k \omega_i. \quad (10)$$

It is worth noting that the transmitting time series $T_{1,ij}^k$ and $T_{3,ij}^k$ are timing tags recorded by the member nodes. For the sake of convenience and building up the discrete time state model, they normally are set as periodic series. In this case, $T_{1,ij}^k$ and $T_{3,ij}^k$ could be considered as known parameters, and the left sides of (7) and (8) could be considered as observations. The state estimation could be updated once a new observation has been collected. Therefore, this process can be expressed by a nonlinear state estimation model:

$$\begin{cases} \mathbf{X}_k^+ = \mathbf{A}\mathbf{X}_k + \mathbf{r}_k, \\ \mathbf{z}_k = \mathbf{f}_k(\mathbf{X}_k) + \mathbf{v}_k, \end{cases} \quad (11)$$

where the unknown state matrix $\mathbf{X}_k = [\omega, \phi, \tau]^T \in \mathbb{R}^{(2N-2+(L/2)) \times 1}$. $\mathbf{A} = \mathbf{I}_{(2N-2+(L/2))}$ and $\{\mathbf{v}_k\}$ are white Gaussian, independent random processes with zero mean and covariance matrix $E[\mathbf{r}_k \mathbf{r}_k^T] = \mathbf{R}_k \in \mathbb{R}^{(2N-2+(L/2)) \times (2N-2+(L/2))}$ and

$E[\mathbf{v}_k \mathbf{v}_k^T] = \mathbf{Q}_k \in \mathbb{R}^{1 \times 1}$. $\mathbf{f}_k(\mathbf{X}_k)$ varies versus the direction of the link. From (9) and (10), we can infer that if the transmission direction is from node i to node j , the observation equation follows (9), otherwise it follows (10). The estimation could be performed by an extended Kalman filter (EKF) in this case. In addition, if the two-way timing stamps exchange mechanism operates between the master and slave, the observation function $\mathbf{f}_k(\mathbf{X}_k)$ can be simplified as $T_{1,ij}^k \omega_j + \phi_j + \tau_{i,j} \omega_j$ and $(T_{3,ij}^k / \omega_j) - (\phi_j / \omega_j) + \tau_{i,j}$, respectively.

Based on the discussions above, the proposed EKF estimator is expressed by Algorithm 1.

Remark 1 (off-line computation). Note that the matrices \mathbf{K} and $\mathbf{P}_{m|m}$ can be computed off-line; the reason is that they do not depend on the timing stamp data of reception. Therefore, the computation of \hat{X} can be performed very rapidly on the center processing unit. It can be found that one disadvantage of EKF is that it might take a large amount of memories to store the transmission data especially when the data is large, but this problem could be easily solved by setting the transmission time instants periodically.

Remark 2 (real-time process). Compared with the existed LS-based estimators, which need to collect the information with respect to forward links and reverse links as much as possible to improve the estimation accuracy, EKF could update its estimation once there is a new observation of a forward or a reverse link. On the contrary, although the Kalman-filter-based algorithms proposed in [17, 18] are also able to carry out real-time processing, the symmetrical links are required to eliminate the nuisance term. In addition, another advantage owing to not to put any constraint on symmetrical structure is that the update rate could be faster than those symmetrical required algorithms. This kind of feature is paramount especially for the low power consumption systems, such as the sleep wake-up scheduling system.

Remark 3 (one-way estimation). According to (7) and (8), we can infer that the EKF estimator will not lose its entire function even when the transmitting device or receiving device on any node suddenly stops working. For instance, if (10) could not be observed, that is to say, only a series of information with respect to (9) can be used to perform estimation, in this case, the estimation rate and the robustness of the estimator could be improved.

3. Bayesiam Cramér–Rao Lower Bound

The Cramér–Rao lower bound on the estimation error states

$$E[(\hat{X} - \mathbf{X})(\hat{X} - \mathbf{X})^T] \geq \mathbf{J}^{-1}, \quad (12)$$

where \mathbf{J} is the Fisher information matrix, which obeys the recursion [19]:

$$\mathbf{J}_{k+1} = \mathbf{D}_k^{22} - (\mathbf{D}_k^{12})^T (\mathbf{J}_k + \mathbf{D}_k^{11})^{-1} \mathbf{D}_k^{12}, \quad (13)$$

where

Require: $\{T_{1,ij}^1, T_{1,ij}^2, \dots, T_{1,ij}^{K_i}\}, \{T_{3,ij}^1, T_{3,ij}^2, \dots, T_{3,ij}^{U_i}\}, K \in \mathbb{N}, \sigma_p, \sigma_q, \hat{X}_{0|0}, \mathbf{P}_{0|0}$

- (1) $m = 0$
- (2) **repeat**
- (3) $m = m + 1$
- (4) $\hat{X}_{m|m-1} = \mathbf{A}\hat{X}_{m-1|m-1}$
- (5) $\mathbf{P}_{m|m-1} = \mathbf{A}\mathbf{P}_{m-1|m-1}\mathbf{A}^T + \mathbf{R}_m$
- (6) **if** $j \rightarrow i$ **then**
- (7) $\mathbf{z}_m = T_{4,ij}^u$
- (8) $\mathbf{f}_m(\mathbf{X}_m) = T_{3,ij}^k/\omega_j - \phi_j/\omega_j + \tau_{i,j}$, if $i == 1$
- (9) $\mathbf{f}_m(\mathbf{X}_m) = T_{3,ij}^k\omega_i/\omega_j - \phi_j\omega_i/\omega_j + \phi_i + \tau_{i,j}\omega_i$, if $i! = 1$
- (10) **else**
- (11) $\mathbf{z}_m = T_{2,ij}^u$
- (12) $\mathbf{f}_m(\mathbf{X}_m) = T_{1,ij}^k\omega_j - \phi_i\omega_j + \phi_j + \tau_{i,j}\omega_j$, if $i == 1$
- (13) $\mathbf{f}_m(\mathbf{X}_m) = T_{1,ij}^k\omega_j/\omega_i - \phi_i\omega_j/\omega_i + \phi_j + \tau_{i,j}\omega_j$, if $i! = 1$
- (14) $\mathbf{F}_m = (\partial\mathbf{f}_m/\partial\mathbf{X}_m)|_{\mathbf{X}_m=\hat{X}_{m|m-1}}$
- (15) $\mathbf{K}_m = \mathbf{P}_{m|m-1}\mathbf{F}_m^T(\mathbf{Q}_m + \mathbf{F}_m\mathbf{P}_{m|m-1}\mathbf{F}_m^T)^{-1}$
- (16) $\hat{X}_{m|m} = \hat{X}_{m|m-1} + \mathbf{K}_m(\mathbf{z}_m - \mathbf{f}_m(\hat{X}_{m|m-1}))$
- (17) $\mathbf{P}_{m|m} = (\mathbf{I} - \mathbf{K}_m\mathbf{F}_m)\mathbf{P}_{m|m-1}$
- (18) **end if**
- (19) **until** $m = K_i + U_i$

ALGORITHM 1: Proposed EKF estimator.

$$\mathbf{D}_k^{11} = E \left\{ -\frac{\partial^2}{\partial \mathbf{X}_k \partial^T \mathbf{X}_k} \log p(\mathbf{X}_k^+ | \mathbf{X}_k) \right\}, \quad (14)$$

$$\mathbf{D}_k^{12} = E \left\{ -\frac{\partial^2}{\partial \mathbf{X}_k \partial^T \mathbf{X}_k^+} \log p(\mathbf{X}_k^+ | \mathbf{X}_k) \right\}, \quad (15)$$

$$\begin{aligned} \mathbf{D}_k^{22} = E \left\{ -\frac{\partial^2}{\partial \mathbf{X}_k^+ \partial^T \mathbf{X}_k^+} \log p(\mathbf{X}_k^+ | \mathbf{X}_k) \right\} \\ + E \left\{ -\frac{\partial^2}{\partial \mathbf{X}_k^+ \partial^T \mathbf{X}_k^+} \log p(z_k^+ | \mathbf{X}_k^+) \right\}. \end{aligned} \quad (16)$$

From (11), it follows that

$$-\log p(\mathbf{X}_k^+ | \mathbf{X}_k) = c_1 + \frac{1}{2}(\mathbf{X}_k^+ - \mathbf{A}\mathbf{X}_k)^T \cdot \mathbf{R}^{-1}(\mathbf{X}_k^+ - \mathbf{A}\mathbf{X}_k), \quad (17)$$

$$-\log p(z_k^+ | \mathbf{X}_k^+) = c_2 + \frac{1}{2}[z_k^+ - \mathbf{f}_k^+(\hat{X}_k^+)]^T \cdot \mathbf{Q}^{-1}[z_k^+ - \mathbf{f}_k^+(\hat{X}_k^+)], \quad (18)$$

where c_1 and c_2 are constants. A straightforward calculation of (14)–(18) leads to $\mathbf{D}_k^{11} = \mathbf{A}^T \mathbf{R}^{-1} \mathbf{A}$, $\mathbf{D}_k^{12} = -\mathbf{A}^T \mathbf{R}^{-1}$, and $\mathbf{D}_k^{22} = \mathbf{R}^{-1} + \left\{ (\partial \mathbf{f}_k^+ / \partial \hat{X}_k^+) \right\}^T \mathbf{Q}^{-1} (\partial \mathbf{f}_k^+ / \partial \hat{X}_k^+)$. Hence, (13) can be written as

$$\begin{aligned} \mathbf{J}_{k+1} = \mathbf{R}^{-1} + \left\{ \frac{\partial \mathbf{f}_k^+(\hat{X}_k^+)}{\partial \hat{X}_k^+} \right\}^T \mathbf{Q}^{-1} \frac{\partial \mathbf{f}_k^+(\hat{X}_k^+)}{\partial \hat{X}_k^+} \\ - (\mathbf{R}^{-1})^T \mathbf{A} (\mathbf{J}_k + \mathbf{A}^T \mathbf{R}^{-1} \mathbf{A})^{-1} \mathbf{A}^T \mathbf{R}^{-1}. \end{aligned} \quad (19)$$

Applying the matrix inversion lemma, (19) can be further simplified as

$$\mathbf{J}_{k+1} = (\mathbf{R} + \mathbf{A}\mathbf{J}_k^{-1}\mathbf{A}^T)^{-1} + \left\{ \frac{\partial \mathbf{f}_k^+(\hat{X}_k^+)}{\partial \hat{X}_k^+} \right\}^T \mathbf{Q}^{-1} \frac{\partial \mathbf{f}_k^+(\hat{X}_k^+)}{\partial \hat{X}_k^+}. \quad (20)$$

The inverse of the Fisher information matrix for θ could be written as [20]

$$\mathbf{\Gamma}^{-1}(\boldsymbol{\theta}) = \mathbf{J}^{-1}. \quad (21)$$

4. Simulations

In this section, we provide a scenario with asynchronous links to evaluate the performance of the EKF estimator. The number of the nodes $N=10$, and each node is capable of performing two-way communications with the other nodes. All the member nodes are assumed to be visible during the signal transaction process. The locations of the member nodes are uniformly distributed variables in the region of $1 \text{ km} \times 1 \text{ km}$. The propagation speed is the speed of light. The clock skew and clock offset of the slave nodes are randomly distributed in the range $[0.9998, 1.0002]$ and $[-2, 2]$ s, respectively. The transmission time interval for both forward links and reverse links is 5 s, and the reverse links always transmit backwards by 3 s. The noise standard deviations $\sigma_p = 1 \times 10^{-3}$ and $\sigma_q = 1 \times 10^{-3}$. The Root Mean Square Error (RMSE) is adopted as the performance metric. All the given results are averaged over 10,000 Monte Carlo runs.

In order to investigate the performance of the proposed method, the conventional Kalman estimator based on the traditional clock model and the iterative form of LCLS are compared in this section. Figure 2 shows the RMSEs of the unknown parameters by applying different estimators. As shown in the figure, the iterative LCLS achieve the best performance than the other estimators. The proposed EKF

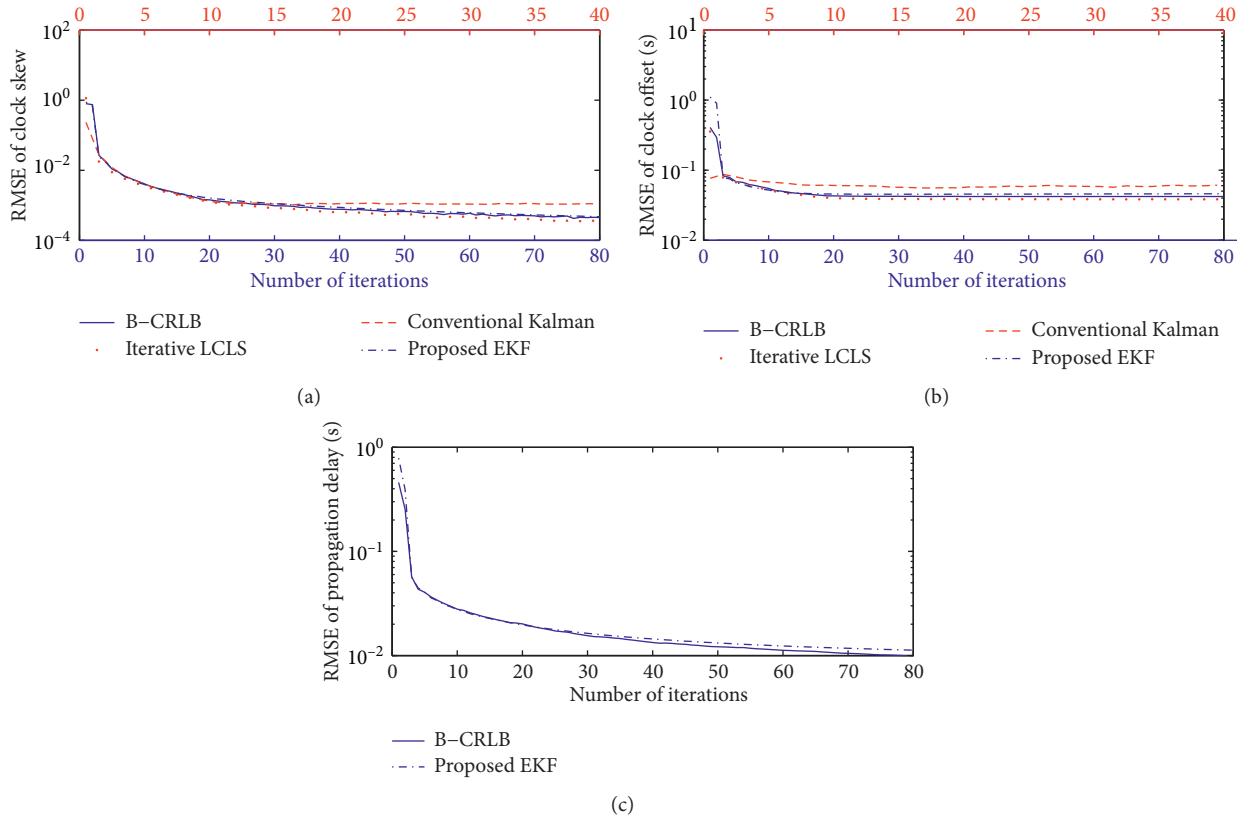


FIGURE 2: RMSEs of the estimated clock skew ω_j (a), offset ϕ_j (b), and propagation delay $\tau_{i,j}$ (c) with respect to the number of iterations.

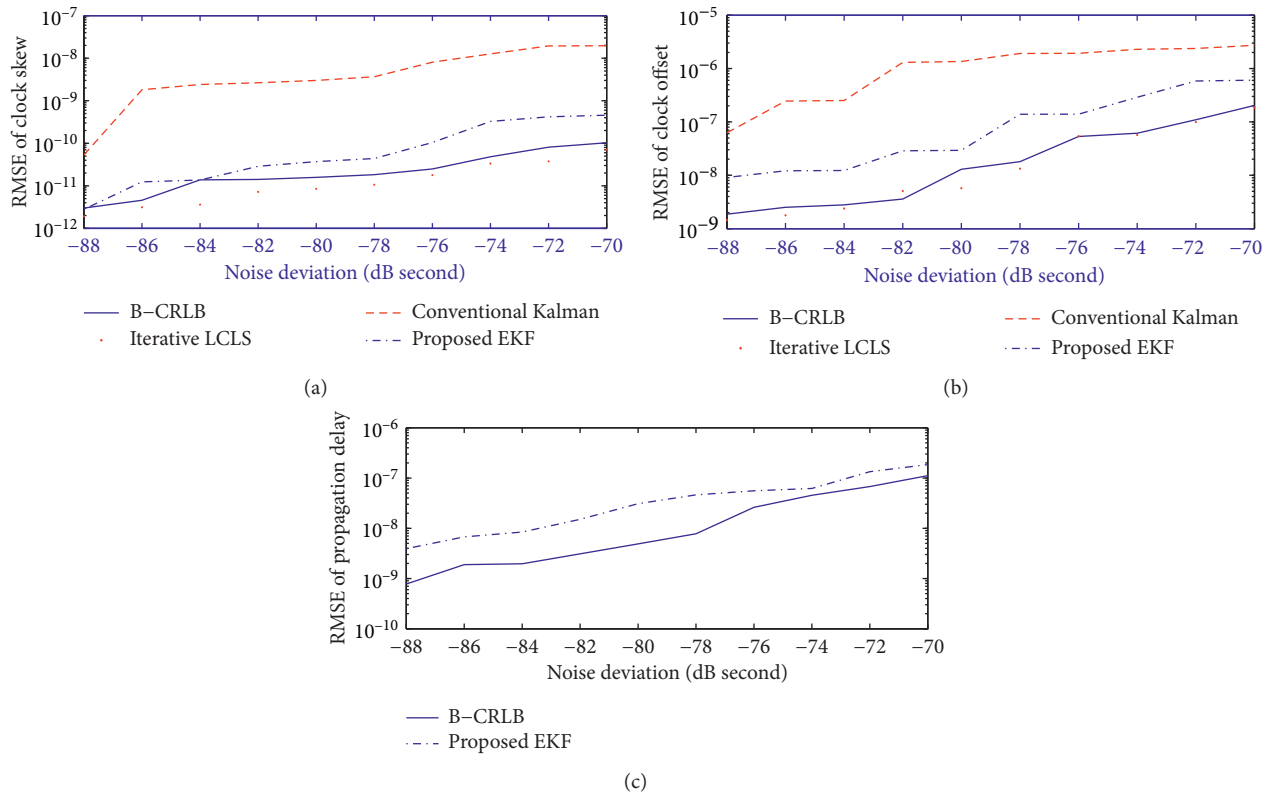


FIGURE 3: RMSEs of the estimated clock skew ω_j (a), offset ϕ_j (b), and propagation delay $\tau_{i,j}$ (c) with respect to the varying noise deviations.

converges at a slower rate than the conventional KF, but performs relatively better on the estimation precision after a long iteration time. Compared with the other two estimators, the precision degradation of the EKF originates from the nonlinear process of the observation equation, and it could be improved by omitting a virtual parameter transforming process. However, the proposed EKF estimator has its own superiorities mentioned in Section 2, namely, its update rate is two times faster than the other estimators and it frees the restriction on the symmetrical structure. Meanwhile, it is able to provide off-line computation.

Furthermore, the performances of the estimators for varying noise deviation are presented in Figure 3. The deviation of the time markers and propagation disturbance are set in the range $[-88, -70]$ dB seconds. It is noted that the meter level accuracies could be achieved by the proposed EKF estimator when the Signal-to-Noise Ratio (SNR) is high. In addition, the proposed estimator is able to carry out range estimation directly, which is coupled with the estimation process of clock parameter tracking.

5. Conclusion

In this work, the problem of joint clock synchronization and ranging for UAS in a complex GPS-denied environment is addressed. The proposed EKF estimator is shown to achieve nearly the same performance compared with the existed Kalman-filter-based estimator, but can provide off-line computations. By comparing its performance bounds with the B-CRLB, it has been found that although the EKF estimator suffers performance degradation, it is capable of estimating the clock skew even in the context of one-way communication and updating the estimates faster than the existed estimators. Furthermore, it has the function of estimating the unknown propagation delay between each node. Future research directions include the extension of recursive joint clock synchronization and localization estimation in anchorless networks.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon reasonable request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Key-Area Research & Development Program of Guangdong Province under Grant 2019B010142001, and the Science & Technology Program of Guangdong under Grant 2019A141401005 and the National Natural Science Foundation of China under Grant 61801132.

References

- [1] A. Zolich, D. Palma, K. Kansanen et al., "Survey on communication and networks for autonomous marine systems," *Journal of Intelligent & Robotic Systems*, vol. 95, no. 3-4, pp. 789–813, 2019.
- [2] Q. P. Ha, L. Yen, and C. Balaguer, "Robotic autonomous systems for earthmoving in military applications," *Automation in Construction*, vol. 107, p. 102934, 2019.
- [3] R. Yang, A. Zhang, L. Zhang, and Y. Hu, "A novel adaptive h-infinity cubature kalman filter algorithm based on sage-husa estimator for unmanned underwater vehicle," *Mathematical Problems in Engineering*, vol. 2020, Article ID 8057028, 10 pages, 2020.
- [4] E. Kaplan and C. Hegarty, *Understanding GPS: Principles and Applications*, Artech House, Norwood, MA, USA, 2005.
- [5] Y. Xiong, N. Wu, Y. Shen, and M. Z. Win, "Cooperative network synchronization: asymptotic analysis," *IEEE Transactions on Signal Processing*, vol. 66, no. 3, pp. 757–772, 2017.
- [6] J. Liu, Z. Wang, J.-H. Cui, S. Zhou, and B. Yang, "A joint time synchronization and localization design for mobile underwater sensor networks," *IEEE Transactions on Mobile Computing*, vol. 15, no. 3, pp. 530–543, 2015.
- [7] N. M. Freris, S. R. Graham, and P. Kumar, "Fundamental limits on synchronizing clocks over networks," *IEEE Transactions on Automatic Control*, vol. 56, no. 6, pp. 1352–1364, 2010.
- [8] R. M. Vaghefi and R. M. Buehrer, "Cooperative joint synchronization and localization in wireless sensor networks," *IEEE Transactions on Signal Processing*, vol. 63, no. 14, pp. 3615–3627, 2015.
- [9] K. Yu and I. Oppermann, "Performance of uwb position estimation based on time-of-arrival measurements," in *Proceedings of the 2004 International Workshop on Ultra Wideband Systems Joint with Conference on Ultra Wideband Systems and Technologies*, pp. 400–404, IEEE, Kyoto, Japan, May 2004.
- [10] L. Cong and W. Zhuang, "Non-line-of-sight error mitigation in tdoa mobile location," in *Proceedings of the GLOBECOM'01, IEEE Global Telecommunications Conference*, pp. 680–684, IEEE, San Antonio, TX, USA, November 2001.
- [11] Y. S. Shmaliy, S. H. Khan, S. Zhao, and O. Ibarra-Manzano, "General unbiased fir filter with applications to gps-based steering of oscillator frequency," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 3, pp. 1141–1148, 2016.
- [12] K.-L. Noh, Q. M. Chaudhari, E. Serpedin, and B. W. Suter, "Novel clock phase offset and skew estimation using two-way timing message exchanges for wireless sensor networks," *IEEE Transactions on Communications*, vol. 55, no. 4, pp. 766–777, 2007.
- [13] M. Leng and Y.-C. Wu, "On clock synchronization algorithms for wireless sensor networks under unknown delay," *Vehicular Technology*, *IEEE Transactions on*, vol. 59, no. 1, pp. 182–190, 2010.
- [14] R. T. Rajan and A.-J. Van der Veen, "Joint ranging and clock synchronization for a wireless network," in *Proceedings of the 2011 4th IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pp. 297–300, IEEE, San Juan, Puerto Rico, December 2011.
- [15] R. T. Rajan, G. Leus, and A.-J. van der Veen, "Relative kinematics of an anchorless network," *Signal Processing*, vol. 157, pp. 266–279, 2019.

- [16] B. Luo, L. Cheng, and Y.-C. Wu, "Fully distributed clock synchronization in wireless sensor networks under exponential delays," *Signal Processing*, vol. 125, pp. 261–273, 2016.
- [17] B. Luo and Y. C. Wu, "Distributed clock parameters tracking in wireless sensor network," *IEEE Transactions on Wireless Communications*, vol. 12, no. 12, pp. 6464–6475, 2013.
- [18] G. Giorgi and C. Narduzzi, "Performance analysis of kalman-filter-based clock synchronization in IEEE 1588 networks," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 8, pp. 2902–2909, 2011.
- [19] P. Tichavský, C. H. Muravchik, and A. Nehorai, "Posterior cramer-rao bounds for discrete-time nonlinear filtering," *IEEE Transactions on Signal Processing*, vol. 46, no. 5, pp. 1386–1396, 1998.
- [20] S. M. Kay, *Fundamentals of Statistical Signal Processing: Practical Algorithm Development*, Pearson Education, London, UK, 2013.

Research Article

Novel Robust Stability Criteria of Uncertain Systems with Interval Time-Varying Delay Based on Time-Delay Segmentation Method and Multiple Integrals Functional

Xing He ¹, Li-Jun Song ¹, Yu-Bin Wu,² and Zi-Yu Zhou¹

¹School of Information and Control Engineering, Xi'an University of Architecture and Technology, Xi'an 710055, Shaanxi, China

²Department of Automation, Xi'an Research Institute of High-Technology, Xi'an 710025, Shaanxi, China

Correspondence should be addressed to Li-Jun Song; songlijun9071@sina.com

Received 18 September 2020; Revised 27 October 2020; Accepted 18 November 2020; Published 10 December 2020

Academic Editor: Guang Li

Copyright © 2020 Xing He et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Interval time-varying delay is common in control process, e.g., automatic robot control system, and its stability analysis is of great significance to ensure the reliable control of industrial processes. In order to improve the conservation of the existing robust stability analysis method, this paper considers a class of linear systems with norm-bounded uncertainty and interval time-varying delay as the research object. Less conservative robust stability criterion is put forward based on augmented Lyapunov-Krasovskii (L-K) functional method and reciprocally convex combination. Firstly, the delay interval is partitioned into multiple equidistant subintervals, and a new Lyapunov-Krasovskii functional comprising quadruple-integral term is introduced for each subinterval. Secondly, a novel delay-dependent stability criterion in terms of linear matrix inequalities (LMIs) is given by less conservative Wirtinger-based integral inequality approach. Three numerical comparative examples are given to verify the superiority of the proposed approach in reducing the conservation of conclusion. For the first example about closed-loop control systems with interval time-varying delays, the proposed robust stability criterion could get MADB (Maximum Allowable Delay Bound) about 0.3 more than the best results in the previous literature; and, for two other uncertain systems with interval time-varying delays, the MADB results obtained by the proposed method are better than those in the previous literature by about 0.045 and 0.054, respectively. All the example results obtained in this paper clearly show that our approach is better than other existing methods.

1. Introduction

Many dynamic model systems in the real world contain very significant time delays in the transmission of data and materials, in automatic robot control system, the acquisition and transmission of sensor signals, and the calculation of controller and the drive of brake may lead to time delay. In many kinds of time-delay types, the interval time-varying delay is more representative. The lower bound of its time delay is not necessarily zero, and the time delay is within a changing interval. It is common in practical application of engineering, especially in chemical reactors, internal combustion engines, and network control [1, 2]. Consequently, the stability analysis about interval time-delay systems has attracted wide attention in these years.

Generally, aiming to analyze the stability of time-delay system, the most common method is to construct an appropriate LK functional (Lyapunov-Krasovskii functional, LKF) in time domain and combine it with linear matrix inequalities (LMIs). In general, the free weight matrix method, the time-delay segmentation method, the integral inequality method, the interactive convex combination method, and so forth are used to analyze its stability. Augmented functional method [3–5] can make full use of the system's time-delay information to reduce the conservativeness of conclusion, but the introduction of matrix variables inevitably burdens the theoretical analysis and engineering calculation. Zhang et al. and Shen et al. [6, 7] obtain a conservative less stable stability criterion for linear systems with time-varying delays by constructing LKF with

triple integral functional terms and optimize the stability conditions of time-delay systems. The integral inequality method has the characteristics of simple form and few matrix variables, which can promote the stability analysis of time-delay systems. Gu [8] first introduced Jensen's inequality into the stability analysis of time-delay systems, and then Ramakrishnan [9, 10], Zhang [11], and Gouaisbaut [12] further promoted Jensen's inequality, resulting in different and novel forms. In various forms, we have obtained effective conclusions of different conservation. As an innovative method, the interactive convex combination method [13, 14] can solve the stability problems of systems with interactive convex combination. Wu et al. [15] studied the issue of robust stability analysis for a sort of uncertain neutral system with mixed time-varying delays, and a novel discrete and neutral delay-dependent stability criterion based on linear matrix inequalities was given, which could greatly reduce the complexity of theoretical derivation and computation. Li et al. [16] deal with a set of positive functions combined with inverse convex weighting parameters by interactive convex combination technique instead of directly ignoring the term and deduce the stability criterion for uncertain neutral systems with mixed time delay. This criterion reduces the number of relevant decision variables while ensuring the conservativeness and avoids the complexity of numerical calculation.

Farnam et al. [17] studied the robust stability problem for a class of linear systems with time-varying delays. By constructing LKF with more time-delay information, the stability condition of LMIs is obtained by means of interactive convex combination definition technique. Finally, the numbers are used to demonstrate that the given stability conditions are less conservative in computational efficiency. Ding et al. [18] construct an augmented functional with specific time-delay information based on the idea of time-delayed partitioning. The free-weight matrix inequality is used to define the cross terms generated by the functional derivatives, and a lower-conservative stability criterion is obtained. Senthilraj et al. [19] introduced a novel method to study the robust stability problem of an interval-delayed neural network system by using a nonuniform time-delay segmentation method and the integral inequality definition technique. Delay-dependent stability conditions for ensuring the stability of the system are obtained. Cheng et al. [20] studied a time-delay-related state feedback control problem for a class of time-varying delay continuous systems via improved interactive convex combination techniques and Wirtinger-based integral inequalities, and new stability conditions and state feedback control are obtained. Zhang et al. [21] proposed a robust stability criterion for a class of linear systems with time-varying delays by using the Wirtinger-based integral inequality and the interactive convex combination lemma to effectively define the cross terms emerging in the LKF derivatives. The conclusions obtained are superior in terms of stability analysis. Chang et al. studied the control problem with time-varying norm bounded uncertainties and discrete-time nonlinear systems with parametric uncertainties; LMI are used to obtain the sufficient conditions for robust stabilization [22, 23].

On the basis of the above research results, for the sake of further revealing the relationship between the asymptotical stability of uncertain systems with interval time-varying delay and the constructed LKF and then to lower the conservatism caused by dealing with the functional derivatives, this paper attempts to study the robust stability problem of uncertain systems with interval time-varying delay by constructing a novel LKF and realizing less conservative integral inequalities. The main contributions of this paper include the following:

In this paper, the robust stability criterion is proposed based on the time-delay segmentation method. Specifically, the time-delay interval is divided into N equal parts. Then, a new LKF with quadruple integral term is constructed for different subintervals.

The constructed LKF is augmented with single integral terms and multiple integrals terms, which can make more connections among different vectors and then eliminate the redundant conservatism arising from estimating the interval time-varying delay. Moreover, in addition to the single integral, the double integral, and the triple integral, the quadruple integral is used as a term to construct the integral functional, which would make full use of more information about the upper and lower bounds of the time delay existing in the systems.

The Wirtinger-based integral inequality and interactive convex combination technique are used to give conclusion in the form of LMIs without any extra parameters.

\mathbb{R}^n denotes n -dimensional Euclidean space. $\mathbb{R}^{n \times m}$ denotes the set of all $n \times m$ real matrices. $*$ denotes symmetric terms in symmetric matrices. I denotes the identity matrix with proper dimensions. $M = M^T > 0$ denotes that M is symmetric matrix. e_i denotes block input matrix with proper dimensions; for instance, $e_6^T = [0 \ 0 \ 0 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0 \ 0]$.

2. Problem Description

The uncertain linear systems with interval time-varying delay are as follows:

$$\begin{cases} \dot{x}(t) = (A + \Delta A(t))x(t) + (B + \Delta B(t))x(t - h(t)), \\ x(t) = \varphi(t), \quad t \in [-h_M, 0], \end{cases} \quad (1)$$

where $x(t) \in \mathbb{R}^n$ is the state vector of the system, A and B are system matrices with appropriate dimensions, $h(t)$ is time-varying delay satisfying $0 \leq h_m \leq h(t) \leq h_M$, and $\Delta A(t)$ and $\Delta B(t)$ are unknown matrices with time-varying structure uncertainty. When $\Delta A(t)$ and $\Delta B(t)$ have norm bounded uncertainty, they can be described as follows:

$$[\Delta A(t) \ \Delta B(t)] = DF(t)[E_a \ E_b], \quad (2)$$

where D , E_a , and E_b are known matrices with appropriate dimensions, while $F(t)$ is an uncertain matrix with measurable elements satisfying $F(t)^T F(t) \leq I, \forall t$, in which I represents the unit matrix of the appropriate dimension. When $F(t) = 0$, system (1) becomes a nominal system.

In this paper, assuming that N is a positive integer greater than zero, $h_i (i = 1, 2, \dots, N + 1)$ are scalars, and the time-delay interval $[h_m, h_M]$ can be averaged as follows:

$$h_m = h_1 < h_2 < h_3 < \dots < h_N < h_{N+1} = h_M, \quad (3)$$

where $h_m = h_1$; $h_M = h_{N+1}$; then h_Δ represents the length of subinterval $[h_i, h_{i+1}]$; namely, $h_\Delta = h_{i+1} - h_i = (h_M - h_m)/N$.

To facilitate the proof of stability criteria, the following lemmas are summarized as follows:

Lemma 1 (see [12]). *Assuming any positive definite matrix $M = M^T > 0$, scalar $h > 0$, and continuous vector functions $x(t): [0, h] \rightarrow \mathbb{R}^n$, the following inequality is established:*

$$\begin{aligned} -h \int_{t-h}^t x^T(s) M x(s) ds &\leq - \int_{t-h}^t x^T(s) ds M \int_{t-h}^t x(s) ds \\ &\quad - (h^2/2) \int_{-h}^0 \int_{t+\beta}^t x^T(s) M x(s) ds d\beta \\ &\leq - \int_{-h}^0 \int_{t+\beta}^t x^T(s) ds d\beta M \int_{-h}^0 \int_{t+\beta}^t x(s) ds d\beta, \\ &\quad - (h^3/6) \int_{-h}^0 \int_{\beta}^0 \int_{t+\lambda}^t x^T(s) M x(s) ds d\lambda d\beta \\ &\leq - \int_{-h}^0 \int_{\beta}^0 \int_{t+\lambda}^t x^T(s) ds d\lambda d\beta M \int_{-h}^0 \int_{\beta}^0 \int_{t+\lambda}^t x(s) ds d\lambda d\beta. \end{aligned} \quad (5)$$

Lemma 3 (see [17]). *Assuming any positive definite matrix $M = M^T > 0$, scalars $0 \leq \alpha, \varepsilon \leq 1$, $\alpha = ((h(t) - h_i)/(h_{i+1} - h_i))$, $\varepsilon = ((h(t))^2 - h_i^2)/(h_{i+1}^2 - h_i^2)$, $h_i \leq h(t) \leq h_{i+1}$, and vector*

$$\begin{aligned} -h \int_{t-h}^t \dot{x}^T(s) M \dot{x}(s) ds &\leq - [x(t) - x(t-h)]^T \\ &\quad \cdot M [x(t) - x(t-h)] - 3\Theta^T M \Theta, \end{aligned} \quad (4)$$

where $\Theta = x(t) + x(t-h) - (2/h) \int_{t-h}^t x(s) ds$.

Lemma 2 (see [17]). *Assuming any positive definite matrix $M = M^T > 0$, scalar $h > 0$, and continuous vector functions $x(t): [0, h] \rightarrow \mathbb{R}^n$, the following inequality is established:*

functions $x(t): [0, h] \rightarrow \mathbb{R}^n$, the following inequality is established:

$$\begin{aligned} -(h_{i+1} - h_i) \int_{t-h_{i+1}}^{t-h_i} x^T(s) M x(s) ds &\leq - \zeta^T(t) (e_7 M e_7^T + e_6 M e_6^T) \zeta(t) \\ &\quad - \alpha \zeta^T(t) e_7 M e_7^T \zeta(t) - (1 - \alpha) \zeta^T(t) e_6 M e_6^T \zeta(t), \\ &\quad - ((h_{i+1}^2 - h_i^2)/2) \int_{-h_{i+1}}^{-h_i} \int_{t+\beta}^t x^T(s) M x(s) ds d\beta \\ &\leq - \zeta^T(t) (e_{10} M e_{10}^T + e_9 M e_9^T) \zeta(t) \\ &\quad - \varepsilon \zeta^T(t) e_{10} M e_{10}^T \zeta(t) - (1 - \varepsilon) \zeta^T(t) e_9 M e_9^T \zeta(t), \end{aligned} \quad (6)$$

where

$$\begin{aligned} \zeta^T(t) &= \left[x(t) x(t-h(t)) x(t-h_i) x(t-h_{i+1}) \int_{t-h_i}^t x(s) ds \int_{t-h(t)}^{t-h_i} x(s) ds \int_{t-h_{i+1}}^{t-h(t)} x(s) ds \int_{-h_i}^0 \int_{t+\beta}^t x(s) ds d\beta \right. \\ &\quad \left. \cdot \int_{-h(t)}^{-h_i} \int_{t+\beta}^t x(s) ds d\beta \int_{-h_{i+1}}^{-h(t)} \int_{t+\beta}^t x(s) ds d\beta \right]. \end{aligned} \quad (7)$$

3. Main Results

In this section, the stability of the system is discussed in two steps. First, the stability criterion of the nominal system is given, and then the stability of the uncertain system is analyzed. The nominal system of system (1) is as follows:

$$\begin{cases} \dot{x}(t) = Ax(t) + Bx(t-h(t)), \\ x(t) = \varphi(t), \quad t \in [-h_M, 0]. \end{cases} \quad (8)$$

For nominal systems (8), a new quadruple integral term L-K functional containing more time-delay information is

constructed in each subinterval. The following conclusions are obtained by combining Lemmas 1–3.

Theorem 1. For given scalars h_m , h_M , and λ_1, λ_2 ($\lambda_1 > \lambda_2$), it is asymptotically stable for the nominal system (8), if there exist positive definite symmetric matrices P_i ($i = 1, 2, 3, 4, 5$), $Q_1, Q_2, U_1, U_2, X_j, R_j$ ($j = 1, 2, 3, 4$), such that the following linear matrix inequalities (LMIs) hold:

$$\Phi = (\Phi_{i,j})_{10 \times 10} < 0, \quad (9)$$

where

$$\begin{aligned} \Phi_{11} &= P_1 A + A^T P_1 + Q_1 + h_i^2 X_1 + h_i^2 A^T X_2 A - X_2 + h_\Delta^2 X_3 + h_\Delta^2 A^T X_4 A + \left(\frac{h_i^4}{4}\right) R_1 - h_i^2 R_2 \\ &+ \left(\frac{h_i^4}{4}\right) A^T R_2 A + \left(\frac{(h_{i+1}^2 - h_i^2)^2}{4}\right) R_3 - 2h_\Delta^2 R_4 + \left(\frac{(h_{i+1}^2 - h_i^2)^2}{4}\right) A^T R_4 A - \left(\frac{h_i^4}{4}\right) U_1 + \left(\frac{h_i^6}{36}\right) A^T U_1 A \\ &+ \left(\frac{(h_{i+1}^2 - h_i^2)^2}{4}\right) U_2 - \left(\frac{(h_{i+1}^3 - h_i^3)^2}{36}\right) A^T U_2 A, \\ \Phi_{12} &= P_1 B + h_i^2 A^T X_2 B + h_\Delta^2 A^T X_4 B + \left(\frac{h_i^4}{4}\right) A^T R_2 B \\ &+ \left(\frac{(h_{i+1}^2 - h_i^2)^2}{4}\right) A^T R_4 B + \left(\frac{h_i^6}{36}\right) A^T U_1 B + \left(\frac{(h_{i+1}^3 - h_i^3)^2}{36}\right) A^T U_2 B, \\ \Phi_{13} &= X, \Phi_{14} = 0, \Phi_{15} = P_2 + h_i R_2, \Phi_{16} = \Phi_{17} = h_\Delta R_4, \\ \Phi_{18} &= h_i P_4 + (h_i^2/2) U_1, \Phi_{19} = \Phi_{110} = h_\Delta P_5 + \left(\frac{(h_{i+1}^2 - h_i^2)}{2}\right) U_2, \\ \Phi_{22} &= h_i^2 B^T X_2 B - 2X_4 + h_\Delta^2 B^T X_4 B + \left(\frac{h_i^4}{4}\right) B^T R_2 B + \left(\frac{h_i^6}{36}\right) B^T U_1 B \\ &+ \left(\frac{(h_{i+1}^2 - h_i^2)^2}{4}\right) B^T R_4 B + \left(\frac{(h_{i+1}^3 - h_i^3)^2}{36}\right) B^T U_2 B, \\ \Phi_{23} &= \Phi_{24} = X_4, \Phi_{25} = \Phi_{26} = \Phi_{27} = \Phi_{28} = \Phi_{29} = \Phi_{210} = 0, \\ \Phi_{33} &= -Q_1 + Q_2 - X_2 - X_4, \Phi_{34} = 0, \Phi_{35} = -P_2, \Phi_{36} = \Phi_{37} = P_3, \\ \Phi_{38} &= \Phi_{39} = \Phi_{310} = 0, \Phi_{44} = -Q_2 - X_4, \Phi_{45} = 0, \Phi_{46} = \Phi_{47} = -P_3, \\ \Phi_{48} &= \Phi_{49} = \Phi_{410} = 0, \Phi_{55} = -X_1 - R_2, \Phi_{56} = \Phi_{57} = 0, \Phi_{58} = -P_4, \\ \Phi_{59} &= \Phi_{510} = 0, \Phi_{66} = -X_3 - R_4, \Phi_{67} = \Phi_{68} = 0, \Phi_{69} = \Phi_{610} = -P_5, \\ \Phi_n &= -X_3 - R_4, \Phi_{78} = 0, \Phi_{79} = \Phi_{710} = -P_5, \Phi_{88} = -R_1 - U_1, \\ \Phi_{89} &= \Phi_{810} = 0, \Phi_{99} = -R_3 - U_2, \Phi_{910} = U_2, \Phi_{1010} = -R_3 - U_2, \\ h_\Delta &= h_{i+1} = h_i = \frac{(h_M - h_m)}{N}, h_i = h_1 + \frac{(i-1)(h_M - h_m)}{N}. \end{aligned} \quad (10)$$

Proof. For the sake of simplicity, Theorem 1 holds when $h(t) \in [h_2, h_3]$ first; and then Theorem 1 is generalized to be established when $h(t) \in [h_i + h_{i+1}] (i = 1, 3, \dots, N)$.

When $h(t) \in [h_2, h_3]$, the L-K functional is constructed as follows:

$$V_2(x(t)) = V_{21}(x(t)) + V_{22}(x(t)) + V_{23}(x(t)) + V_{24}(x(t)) + V_{25}(x(t)), \quad (11)$$

where

$$\begin{aligned} V_{21}(x(t)) &= x^T(t)P_1x(t) + \int_{t-h_2}^t x^T(s)dsP_2 \int_{t-h_2}^t x(s)ds \\ &+ \int_{t-h_3}^{t-h_2} x^T(s)dsP_3 \int_{t-h_3}^{t-h_2} x(s)ds + \int_{-h_2}^0 \int_{t+\beta}^t x^T(s)dsd\beta P_4 \int_{-h_2}^0 \int_{t+\beta}^t x(s)dsd\beta \\ &+ \int_{-h_3}^{-h_2} \int_{t+\beta}^t x^T(s)dsd\beta P_5 \int_{-h_3}^{-h_2} \int_{t+\beta}^t x(s)dsd\beta, \\ V_{22}(x(t)) &= \int_{t-h_2}^t x^T(s)Q_1x(s)ds + \int_{t-h_3}^{t-h_2} x^T(s)Q_2x(s)ds, \\ V_{23}(x(t)) &= h_2 \int_{-h_2}^0 \int_{t+\beta}^t x^T(s)X_1x(s)dsd\beta + h_2 \int_{-h_2}^0 \int_{t+\beta}^t \dot{x}^T(s)X_2\dot{x}(s)dsd\beta \\ &+ (h_3 - h_2) \int_{-h_3}^{-h_2} \int_{t+\beta}^t x^T(s)X_3x(s)dsd\beta + (h_3 - h_2) \int_{-h_3}^{-h_2} \int_{t+\beta}^t \dot{x}^T(s)X_4\dot{x}(s)dsd\beta, \\ V_{24}(x(t)) &= \left(\frac{h_2^2}{2}\right) \int_{-h_2}^0 \int_{\beta}^0 \int_{t+\lambda}^t x^T(s)R_1x(s)dsd\lambda d\beta \\ &+ \left(\frac{h_2^2}{2}\right) \int_{-h_2}^0 \int_{\beta}^0 \int_{t+\lambda}^t \dot{x}^T(s)R_2\dot{x}(s)dsd\lambda d\beta \\ &+ \left(\frac{(h_3^2 - h_2^2)}{2}\right) \int_{-h_3}^{-h_2} \int_{\beta}^0 \int_{t+\lambda}^t x^T(s)R_3x(s)dsd\lambda d\beta \\ &+ \left(\frac{(h_3^2 - h_2^2)}{2}\right) \int_{-h_3}^{-h_2} \int_{\beta}^0 \int_{t+\lambda}^t \dot{x}^T(s)R_4\dot{x}(s)dsd\lambda d\beta, \\ V_{25}(x(t)) &= \left(\frac{h_2^3}{6}\right) \int_{-h_2}^0 \int_{\beta}^0 \int_{\lambda}^0 \int_{t+\varphi}^t \dot{x}^T(s)U_1\dot{x}(s)dsd\varphi d\lambda d\beta \\ &+ \left(\frac{(h_3^3 - h_2^3)}{6}\right) \int_{-h_3}^{-h_2} \int_{\beta}^0 \int_{\lambda}^0 \int_{t+\varphi}^t \dot{x}^T(s)U_2\dot{x}(s)dsd\varphi d\lambda d\beta. \end{aligned} \quad (12)$$

The derivative of L-K functional $V(t)$ along the nominal system (8) is calculated as follows:

$$\dot{V}_2(t) = \dot{V}_{21}(t) + \dot{V}_{22}(t) + \dot{V}_{23}(t) + \dot{V}_{24}(t) + \dot{V}_{25}(t), \quad (13)$$

where

$$\begin{aligned}
\dot{V}_{21}(t) &= 2x^T(t)A^T P_1 x(t) + 2x^T(t-h(t))B^T P_1 x(t) + 2x^T(t)P_2 \int_{t-h_2}^t x(s)ds \\
&\quad - 2x^T(t-h_2)P_2 \int_{t-h_2}^t x(s)ds + 2x^T(t-h_2)P_3 \int_{t-h_3}^{t-h_2} x(s)ds \\
&\quad - 2x^T(t-h_3)P_3 \int_{t-h_3}^{t-h_2} x(s)ds - 2 \int_{t-h_2}^t x^T(s)ds P_4 \int_{-h_2}^0 \int_{t+\beta}^t x(s)dsd\beta \\
&\quad + 2h_2 x^T(t)P_4 \int_{-h_2}^0 \int_{t+\beta}^t x(s)dsd\beta + 2(h_3-h_2)x^T(t)P_5 \int_{-h_3}^{-h_2} \int_{t+\beta}^t x(s)dsd\beta \\
&\quad - 2 \int_{t-h_3}^{t-h_2} x^T(s)ds P_5 \int_{-h_3}^{-h_2} \int_{t+\beta}^t x(s)dsd\beta, \\
\dot{V}_{22}(t) &= x^T(t)Q_1 x(t) - x^T(t-h_2)Q_1 x(t-h_2) + x^T(t-h_2)Q_2 x(t-h_2) - x^T(t-h_3)Q_2 x(t-h_3), \\
\dot{V}_{23}(t) &= h_2^2 x^T(t)X_1 x(t) - h_2 \int_{t-h_2}^t x^T(s)X_1 x(s)ds - h_2 \int_{t-h_2}^t \dot{x}^T(s)X_2 \dot{x}(s)ds \\
&\quad + h_2^2 \dot{x}^T(t)X_2 \dot{x}(t) + (h_3-h_2)^2 x^T(t)X_3 x(t) + (h_3-h_2)^2 \dot{x}^T(t)X_4 \dot{x}(t) \\
&\quad - (h_3-h_2) \int_{t-h_3}^{t-h_2} x^T(s)X_3 x(s)ds - (h_3-h_2) \int_{t-h_3}^{t-h_2} \dot{x}^T(s)X_4 \dot{x}(s)ds, \\
\dot{V}_{24}(t) &= \left(\frac{h_2^4}{4}\right) x^T(t)R_1 x(t) - \left(\frac{h_2^2}{2}\right) \int_{-h_2}^0 \int_{t+\beta}^t x^T(s)R_1 x(s)dsd\beta \\
&\quad + \left(\frac{h_2^4}{4}\right) \dot{x}^T(t)R_2 \dot{x}(t) - \left(\frac{h_2^2}{2}\right) \int_{-h_2}^0 \int_{t+\beta}^t \dot{x}^T(s)R_2 \dot{x}(s)dsd\beta \\
&\quad + \left(\frac{(h_3^2-h_2^2)^2}{4}\right) x^T(t)R_3 x(t) - \left(\frac{(h_3^2-h_2^2)}{2}\right) \int_{-h_3}^{-h_2} \int_{t+\beta}^t x^T(s)R_3 x(s)dsd\beta \\
&\quad + \left(\frac{(h_3^2-h_2^2)^2}{4}\right) \dot{x}^T(t)R_4 \dot{x}(t) - \left(\frac{(h_3^2-h_2^2)}{2}\right) \int_{-h_3}^{-h_2} \int_{t+\beta}^t \dot{x}^T(s)R_4 \dot{x}(s)dsd\beta, \\
\dot{V}_{25}(t) &= \left(\frac{h_2^6}{36}\right) \dot{x}^T(t)U_1 \dot{x}(t) + \left(\frac{(h_3^3-h_2^3)^2}{36}\right) \dot{x}^T(t)U_2 \dot{x}(t) \\
&\quad - \left(\frac{h_2^3}{6}\right) \int_{-h_2}^0 \int_{\beta}^0 \int_{t+\lambda}^t \dot{x}^T(s)U_1 \dot{x}(s)dsd\lambda d\beta - \left(\frac{(h_3^3-h_2^3)}{6}\right) \int_{-h_3}^{-h_2} \int_{\beta}^0 \int_{t+\lambda}^t \dot{x}^T(s)U_2 \dot{x}(s)dsd\lambda d\beta.
\end{aligned} \tag{14}$$

From Lemmas 1 and 2, we can obtain the following:

$$-h_2 \int_{t-h_2}^t x^T(s) X_1 x(s) ds \leq -\zeta^T(t) e_5 X_1 e_5^T \zeta(t), \quad (15)$$

$$\begin{aligned} & -h_2 \int_{t-h_2}^t \dot{x}^T(s) X_2 \dot{x}(s) ds \leq -\zeta^T(t) (e_1 - e_3) X_2 (e_1^T - e_3^T) \zeta(t) - \\ & \cdot 3\zeta^T(t) \left(e_1 + e_3 - \left(\frac{2}{h_2} \right) e_5 \right) X_2 \left(e_1^T + e_3^T - \left(\frac{2}{h_2} \right) e_5^T \right) \zeta(t), \end{aligned} \quad (16)$$

where $\zeta(t)$ is consistent with $i = 2$ in Lemma 3.

From Lemma 3, we can obtain the following:

$$\begin{aligned} & -(h_3 - h_2) \int_{t-h_3}^{t-h_2} x^T(s) X_3 x(s) ds \leq -\zeta^T(t) e_7 X_3 e_7^T \zeta(t) \\ & -\zeta^T(t) e_6 X_3 e_6^T \zeta(t) - \alpha \zeta^T(t) e_7 X_3 e_7^T \zeta(t) - (1 - \alpha) \zeta^T(t) e_6 X_3 e_6^T \zeta(t). \end{aligned} \quad (17)$$

Similarly, according to Lemma 3, we can obtain the following:

$$\begin{aligned} & -(h_3 - h_2) \int_{t-h_3}^{t-h_2} \dot{x}^T(s) X_4 \dot{x}(s) ds \leq -\zeta^T(t) (e_2 - e_4) X_4 (e_2^T - e_4^T) \zeta(t) \\ & -\zeta^T(t) (e_3 - e_2) X_4 (e_3^T - e_2^T) \zeta(t) - \alpha \zeta^T(t) (e_2 - e_4) X_4 (e_2^T - e_4^T) \zeta(t) \\ & - (1 - \alpha) \zeta^T(t) (e_3 - e_2) X_4 (e_3^T - e_2^T) \zeta(t), \end{aligned} \quad (18)$$

$$-\left(\frac{h_2^2}{2} \right) \int_{-h_2}^0 \int_{t+\beta}^t x^T(s) R_1 x(s) ds d\beta \leq -\zeta^T(t) e_8 R_1 e_8^T \zeta(t), \quad (19)$$

$$-\left(\frac{h_2^2}{2} \right) \int_{-h_2}^0 \int_{t+\beta}^t \dot{x}^T(s) R_2 \dot{x}(s) ds d\beta \leq -\zeta^T(t) (h_2 e_1 - e_5) R_2 (h_2 e_1^T - e_5^T) \zeta(t), \quad (20)$$

$$\begin{aligned} & -\left(\frac{(h_3^2 - h_2^2)}{2} \right) \int_{-h_3}^{-h_2} \int_{t+\beta}^t x^T(s) R_3 x(s) ds d\beta \leq -\zeta^T(t) e_{10} R_3 e_{10}^T \zeta(t) \\ & -\zeta^T(t) e_9 R_3 e_9^T \zeta(t) - \varepsilon \zeta^T(t) e_{10} R_3 e_{10}^T \zeta(t) - (1 - \varepsilon) \zeta^T(t) e_9 R_3 e_9^T \zeta(t), \end{aligned} \quad (21)$$

$$\begin{aligned} & -\left(\frac{(h_3^2 - h_2^2)}{2} \right) \int_{-h_3}^{-h_2} \int_{t+\beta}^t \dot{x}^T(s) R_4 \dot{x}(s) ds d\beta \leq \\ & -\zeta^T(t) ((h_3 - h_2) e_1 - e_7) R_4 ((h_3 - h_2) e_1^T - e_7^T) \zeta(t) \\ & -\zeta^T(t) ((h_3 - h_2) e_1 - e_6) R_4 ((h_3 - h_2) e_1^T - e_6^T) \zeta(t) \\ & -\varepsilon \zeta^T(t) ((h_3 - h_2) e_1 - e_7) R_4 ((h_3 - h_2) e_1^T - e_7^T) \zeta(t) \\ & - (1 - \varepsilon) \zeta^T(t) ((h_3 - h_2) e_1 - e_6) R_4 ((h_3 - h_2) e_1^T - e_6^T) \zeta(t), \end{aligned} \quad (22)$$

$$\begin{aligned}
& - \left(\frac{h_2^3}{6} \right) \int_{-h_2}^0 \int_{\beta}^0 \int_{t+\lambda}^t \dot{x}^T(s) U_1 \dot{x}(s) ds d\lambda d\beta \leq \\
& - \zeta^T(t) \left(\left(\frac{h_2^2}{2} \right) e_1 - e_8 \right) U_1 \left(\left(\frac{h_2^2}{2} \right) e_1^T - e_8^T \right) \zeta(t),
\end{aligned} \tag{23}$$

$$\begin{aligned}
& - \left(\frac{(h_3^3 - h_2^3)}{6} \right) \int_{-h_3}^{-h_2} \int_{\beta}^0 \int_{t+\lambda}^t \dot{x}^T(s) U_2 \dot{x}(s) ds d\lambda d\beta \leq \\
& - \zeta^T(t) \left(\left(\frac{(h_3^2 - h_2^2)}{2} \right) e_1 - e_9 - e_{10} \right) U_2 \left(\left(\frac{(h_3^2 - h_2^2)}{2} \right) e_1^T - e_9^T - e_{10}^T \right) \zeta(t).
\end{aligned} \tag{24}$$

Substituting (15)~(24) into (13), $\dot{V}_2(x(t))$ can be expressed as follows: where

$$\dot{V}_2(x(t)) \leq \zeta^T(t) [\alpha\Gamma_1 + (1-\alpha)\Gamma_2 + \varepsilon\Gamma_3 + (1-\varepsilon)\Gamma_4] \zeta(t), \tag{25}$$

$$\begin{aligned}
\Gamma_{i1} &= \left(\frac{\Phi}{2} \right) - e_7 X_3 e_7^T - (e_2 - e_4) X_4 (e_2^T - e_4^T), \\
\Gamma_{i2} &= \left(\frac{\Phi}{2} \right) - e_6 X_3 e_6^T - (e_3 - e_2) X_4 (e_3^T - e_2^T), \\
\Gamma_{i3} &= \left(\frac{\Phi}{2} \right) - e_{10} R_3 e_{10}^T - ((h_{i+1} - h_i) e_1 - e_7) R_4 ((h_{i+1} - h_i) e_1^T - e_7^T), \\
\Gamma_{i4} &= \left(\frac{\Phi}{2} \right) - e_9 R_3 e_9^T - ((h_{i+1} - h_i) e_1 - e_6) R_4 ((h_{i+1} - h_i) e_1^T - e_6^T).
\end{aligned} \tag{26}$$

For $0 \leq \alpha, \varepsilon \leq 1$, according to convex combination technique, the following inequality is established:

$$\begin{aligned}
\alpha(\Gamma_1 + \lambda_1 I) + (1-\alpha)(\Gamma_2 + \lambda_1 I) &< 0, \\
\varepsilon(\Gamma_3 + \lambda_2 I) + (1-\varepsilon)(\Gamma_4 + \lambda_2 I) &< 0.
\end{aligned} \tag{27}$$

Namely,

$$\alpha\Gamma_1 + (1-\alpha)\Gamma_2 < -\lambda_1 I, \tag{28}$$

$$\varepsilon\Gamma_3 + (1-\varepsilon)\Gamma_4 < \lambda_2 I. \tag{29}$$

As a result of $\lambda_1 > \lambda_2$, combining (28) and (29), the following formula is available:

$$\alpha\Gamma_1 + (1-\alpha)\Gamma_2 + \varepsilon\Gamma_3 + (1-\varepsilon)\Gamma_4 < (\lambda_2 - \lambda_1) I < 0. \tag{30}$$

If $\alpha\Gamma_1 + (1-\alpha)\Gamma_2 + \varepsilon\Gamma_3 + (1-\varepsilon)\Gamma_4 < 0$, according to L-K stability theorem, there exists a sufficient small positive number δ_2 for $\dot{V}_2(t) < -\delta_2 \|x(t)\|^2$ to hold, and then the nominal system (8) is asymptotically stable.

Without losing generality, when $h(t) \in [h_i, h_{i+1}]$ ($i = 1, 3, \dots, N$), the L-K function is constructed as follows:

$$\begin{aligned}
V_i(x(t)) &= V_{i1}(x(t)) + V_{i2}(x(t)) + V_{i3}(x(t)) \\
&\quad + V_{i4}(x(t)) + V_{i5}(x(t)),
\end{aligned} \tag{31}$$

where

$$\begin{aligned}
V_{i1}(x(t)) &= x^T(t)P_1x(t) + \int_{t-h_i}^t x^T(s)dsP_2 \int_{t-h_i}^t x(s)ds \\
&\quad + \int_{t-h_{i+1}}^{t-h_i} x^T(s)dsP_3 \int_{t-h_{i+1}}^{t-h_i} x(s)ds + \int_{-h_i}^0 \int_{t+\beta}^t x^T(s)dsd\beta P_4 \int_{-h_i}^0 \int_{t+\beta}^t x(s)dsd\beta \\
&\quad + \int_{-h_{i+1}}^{t-h_i} \int_{t+\beta}^t x^T(s)dsd\beta P_5 \int_{-h_{i+1}}^{t-h_i} \int_{t+\beta}^t x(s)dsd\beta, \\
V_{i2}(x(t)) &= \int_{t+\beta}^t x^T(s)Q_1x(s)ds + \int_{t-h_{i+1}}^{h_i} x^T(s)Q_2x(s)ds, \\
V_{i3}(x(t)) &= h_i \int_{-h_i}^0 \int_{t+\beta}^t x^T(s)X_1x(s)dsd\beta + h_i \int_{-h_i}^0 \int_{t+\beta}^t \dot{x}^T(s)X_2\dot{x}(s)dsd\beta \\
&\quad + (h_{i+1} - h_i) \int_{-h_{i+1}}^{t-h_i} \int_{t+\beta}^t x^T(s)X_3x(s)dsd\beta + (h_{i+1} - h_i) \int_{-h_{i+1}}^{t-h_i} \int_{t+\beta}^t \dot{x}^T(s)X_4\dot{x}(s)dsd\beta, \\
V_{i4}(x(t)) &= \left(\frac{h_i^2}{2}\right) \int_{-h_i}^0 \int_{\beta}^0 \int_{t+\lambda}^t x^T(s)R_1x(s)dsd\lambda d\beta + \left(\frac{h_i^2}{2}\right) \int_{-h_i}^0 \int_{\beta}^0 \int_{t+\lambda}^t \dot{x}^T(s)R_2\dot{x}(s)dsd\lambda d\beta \\
&\quad + \left(\frac{h_{i+1}^2 - h_i^2}{2}\right) \int_{-h_{i+1}}^{t-h_i} \int_{\beta}^0 \int_{t+\lambda}^t x^T(s)R_3x(s)dsd\lambda d\beta + \left(\frac{h_{i+1}^2 - h_i^2}{2}\right) \int_{-h_{i+1}}^{t-h_i} \int_{\beta}^0 \int_{t+\lambda}^t \dot{x}^T(s)R_4\dot{x}(s)dsd\lambda d\beta, \\
V_{i5}(x(t)) &= \left(\frac{h_i^3}{6}\right) \int_{-h_i}^0 \int_{\beta}^0 \int_{\lambda}^0 \int_{t+\varphi}^t x^T(s)U_1\dot{x}(s)dsd\varphi d\lambda d\beta + \left(\frac{h_{i+1}^3 - h_i^3}{6}\right) \int_{-h_{i+1}}^{t-h_i} \int_{\beta}^0 \int_{\lambda}^0 \int_{t+\varphi}^t \dot{x}^T(s)U_2\dot{x}(s)dsd\varphi d\lambda d\beta,
\end{aligned} \tag{32}$$

where the definition of $\zeta(t)$ is the same as that in Lemma 3. P_i ($i = 1, 2, 3, 4, 5$), Q_1 , Q_2 , Q_3 , U_1 , U_2 , X_j , and R_j ($j = 1, 2, 3, 4$) are the matrices defined in the same formula (9). The same method is available. The following conclusions can be reached by the same method:

$$\dot{V}_i(x(t)) \leq \zeta^T(t) [\alpha\Gamma_{i1} + (1 - \alpha)\Gamma_{i2} + \varepsilon\Gamma_{i3} + (1 - \varepsilon)\Gamma_{i4}] \zeta(t), \tag{33}$$

where

$$\begin{aligned}
\Gamma_{i1} &= \frac{\Phi}{2} - e_7X_3e_7^T - (e_2 - e_4)X_4(e_2^T - e_4^T), \\
\Gamma_{i2} &= \frac{\Phi}{2} - e_6X_3e_6^T - (e_3 - e_2)X_4(e_3^T - e_2^T), \\
\Gamma_{i3} &= \frac{\Phi}{2} - e_{10}R_3e_{10}^T - ((h_{i+1} - h_i)e_1 - e_7)R_4((h_{i+1} - h_i)e_1^T - e_7^T), \\
\Gamma_{i4} &= \frac{\Phi}{2} - e_9R_3e_9^T - ((h_{i+1} - h_i)e_1 - e_6)R_4((h_{i+1} - h_i)e_1^T - e_6^T).
\end{aligned} \tag{34}$$

In the same way, it is known that there exists a sufficient small positive number δ_i to make $\dot{V}_t(t) < -\delta\|x(t)\|^2$ hold, and then the nominal system (8) is asymptotically stable.

The combination of (25) and (33) is equivalent to (9). This fulfills the proof.

Remark 1. Firstly, different from [15], in which the delay range was divided into two equidistant subintervals, new LKF comprising quadruple-integral term and quadratic forms of double-integral term was constructed. In this paper, for each subinterval, the time-delay interval is divided into N

equal parts by using the method of time-delay partitioning. A new LKF with four integral terms is designed for each partitioned interval, and the quadratic form of double integral is introduced, such as $\iint x^T(s)dsd\beta M \iint x(s)dsd\beta$. Although the double integral functional term $\int_{-h}^0 \int_{t+\beta}^t x(s)dsd\beta$ is also used in [1, 10], it is not introduced into the definition of augmented vector. Secondly, the triple integral functional term integrand used in the new LKF contains the state vector x , and the lower bound information of the delay interval is introduced. Thanks to the coexistence of the quadratic integral functional term and the quadratic term $\int x^T(s)dsd\beta$, the conservativeness of the stability conclusion is significantly reduced.

Remark 2. In formula (9), the new stability criterion does not involve redundant free-weight matrices but skillfully uses Wirtinger-based integral inequality to define the cross terms generated by LKF derivatives and uses a few free matrices to represent the relationship between the relevant terms. Therefore, the complexity of theoretical derivation and computation is reduced, and the conservatism of conclusions is reduced.

Remark 3. For a given scalar μ and time-delay rate $\dot{h}(t)$ satisfying $0 < \dot{h}(t) \leq \mu$, substituting the functional term $\int_{t-h(t)}^t x^T(s)Q_3x(s)ds$ into the LKF constructed, the stability criterion containing the time-delay rate μ can be obtained according to the proof process of Theorem 1. The form is shown in Theorem 2.

Theorem 2. For the scalars h_m, h_M , and $\mu, \lambda_1, \lambda_2$ ($\lambda_1 > \lambda_2$), it is asymptotically stable for the nominal system (8), if there exist positive definite symmetric matrices P_i ($i = 1, 2, 3, 4, 5$), $Q_1, Q_2, Q_3, U_1, U_2, X_j$, and R_j ($j = 1, 2, 3, 4$), such that the following LMIs hold:

$$\tilde{\Phi} = (\tilde{\Phi}_{i,j})_{10 \times 10} < 0, \quad (35)$$

where $\tilde{\Phi}_{11} = \Phi_{11} + Q_3$ and $\tilde{\Phi}_{22} = \Phi_{22} - \mu Q_3$; other items in $\tilde{\Phi}$ are defined the same as in Φ Theorem 1.

Next, the robust stability of uncertain systems with interval time-varying delays (1) is considered.

Theorem 3. For the scalars $0 < h_m < h_M$ and $\mu, \lambda_1, \lambda_2$ ($\lambda_1 > \lambda_2$), it is asymptotically stable for the uncertain system (1), if there exist positive definite symmetric matrices P_i ($i = 1, 2, 3, 4, 5$), $Q_1, Q_2, Q_3, U_1, U_2, X_j$, and R_j ($j = 1, 2, 3, 4$), the scalar $\delta > 0$, and the free matrices with suitable dimension T_1, T_2 , such that the following LMIs hold:

$$\begin{bmatrix} \tilde{\Phi} & \Gamma_1 D & \delta \Gamma_2^T \\ * & -\delta I & 0 \\ * & * & -\delta I \end{bmatrix} < 0, \quad (36)$$

where

$$\begin{aligned} \Gamma_1 &= [T_1^T \ 0 \ 0 \ 0 \ 0 \ 0 \ T_2^T], \\ \Gamma_2 &= [E_a \ 0 \ E_b \ 0 \ 0 \ 0 \ 0]. \end{aligned} \quad (37)$$

Proof. For the uncertain system (1), A and B in equation (9) are replaced by $A + \Delta A$ and $B + \Delta B$, respectively. According to the proof of Theorem 1, the asymptotic stability of system (1) is obtained. This fulfills the proof.

4. Numerical Examples

The following three numerical examples are used to compare the results of the existing literature with the method proposed in this paper. MADB (Maximum Allowable Delay Bound) is defined as the upper bound of the maximum allowable delay to ensure the stability of the system, and it is the most common criterion to compare the conservativeness of the stability conclusions of time-delay systems.

Example 1. First consider the following closed-loop control systems with interval time-varying delays:

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ -1 & -2 \end{bmatrix} x(t) + \begin{bmatrix} 0 & 0 \\ -1 & 1 \end{bmatrix} x(t - h(t)). \quad (38)$$

For given h_m , according to (35) in Theorem 2 and (9) in Theorem 1, Tables 1 and 2 give corresponding MADB from two aspects, $\mu = 0.3$ and $\mu = \text{any}$, respectively. It can be clearly seen from Tables 1 and 2 that the method proposed in this paper is obviously better than the conclusion in the existing literature.

To verify the validity of the results, given $\mu = 0.3, h_m = 1$, and $h_M = 3.0796$ and given initial condition $x(t) = [2 \ -2]^T$, the state response curve of $x(t)$ is shown in Figure 1. It can be seen that the state trajectory of the above-mentioned system can quickly reach a stable state under the action of the obtained MADB, which further verifies the correctness of the proposed stability criterion.

Example 2. Uncertain systems with interval time-varying delays are considered:

$$\begin{aligned} \dot{x}(t) &= \begin{bmatrix} -2 + \lambda_1 & 0 \\ 0 & -1 + \lambda_1 \end{bmatrix} x(t) \\ &+ \begin{bmatrix} -1 + \lambda_3 & 0 \\ -1 & -1 + \lambda_4 \end{bmatrix} x(t - h(t)), \end{aligned} \quad (39)$$

where $\lambda_1, \lambda_2, \lambda_3$, and λ_4 are unknown parameters satisfying $|\lambda_1| \leq 1.6, |\lambda_2| \leq 0.05, |\lambda_3| \leq 0.1, |\lambda_4| \leq 0.3$.

For given h_m , according to (36) in Theorem 3, Table 3 gives corresponding MADB in the simulation. From the comparison results, it can be seen that, for this example, this method improves the conclusions of the existing literature.

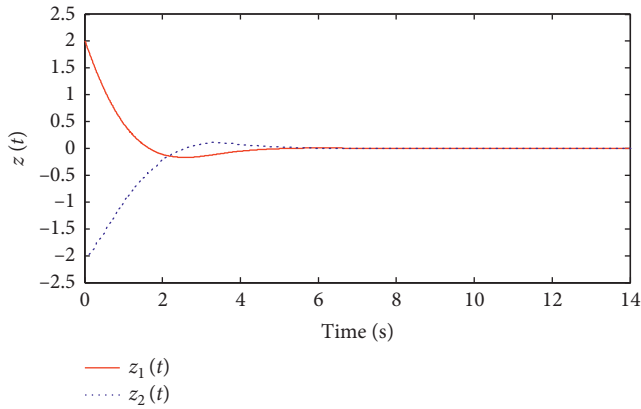
Given the initial condition $x(t) = [0.1 \ 0.2]^T$, the state response curve of $x(t)$ is shown in Figure 2 when the constant of time-varying delay $h(t)$ is 1.4723. When $h(t)$ takes variable $1.51 + 0.51 \sin t$, the state response curve of $x(t)$ is shown in Figure 3. It can be seen that $x(t)$ can quickly reach a stable state under the action of the nonlinear disturbance and the obtained MADB, thus verifying the correctness of the proposed stability criterion.

TABLE 1: In Example 1, MADB is simulated to be obtained for different h_m and different methods.

μ	Method	$h_m = 1$	$h_m = 2$	$h_m = 3$
0.3	Literature [24]	2.4042	2.5870	3.4766
	Literature [18]	2.4328	2.6322	—
	Literature [11] ($N=2$)	2.5278	3.0744	3.9136
	Literature [11] ($N=3$)	2.7368	3.4836	4.2857
	Literature [25]	3.16	3.50	4.32
	Theorem 2	3.0796	3.9064	4.4152

TABLE 2: In Example 1, when $\mu = any$, MADB is simulated to be obtained for different h_m and different methods.

Method	$h_m = 0.3$	$h_m = 0.5$	$h_m = 0.8$
Literature [13] ($N=2$)	1.1677	1.3078	1.5333
Literature [13] ($N=4$)	1.2043	1.3429	1.5633
Literature [18]	1.3531	1.4663	1.6592
Literature [26]	1.4347	1.5336	1.7140
Literature [14]	1.6837	1.8120	2.0209
Literature [27]	1.78	1.81	1.90
Theorem 1	1.9236	2.1384	2.2473

FIGURE 1: State response curve of $\mathbf{x}(t)$ when $h_M = 3.0796$.

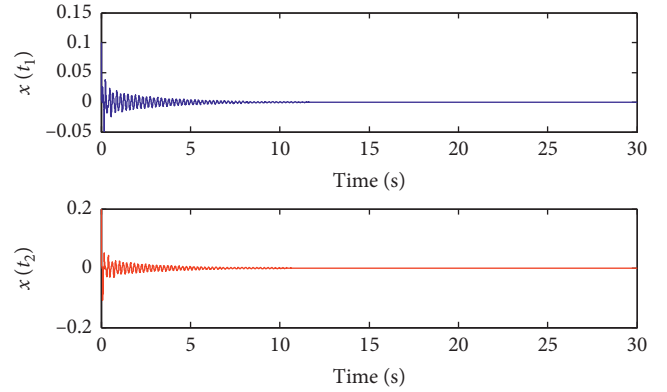
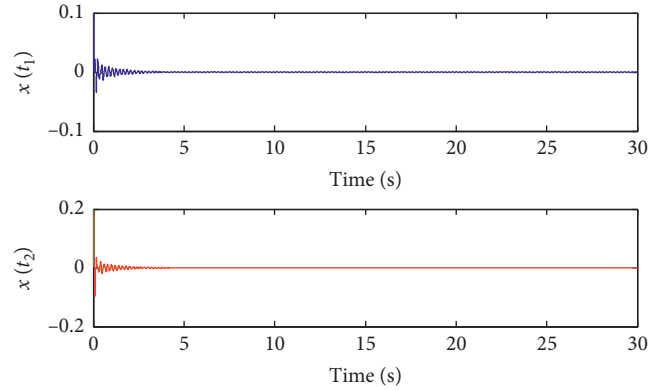
Example 3. Consider another uncertain system with interval time-varying delays. The system parameters are as follows:

$$\begin{aligned}
 A &= \begin{bmatrix} -0.4 & 0 \\ 0 & -1 \end{bmatrix}, \\
 B &= \begin{bmatrix} -0.9 & 0 \\ -1 & -0.7 \end{bmatrix}, \\
 D &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \\
 E_{a=} &= E_b = \begin{bmatrix} 0.2 & 0 \\ 0 & 0.2 \end{bmatrix}.
 \end{aligned} \tag{40}$$

Similarly, according to (36) in Theorem 3, for given h_m and $\mu = any$, Table 4 gives corresponding MADB in the simulation. As can be seen from Table 4, the robust stability theorem proposed in this paper enlarges the upper bound of

TABLE 3: In Example 2, MADB is simulated to be obtained for different h_m and different methods.

Method	$h_m = 0.2$	$h_m = 0.4$	$h_m = 0.6$
Literature [10] ($N=2$)	1.1337	1.1703	1.2123
Literature [28] ($N=2$)	1.1783	1.2123	1.2527
Literature [28] ($N=4$)	1.1871	1.2246	1.2686
Literature [11] ($N=2$)	1.3369	1.3571	1.3817
Literature [11] ($N=3$)	1.3809	1.4003	1.4216
Theorem 3	1.4241	1.4413	1.4723

FIGURE 2: State response curve of $\mathbf{x}(t)$ when $h(t) = 1.4723$.FIGURE 3: State response curve of $\mathbf{x}(t)$ when $h(t) = 1.51 + 0.51 \sin t$.TABLE 4: In Example 3, when $\mu = any$, MADB is simulated to be obtained for different h_m and different methods.

Method	$h_m = 0$	$h_m = 0.4$	$h_m = 0.8$
Literature [9]	1.0571	1.1385	1.2392
Literature [10] ($N=2$)	1.1030	1.1703	1.2594
Literature [11] ($N=2$)	1.3213	1.3571	1.4102
Literature [11] ($N=3$)	1.3634	1.4003	1.4445
Theorem 3	1.4127	1.4594	1.4987

the maximum allowable delay to guarantee the stability of the system. It has lower conservatism.

5. Conclusion

In this paper, we study the robust stability of a class of uncertain systems with interval time-varying delays. A new stability criterion based on LMI is proposed by constructing a new LKF containing a generalized term of quadruple integral. In order to improve the computational efficiency and simplify the conclusion, the criterion avoids the use of model transformation and free weight matrix definition techniques. Instead, Wirtinger-based integral inequalities and interactive convex combination techniques with tighter definition techniques are adopted, which make full use of the lower bound information of the delay and obtain a lower conservative conclusion. Finally, numerical simulations show that the proposed criterion enlarges the upper bound of the maximum delay allowed to guarantee the stability of the system and is more competitive than the existing methods.

However, the new stability criterion proposed in this paper mainly focuses on a class of linear systems with norm-bounded uncertainty and interval time-varying delay. How to get the similar conclusion for nonlinear system is another interesting topic and the next work for us; and some related researches are hopeful to supply reference to us [29, 30].

Data Availability

Three numerical examples are used to compare the results of the existing literature with the method proposed in this paper.

Conflicts of Interest

The authors declare that there are no conflicts of interest with respect to the research, authorship, and/or publication of this article.

Acknowledgments

This research was partially supported by the Young Scientists Fund of the National Natural Science Foundation of China (Grant no. 61503391), China Postdoctoral Fund under Grant no. 2017T100770, the Key Laboratory Fund under Grant no. 6142003190204, and the Special Scientific Research Program of Department of Education of Shaanxi Province (Grant no. 20JK0728).

References

- [1] A. Farnam and R. Mahboobi Esfanjani, "Improved stabilization method for networked control systems with variable transmission delays and packet dropout," *ISA Transactions*, vol. 53, no. 6, pp. 1746–1753, 2014.
- [2] W. Chen, S. Xu, Y. Li, and Z. Zhang, "Stability analysis of neutral systems with mixed interval time-varying delays and nonlinear disturbances," *Journal of the Franklin Institute*, vol. 357, no. 6, pp. 3721–3740, 2020.
- [3] J. Zhang, C. Peng, and M. Zheng, "Improved results for linear discrete-time systems with an interval time-varying input delay," *International Journal of Systems Science*, vol. 47, no. 2, pp. 492–499, 2015.
- [4] R. Mohajerpoor, L. Shanmugam, H. Abdi, R. Rakkiyappan, S. Nahavandi, and P. Shi, "New delay range-dependent stability criteria for interval time-varying delay systems via Wirtinger-based inequalities," *International Journal of Robust and Nonlinear Control*, vol. 28, no. 2, pp. 661–677, 2018.
- [5] P. Singkibud and K. Mukdasai, "On robust stability for uncertain neutral systems with non-differentiable interval time-varying discrete delay and nonlinear perturbations," *Asian-European Journal of Mathematics*, vol. 11, no. 1, pp. 2253–2261, 2018.
- [6] B. L. Zhang, L. H. Cheng, K. J. Pan, and X. M. Zhang, "Reducing conservatism of stability criteria for linear systems with time-varying delay using an improved triple-integral inequality," *Applied Mathematics & Computation*, vol. 380, 2020.
- [7] C. Shen, Y. Li, X. Zhu, and W. Duan, "Improved stability criteria for linear systems with two additive time-varying delays via a novel Lyapunov functional," *Journal of Computational and Applied Mathematics*, vol. 363, pp. 312–324, 2020.
- [8] K. Gu, V. L. Kharitonov, and J. Chen, *Stability of Time-Delay Systems*, Birkhäuser, Basel, Switzerland, 2003.
- [9] K. Ramakrishnan and G. Ray, "Delay-dependent robust stability criteria for linear uncertain systems with interval time varying delay," in *TENCON 2009-2009 IEEE Region 10 Conference*, IEEE, Singapore, January 2009.
- [10] K. Ramakrishnan and G. Ray, "Robust stability criteria for uncertain linear systems with interval time-varying delay," *Journal of Control Theory and Applications*, vol. 9, no. 4, pp. 559–566, 2011.
- [11] H. X. Zhang, J. J. Hui, X. Zhou, and G. L. Li, "New robust stability criteria for uncertain systems with interval time-varying delay based on delay-partitioning approach," *Control and Decision*, vol. 29, no. 5, pp. 907–912, 2014.
- [12] F. Gouaisbaut and A. Seuret, "Wirtinger-based integral inequality: application to time-delay systems," *Automatica*, vol. 49, no. 9, pp. 2860–2866, 2013.
- [13] X.-L. Zhu, G.-H. Wang, and G. H. Yang, "New stability criteria for continuous-time systems with interval time-varying delay," *IET Control Theory & Applications*, vol. 4, no. 6, pp. 1101–1107, 2010.
- [14] J. An, Z. Li, and X. Wang, "A novel approach to delay-fractional-dependent stability criterion for linear systems with interval delay," *ISA Transactions*, vol. 53, no. 2, pp. 210–219, 2014.
- [15] Y. B. Wu, H. X. Zhang, X. X. Hu, J. J. Hui, and G. L. Li, "Novel robust stability condition for uncertain neutral systems with mixed time-varying delays," *Advances in Mechanical Engineering*, vol. 9, no. 10, pp. 1–11, 2017.
- [16] Y. B. Li and X. Q. Xue, "Stability of uncertain neutral system with mixed time delays based on reciprocally convex combination approach," *Control and Decision*, vol. 31, no. 6, pp. 1105–1110, 2016.
- [17] A. Farnam and R. Mahboobi Esfanjani, "Improved linear matrix inequality approach to stability analysis of linear systems with interval time-varying delays," *Journal of Computational and Applied Mathematics*, vol. 294, pp. 49–56, 2016.
- [18] L. Ding, Y. He, M. Wu, and Z. Zhang, "A novel delay partitioning method for stability analysis of interval time-varying delay systems," *Journal of the Franklin Institute*, vol. 354, no. 2, pp. 1209–1219, 2017.
- [19] S. Senthilraj, R. Raja, Q. Zhu, R. Samidurai, and Z. Yao, "New delay-interval-dependent stability criteria for static neural

- networks with time-varying delays,” *Neurocomputing*, vol. 186, pp. 1–7, 2016.
- [20] J. Cheng, H. Wang, S. Chen, Z. Liu, and J. Yang, “Robust delay-derivative-dependent state-feedback control for a class of continuous-time system with time-varying delays,” *Neurocomputing*, vol. 173, pp. 827–834, 2016.
- [21] C.-K. Zhang, Y. He, L. Jiang, M. Wu, and H.-B. Zeng, “Stability analysis of systems with time-varying delay via relaxed integral inequalities,” *Systems & Control Letters*, vol. 92, pp. 52–61, 2016.
- [22] X.-H. Chang, J. Xiong, and J. H. Park, “Fuzzy robust dynamic output feedback control of nonlinear systems with linear fractional parametric uncertainties,” *Applied Mathematics and Computation*, vol. 291, pp. 213–225, 2016.
- [23] X. H. Chang, L. Zhang, and J. H. Park, “Robust static output feedback hinf control for uncertain fuzzy systems,” *Fuzzy Sets and Systems*, vol. 273, pp. 87–104, 2015.
- [24] L. V. Hien and H. Trinh, “An enhanced stability criterion for time-delay systems via a new bounding technique,” *Journal of the Franklin Institute*, vol. 352, no. 10, pp. 4407–4422, 2015.
- [25] W. Qian, Y. Gao, Y. Chen, and J. Yang, “The stability analysis of time-varying delayed systems based on new augmented vector method,” *Journal of the Franklin Institute*, vol. 356, no. 3, pp. 1268–1286, 2019.
- [26] O. M. Kwon, M. J. Park, J. H. Park, and S. M. Lee, “Enhancement on stability criteria for linear systems with interval time-varying delays,” *International Journal of Control, Automation and Systems*, vol. 14, no. 1, pp. 12–20, 2016.
- [27] P. G. Park, W. I. Lee, and S. Y. Lee, “Improved stability criteria for linear systems with interval time-varying delays: generalized zero equalities approach,” *Applied Mathematics and Computation*, vol. 292, pp. 336–348, 2017.
- [28] J.-J. Hui, H.-X. Zhang, X.-Y. Kong, and X. Zhou, “On improved delay-dependent robust stability criteria for uncertain systems with interval time-varying delay,” *International Journal of Automation and Computing*, vol. 12, no. 1, pp. 102–108, 2015.
- [29] L. Ma, G. Zong, X. Zhao, and X. Huo, “Observed-based adaptive finite-time tracking control for a class of nonstrict-feedback nonlinear systems with input saturation,” *Journal of the Franklin Institute*, vol. 357, no. 16, pp. 11518–11544, 2020.
- [30] L. Ma, N. Xu, X. Huo, and X. Zhao, “Adaptive finite-time output-feedback control design for switched pure-feedback nonlinear systems with average dwell time,” *Nonlinear Analysis: Hybrid Systems*, vol. 37, Article ID 100908, 2020.

Research Article

Locomotion Prediction for Lower Limb Prostheses in Complex Environments via sEMG and Inertial Sensors

Fang Peng ^{1,2}, Cheng Zhang ³, Bugong Xu ², Jiehao Li ⁴, Zhen Wang ⁴ and Hang Su ⁴

¹University of Electronic Science and Technology of China, Zhongshan Institute, Zhongshan 528402, China

²School of Automation Science and Engineering, South China University of Technology, Guangzhou 510641, China

³Department of Computer Science and Communications Engineering, Waseda University, Tokyo, Japan

⁴Department of Electronics, Information and Bioengineering, Politecnico di Milano, Milano 20133, Italy

Correspondence should be addressed to Hang Su; hang.su@polimi.it

Received 30 August 2020; Revised 15 October 2020; Accepted 22 November 2020; Published 4 December 2020

Academic Editor: Rongxin Cui

Copyright © 2020 Fang Peng et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Previous studies have shown that the motion intention recognition for lower limb prosthesis mainly focused on the identification of performed gait. However, the bionic prosthesis needs to know the next movement at the beginning of a new gait, especially in complex operation environments. In this paper, an upcoming locomotion prediction scheme via multilevel classifier fusion was proposed for the complex operation. At first, two motion states, including steady state and transient state, were defined. Steady-state recognition was backtracking of a completed gait, which would be used as prior knowledge of motion prediction. In steady-state recognition, surface electromyographic (sEMG) and inertial sensors were fused to improve recognition accuracy; five typical locomotion modes were recognized by random forest classifier with over 97.8% accuracy. The transient state was defined as an observation period at the initial stage of upcoming movement, in which only the sEMG signal was recorded due to the limitation of sliding window length. LightGBM classifier was validated to outperform other methods in the accuracy and prediction time of transient-state recognition. Finally, a simplified HMM model based on prior knowledge and observation result was constructed to predict upcoming locomotion. The results indicated that the locomotion prediction was over 91% accuracy. The proposed scheme implements the locomotion prediction at the initial stage of each gait and provides critical information for the gait control of lower limb prosthesis.

1. Introduction

The powered lower limb prosthesis, which can provide active torque for amputees and imitate the movement of healthy human leg better than a passive prosthesis, has been widely studied for a decade [1–4]. For developing powered lower limb prostheses, one major challenge is how to recognize the current locomotion mode of amputee and further identify the motion intention of amputee under complex scenarios [5–7] so as to realize the seamless transition of different locomotive tasks and then control lower limb prostheses with correct parameters. Otherwise, the actual trajectory or torque of the lower limb prostheses will deviate from expectations. For example, the actual terrain is upstairs, but the

prostheses controller still uses the control parameters in level walking, which will lead to discomfort for amputees and even the risk of falling down. Therefore, it is necessary to consider the suitable control method for safe locomotion.

Therefore, several sensors, especially electromyogram and mechanical sensors, have been widely used for locomotion detection. Surface electromyography (sEMG) signals can adequately represent the action state of the corresponding muscle group; many researchers used them as the only control input to recognize human locomotion [8–12]. Huang et al. [8] used the sEMG signals of two gluteal muscles and nine residual thigh muscles to realize the gait phase classification. However, sEMG signals are susceptible to skin sweat, muscle fatigue, and physical illness of subjects.

Thus, other researchers have tried to confirm that a set of different mechanical sensors could identify locomotion modes [13–15]. These mechanical sensors can be commonly classified into two groups: kinematic sensors and kinetic sensors. The former included angular, velocity, acceleration, etc. The latter included interaction force or torque between user and prosthesis (or environment). In [16], the sensor system consisted of an accelerometer and a gyroscope, respectively, located on the prosthetic pylon and two pressure sensors under the prosthetic foot. In [17], three six-axis IMUs were employed for identifying different locomotion. Although mechanical sensors are more easily embedded in lower limb prostheses than sEMG, they are challenging to achieve onset prediction alone [18]. Compared to the physical data from the prostheses or sound side leg, the sEMG signal can directly reflect the volitional control of the human body. A few studies have fused both mechanical information and sEMG signals to recognize locomotion modes. On the basis of the sEMG signals mentioned above [8], the authors added a 6-DOF load cell mounted on the prosthetic socket for better identifying continuous locomotion modes. Ai et al. [19] fused sEMG and accelerometer signals to classify five lower limb's motions. Additionally, Young et al. [20] compared the contribution of sEMG sensors and mechanical sensors embedded on powered prostheses and found that the recognition accuracy obtained by sEMG and inertial sensors (a six-axis inertial measurement unit (IMU) located on the shank) was significantly higher than that of other sensors. These results have indicated that multisensor fusion can obviously improve classification accuracy; especially, sEMG sensor and inertial sensor fusion is encouraging [21, 22]. Thus, the remaining problem is how to utilize the sensors to predict upcoming locomotion modes, which is essential for the powered prosthesis to actuate the artificial joints correctly.

In most previous work, the recognition of locomotion intention was accomplished by collecting and analyzing multisensor signals before the critical events of gait, such as heel contact or toe off, which denote the beginning of the stance phase or swing phase. At the same time, a lot of pattern recognition schemes have been studied in gait recognition. In [23], a transition period about 300–650 ms was reported before critical events for locomotion switch, and support vector machine (SVM) was used as a classifier. In another study [20], four 300 ms windows were extracted before critical events of the stance and swing phase, and a dynamic Bayesian network (DBN) classification algorithm was employed to recognize movement intention. In [24], the authors proposed a forward predictor to identify and respond to the user's intent, built an adaptive sEMG model, and added the label of backward estimation into forwarding predictor, and they extracted a 300 ms window of data before a gait event for forward prediction and used DBN and linear discriminant analysis (LDA) as forward and backward classifiers, respectively. However, these intention recognition methods mainly focused on the recognition of the performed gait or the transitional movement had been occurred. In fact, when the lower limb prosthesis led by the amputee's residual limb moves, whether the next phase is the

stance phase of weight acceptance of the swing phase of lifting leg, the lower limb prosthesis should know the upcoming movement before action. If an amputee is walking slowly or intermittently, it is not accurate to predict the next step based entirely on the prior gait.

In this paper, an intention prediction scheme in complex environment via sEMG and inertial sensors fusion is proposed to recognize five locomotion modes at gait initiations. We expanded the previous studies from the following aspects:

- (1) A multilevel classifier fusion strategy that combines prior gait information and current observation was proposed to infer the amputee's intent for switch control of lower limb prostheses.
- (2) The designed prediction scheme provided decision at the initiations of each stance phase and swing phase.
- (3) Surface EMG from four thigh muscles and inertial sensors located on thigh and shank were fused to recognize locomotion modes. The contribution of each sensor in gait recognition was studied.
- (4) The pattern recognition schemes, including two traditional algorithms and two ensemble learning algorithms, were analyzed. Besides, several feature extraction and reduction dimension methods are discussed.

2. Experiments and Data Processing

2.1. Experimental Protocol. The previous gait recognition experiment has been conducted on the able-bodied subjects [17, 25] or amputee subjects wearing prosthesis [26–28]. In this study, we recruited five able-bodied subjects between 20 to 50 years old, and one 31-year-old male subject with unilateral amputations was also recruited. The study was conducted following the ethical approval of confidential research involving human participants, and the protocol was approved by the University of Electronic Science and Technology of China, Zhongshan Institute (Project identification code is 2016A020220003).

Surface EMG signals were recorded from the key thigh muscles: rectus femoris, lateral thigh muscle, medial thigh muscle, and biceps femoris, which were verified to be effective for gait recognition in [1]. The locations for electrode placements are shown in Figure 1; the center spacing for electrodes is about 3 cm. The sEMG signals were collected from all subjects with a 16-channel EMG sensor (Myomove) and uploaded to the processing system with a sampling frequency of 1024 Hz. The processing system filtered sEMG signals with a comb filter and an IIR bandpass filter. The former was selected to offset the noise of the 50 Hz band. The latter signals are between 10 and 500 Hz. The inertial sensors including two nine-axis IMUs (Witmotion JY-901) were located on the thigh and shank, respectively. Each JY-901 integrated a triaxial gyroscope, a triaxial accelerometer, and a triaxial geomagnetic survey. With the addition of the geomagnetic sensor, JY-901 can obtain more accurate triaxial posture information than the six-axis IMU by

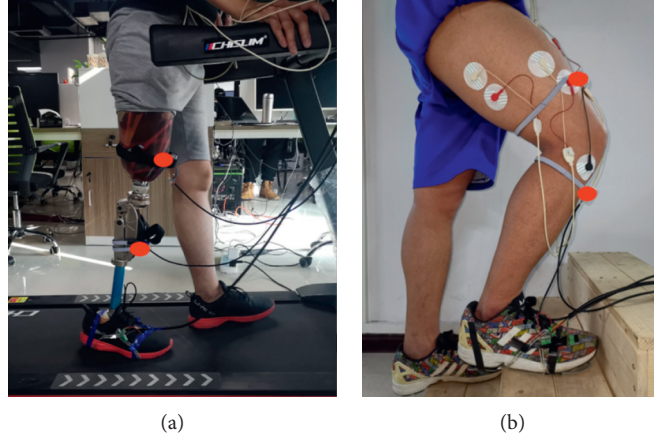


FIGURE 1: Experiments of different locomotion: (a) level walking; (b) stair ascend.

eliminating the accumulation error of direction angle. IMU signals were collected at 200 Hz and filtered with a dynamic Kalman filter.

A pressure sensor detected critical events of gait. Two pressure sensors (FlexiForce-A201) were placed, respectively, on the heel and the first metatarsal of the subject's forefoot. The pressure sensor information is uploaded to the processing system through an acquisition card, and the sampling frequency is 700 Hz.

After receiving instructions and training, each subject was asked to complete five locomotion modes at a natural, average speed. Level walking was tested on the treadmill. Ramp ascent and ramp descent were collected with the aid of a treadmill. We used the treadmill to build a ramp with a slope of about 20 degrees. All subjects were asked to walk on these three terrains three times, one minute at a time. Stair ascends and stair descends were tested on a 3-step staircase, as shown in Figure 1. Each subject was asked to go up and down the stairs 30 times. All subjects should rest for 5 minutes between different locomotion modes to avoid the influence of muscle fatigue on the experiment and ensure the objectivity and accuracy of the experimental data.

2.2. sEMG Signal Acquisition via Compressed Sensing.

Compressed sensing is a signal acquisition framework that could reduce the data storage much lower than that with conventional Nyquist sampling rate. The reduction in data storage or sampling rate could reduce the energy dissipation for the battery-powered wearable sEMG sensor. Moreover, the data deluge problem and the urgency in communication bandwidth could be lightened as well [9].

The work in [29] reported that the sEMG signal is sparser in the transform domain than that in the time domain, and the Daubechies wavelet basis can be chosen as the more appropriate selection for the sparse basis compared with DCT and Haar wavelet basis [30]. Under the framework of compressed sensing, the following linear equation holds:

$$y = \Phi\Psi s + n = Hs + n, \quad (1)$$

where $y \in \mathbb{R}^{M \times 1}$ and $s \in \mathbb{R}^{N \times 1}$ denote the undersampled sEMG measurement and the sparse representation of the sEMG signal in the Daubechies wavelet basis. Note $M < N$ is met under compressed sensing. $\Phi \in \mathbb{R}^{M \times N}$ and $\Psi \in \mathbb{R}^{N \times N}$ stand for the Bernoulli random sampling matrix and the Daubechies wavelet sparse basis, respectively, and $H = \Phi\Psi \in \mathbb{R}^{M \times N}$. $n \in \mathbb{R}^{M \times 1}$ represents the error term occurring from measurement noise and modeling error from sparse representation.

The sparse representation s is solved using the heterogeneous Bayesian compressed sensing (HBCS) algorithm, which has shown superior performances than orthogonal matching pursuit (OMP), basis pursuit (BP), and Bayesian compressed sensing (BCS) in our previous work [31–34]. In HBCS, all the unknowns are taken as random variables and follow certain probability distributions. Here each element of s is considered to be iid (identically independent distribution) and assumed to meet the following zero-mean Gaussian distribution, and the reciprocal of the variance in Gaussian distribution is then imposed by gamma distribution; the two-layer hierarchical prior distribution is shown as follows:

$$\begin{cases} P(s|\alpha) = \prod_{i=1}^N N(s_i|0, \alpha_i^{-1}), \\ P(\alpha|a, b) = \prod_{i=1}^N \text{Gamma}(\alpha_i|a, b), \end{cases} \quad (2)$$

where α_i^{-1} is the variance of the Gaussian distribution and a and b are the user-defined shape and scale parameters in the Gamma distribution. The noise n is also restricted by a two-layer hierarchical iid distribution, which is presented as

$$\begin{cases} P(n|\beta) = \prod_{i=1}^M N(n_i|0, \beta_i^{-1}), \\ P(\beta|c, d) = \prod_{i=1}^M \text{Gamma}(\beta_i|c, d), \end{cases} \quad (3)$$

where β_i^{-1} is the reciprocal of the noise variance and c and d are also used to characterize the shape and scale parameters in gamma distribution.

The conditional distribution is also allocated with Gaussian distribution as follows:

$$P(y|st; n\beta) = N(y|Htsn, qB^{-1}), \quad (4)$$

where $B = \text{diag}(\beta)$, $\beta = [\beta_1, \beta_2, \dots, \beta_M]^T$. The posterior estimation is obtained by seeking for the maximum-a-posterior (MAP) solution, which is expressed as

$$\begin{aligned} P(s|y, \alpha, \beta) &= \frac{P(y|s; \beta)P(s|\alpha)}{P(y, \alpha, \beta)}, \\ &= \frac{P(y|s; \beta)P(s|\alpha)}{\int P(y|s; \beta)P(s|\alpha)ds}, \\ &= (2\pi)^{-(M/2)} |\Sigma|^{-(1/2)} \exp\left[-\frac{1}{2}(s - \hat{s})^T \Sigma^{-1} (s - \hat{s})\right], \end{aligned} \quad (5)$$

where \hat{s} and Σ are represented as follows:

$$\begin{aligned} \hat{s} &= \Sigma H^T B y, \\ \Sigma &= [H^T B H + A]^{-1}, \end{aligned} \quad (6)$$

where $A = \text{diag}(\alpha)$, $\alpha = [\alpha_1, \alpha_2, \dots, \alpha_N]^T$ and $B = \text{diag}(\beta)$, $\beta = [\beta_1, \beta_2, \dots, \beta_M]^T$. For the hyperparameters α and β , they are learned from the measured sEMG signal by performing a type-II maximum likelihood (ML) or evidence procedure. The update formulas are expressed as follows:

$$\begin{cases} \alpha_i = \frac{\gamma_i + 2(a-1)}{\hat{s}_i^2 + 2b}, \\ \beta_i = \frac{1 + 2(c-1)}{(y_i - H_i \hat{s})^2 + \text{tr}(\Sigma H_i^T H_i) + 2d}, \end{cases} \quad (7)$$

where $\gamma_i = 1 - \alpha_i \Sigma_{ii}$ with Σ_{ii} representing the i th diagonal element of Σ , \hat{s}_i is the i th element of \hat{s} , and H_i is the i th row of H . Detailed derivation of the above Bayesian inference procedure has been omitted here (the readers can refer to [31, 35, 36] and the references therein).

After the sparse representation \hat{s} is reconstructed, the estimated sEMG signal x is then obtained via the following transform.

$$x = \Psi \hat{s}. \quad (8)$$

2.3. Data Acquisition. In this paper, we defined two states for collecting sensor data: steady state and transient state. The steady state referred to a completely performed gait. The transient state was a short transitional period when subjects began new gait. The division of the states depended on the identification of the gait phase by pressure signal, as shown in Figure 2(a). The stance phase began when the foot touched the ground (i.e., heel contact) and terminated at the foot off the ground (i.e., toe off). The swing phase was

from toe off to heel contact. Two different frames of the sliding window were designed to process the steady-state and the transient-state information separately. For steady-state recognition, i.e., state backtracking, which was used to identify performed gaits, a sliding window covering a full stride cycle was designed. The steady state was from the heel contact to next heel contact or from the toe off to next toe off, as shown in Figure 2(b), in which, sEMG and IMU sensors were fused to analyze performed locomotion mode. For transient state, the shorter the time of the data acquisition window, the better the smooth control of lower limb prosthesis. Therefore, a 50 ms small sliding window with only sEMG signal recorded was designed at every initial of the heel contact and toe off. Three reasons for choosing 50 ms sEMG signal were (1) the sEMG signal generally precedes mechanical signal at responding to movement onset; (2) it takes about 20 ~ 50 ms for human muscles to respond to an action potential; and (3) due to the limitation of IMU signal acquisition frequency, only several data could be recorded within 50 ms, which might lead to confusion for identification. During the transient state, the control system of the powered prosthesis also used the previous gait controller, which means that the response of the prosthesis was at least 50 ms slower than that of the healthy leg. This delay time is relatively short compared with the whole cycle, which will not have an adverse effect on the walking status of the powered prosthesis but can ensure the accuracy of the next state control.

2.4. Feature Extraction

2.4.1. Feature Extraction of sEMG Signals. To take into account the timeliness of signal processing, the time-domain feature extraction method for sEMG was considered in this work [37]. In preliminary work [12], over 20 features of sEMG signals were quantitatively compared, and the following enumerated features were optimal time-domain feature groups for gait recognition.

(i) Slope sign change (SSC).

SSC characterizes the frequency information of sEMG signals:

$$\text{SSC} = \sum_{i=2}^{N-1} f((x_i - x_{i+1}) * (x_i - x_{i-1})), \quad (9)$$

where $f(x) = \begin{cases} 1, & \text{if } x > Th_{\text{SSC}} \\ 0, & \text{if } x \leq Th_{\text{SSC}} \end{cases}$. We set the threshold Th_{SSC} to 40 mV.

(ii) Willison amplitude (WAMP).

WAMP refers to the number of times that the difference of sEMG signal amplitude between adjacent two points exceeds a predetermined threshold and is related to the level of muscle contraction.

$$\text{WAMP} = \sum_{i=1}^N \text{sgn}(|x_{i+1} - x_i| - Th_{\text{WAMP}}). \quad (10)$$

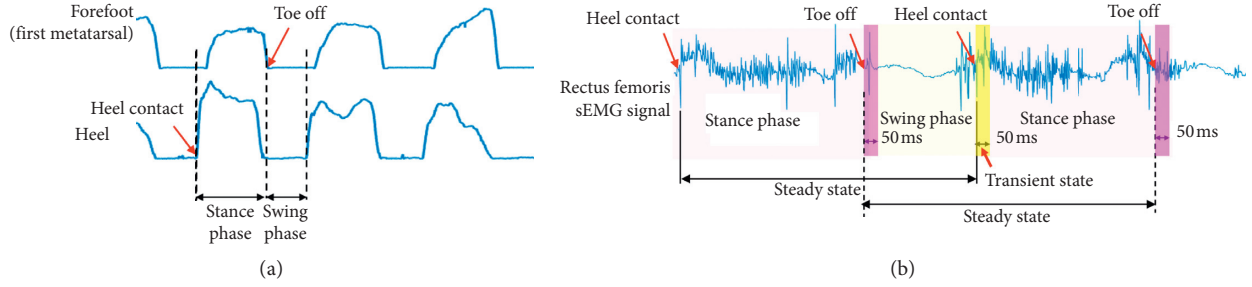


FIGURE 2: Data processing methods. (a) Gait phase detection by pressure sensors. (b) Sliding window scheme and definition.

The parameter Th_{WAMP} is optimized by experimental comparison, and we obtain the optimized Th_{WAMP} value of 50 mV.

(iii) Waveform length (WL).

Waveform length is the cumulative length of the waveform at a given time.

$$WL = \sum_{i=1}^{N-1} |x_{i+1} - x_i|. \quad (11)$$

(iv) Logarithmic variance (LogVAR).

Variance is the average of the deviation square of the variable. However, the average of the sEMG signal is close to zero. Therefore, the variance of sEMG is generally replaced by the following formula:

$$VAR = \frac{1}{N-1} \sum_{i=1}^N x_i^2. \quad (12)$$

In order to make the variance have better normal distribution characteristics, the logarithm of the variance was extracted as a new feature, which was defined as LogVAR.

2.4.2. Feature Extraction of IMU Signals. A multifeature fusion method is proposed to extract the feature vectors of IMU signals. This method can express the motion characteristics of lower limbs more comprehensively and effectively and provide a guarantee for the better training classification model.

(i) Discrete wavelet transform (DWT).

DWT was used to decompose the filtered IMUs data. After wavelet decomposition, the high- and low-frequency coefficients were extracted in time domain to obtain more comprehensive and clearer pattern information and to eliminate the influence of noise further. Wavelet decomposition decomposed the IMU signal into high-frequency details and low-frequency approximation. For gait information, the low-frequency component is quite important and contains the main characteristics of the signal, while the high-frequency component gives the details or differences of the signal. The approximate value $A_j[n]$ (i.e., low frequency part) and detail part $D_j[n]$ (i.e.,

high frequency part) of the decomposed original signal are formulated as

$$A_{j+1}[n] = \sum_k A_j[n]h(2n-k), D_{j+1}[n] = \sum_k A_j[n]g(2n-k), \quad (13)$$

where $h(\cdot)$ means scale function coefficient, $g(\cdot)$ presents wavelet function coefficient, and j represents the scale of decomposition.

By experimental comparison, Daubechies 9 was selected to decompose acceleration and angular velocity signals and extract the characteristic values of wavelet transform coefficients. Furthermore, the time-domain features of the high- and low-frequency coefficients were extracted, including absolute mean (MAV) and standard deviation (STD).

(ii) Cross-correlation coefficient.

The cross-correlation function can extract and analyze the correlation characteristics of different periodic signals or quasiperiodic signals [38, 39]. Therefore, in the human locomotion pattern recognition, the characteristic parameters of different locomotion modes can be represented by the correlation characteristics of different inertial signals.

The cross-correlation function describes the correlation between two random signals $X(t), Y(t)$ at any time. It is given by

$$R_{XY} = \frac{(1/n-1) \sum_{i=1}^n (X_i - \bar{X})(Y_i - \bar{Y})}{\sqrt{\sum_{i=1}^n (X_i - \bar{X})^2} \sqrt{\sum_{i=1}^n (Y_i - \bar{Y})^2}}, \quad (14)$$

where \bar{X} and \bar{Y} denote the average value of a sequence frame and n is the number of samples. We calculated the correlation coefficients of the acceleration and angular velocity between the thigh and shank as the features.

2.5. Feature Reduction Dimension. The feature extraction from the sEMG and IMU signals produced the data of 68 dimensions. To improve the characterization ability of features, we obtain more valuable information and reduce the computational cost of the classification algorithm, and a lower dimensionality should be acquired by dimension

reduction. In this paper, two common dimension reduction approaches were considered: principal component analysis (PCA) and linear discriminant analysis (LDA). PCA is the most classical and widely used dimension reduction algorithm in information fusion. The basic idea of PCA is to retain the main features in the original data (the covariance structure of the data) and map the data from the high-dimensional feature space to the low-dimensional feature space by linear projection. LDA is a supervised linear dimensionality reduction method, which considers the labels of classes and facilitates the discrimination of the data after dimensionality reduction. Both methods were proved to be effective in biometric pattern recognition [40–42].

3. Intention Recognition and Locomotion Prediction Method

Locomotion prediction for the powered lower prosthesis refers to make accurate decisions of upcoming human motion based on sensor information. In this paper, a multilevel classifier fusion scheme was proposed to predict the motion intention as shown in Figure 3. The locomotion prediction system combined the steady-state recognition with transient-state recognition. The former is the backtracking of completed gait, and the latter is a preliminary identification of the upcoming gait. Then, a HMM model as a decision fusion model was designed to fuse the recognition of the steady state and transient state, the result of which is the prediction of locomotion mode.

3.1. Steady-State Recognition. In order to improve the recognition accuracy, the recognizer of the steady state was accomplished based on the fusion of sEMG and inertial signals. A variety of classifiers have been approached for intent pattern recognition. In this paper, we evaluated the performance of four classification algorithms, which include support vector machine (SVM), quadratic discriminant analysis (QDA), light gradient boosting machine (LightGBM), and random forest (RF). The SVM is a machine learning algorithm based on the statistical learning theory. The classification performance of SVM has been proved to be better than that of LDA in sEMG pattern recognition for prosthetic legs [23]. QDA is a variant of LDA, allowing nonlinear separation of data with small computation and high efficiency. The QDA classification had higher classification accuracies than LDA in classifying intention of the knee motion. [43]. LightGBM and RF are both ensemble learning algorithms that train different classifiers (weak classifiers) and then assemble these weak classifiers to form a more reliable final classifier (robust classifier). Compared to the single model learning method, ensemble learning algorithms are more likely to obtain high accuracy and generalization. LightGBM is a distributed framework of the gradient boosting decision (GBDT) tree algorithm. It was proposed by the Microsoft team's Guolin Ke et al. in 2017 to solve the problem of GBDT computing efficiency. Compared with other traditional classifiers, the LightGBM classifier performed well in the accuracy and prediction time of gait

phase recognition based on sEMG [44]. Random forest is also a conventional classification algorithm with a decision tree as the base learner. Random forest has a strong anti-interference ability and is more suitable for processing high-dimensional data than SVM. Random forest has been validated to recognize five types of locomotion of lower limbs and obtain better accuracy than SVM.

3.2. Transient-State Recognition. Transient state is defined as the observation period for a motion switch, during which the lower limb prosthesis detects the upcoming movement. Because of the real-time requirement, a challenge was put forward for recognition accuracy and time. If the recognition time is too long, the prosthetic locomotion mode will be inconsistent with the actual walking terrain, which will lead to the amputee walking unnaturally. In the four classification algorithms mentioned above, QDA and LGBM have the advantage of fast computation speed in a single classification model and ensemble learning model, respectively. All four classifiers were compared and analyzed for transient-state recognition.

3.3. Locomotion Prediction. Locomotion switch is a response on account of the actual walking terrain, while pedestrian facilities of roads or buildings have an extreme regularity. The transition of various terrain is not random. Generally, the ends of a flight of stairs are usually flat, not ramps. Similarly, the ends of a ramp are usually flat, too, not stairs. Therefore, there is a certain probability of human gait switch, as shown in Table 1. For example, the current gait is level walking (LW), and then the next gait is level walking (LW), ramp ascend (RA), ramp descend (RD), stair ascend (SA), or stair descend (SD) with a probability of 20% each. If the prior gait is SA, then the next gait is LW or SA, with the probability of 50% each.

In this way, the recognition of gait intention is transformed into a probabilistic model construction problem. The next gait state recognition process can be described as a typically hidden Markov model (HMM) with strong prior knowledge. HMM is a probability model about time sequence. For human walking, a transition from one gait to another gait is a hidden process that cannot be directly observed, but the observations in transient state can be used to infer the gait transition. In this paper, the first-order HMM was utilized to fuse the backtracking gait information and the transient-state information to achieve the final locomotion prediction.

4. Results and Discussion

4.1. Steady-State Recognition Performance. Four classifiers with PCA and LDA dimensional reduction were conducted based on sEMG and IMU fusion. Recognition evaluation was performed using 10-fold cross validation. A Bayesian method based on tree-structured Parzen estimator (TPE) was utilized to solve the hyperparameter optimization problem for RF and LightGBM classifiers. SVM and QDA parameters were optimized by the grid search method.

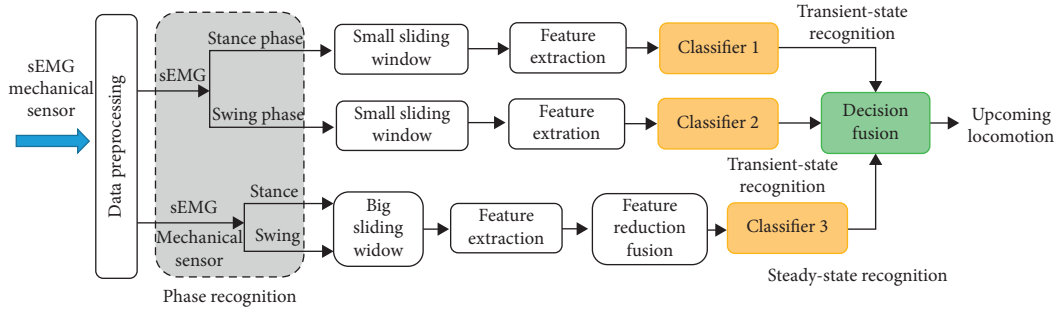
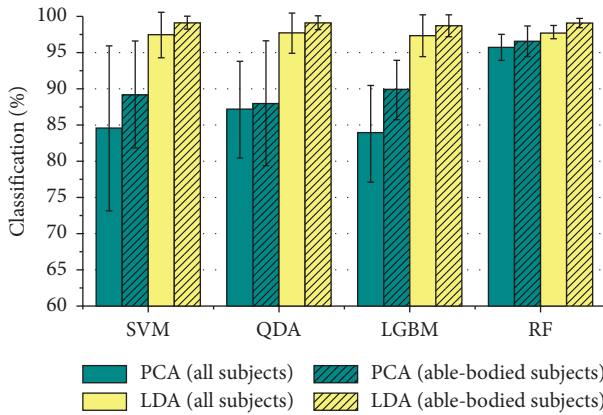


FIGURE 3: Architecture of the locomotion prediction scheme.

TABLE 1: Transition probability between locomotion modes.

	LW	RA	RD	SA	SD
LW	20	20	20	20	20
RA	50	50	0	0	0
RD	50	0	50	0	0
SA	50	0	0	50	0
SD	50	0	0	0	50

Note. LW, level walking; RA, ramp ascend; RD, ramp descend; SA, stair ascend; SD, stair descend.

FIGURE 4: Steady-state accuracy with different dimension reduction methods and classifiers in different subject groups. Error bars represent \pm SD.

Different subject groups, including all subjects and subjects except amputee (i.e., only able-bodied subjects), were compared. The experimental results are shown in Figure 4. In view of the dimensionality reduction algorithm, although different methods had different effects on the classifiers, LDA was more favorable for the accuracy of the classifier than PCA. RF algorithm represented good fusion performance for both dimensional-reduction methods. After LDA dimensional reduction, the classification accuracy of four classifiers was all over 99% for 5 able-bodied subjects. When the data from the amputee subject were added, the accuracy dropped slightly by 2%, but it was also over 97.4%. That means there was a discrepancy in walking posture between a healthy person and a lower limb amputee. In general, LDA + RF was the optimal combination with the highest classification accuracy, 99.1% for five able-bodied subjects

and 97.8% for all six subjects. It is worth noting that all the following results are for all subjects.

The contribution of different sensors or sensor combinations in gait recognition of multisource information perception was analyzed. The IMU and sEMG signals were compared and analyzed, respectively. The result in terms of misclassification for all subjects was reported, as shown in Figure 5. For the steady-state recognition, the effect of using the IMU sensor alone was better than that of using the sEMG signal alone. The recognition accuracy of using IMUs on the thigh or shank alone was the range from 81.3% to 85%; after the combination of the two IMUs, the recognition rate was greatly improved to above 95%. It could be concluded that for improving the identification of lower limb movement identification, it was very significant to detect both the thigh and shank mechanical signals at the same time due to the difference in the movement of them. Furthermore, after the sEMG signal was combined with two IMU signals, the recognition rate continued to be improved by 1 ~ 2%. That means that the multisensor fusion is beneficial and useful for gait recognition.

Actually, different sensor signals had different effects on recognition algorithms. When only using the sEMG signal, the error rate of LDA + LGBM and LDA + RF was lower than the others. However, whatever sensor signals were used, the LDA + RF method had the lowest classification error. From the aforementioned results, there was an essential correlation between the selection of human gait recognition methods and the sensors used in the detection.

The confusion matrix from four classifiers was derived from analyzing the specific situation of the misclassification. It can be seen from Figure 6 that the accuracy of four classifiers in distinguishing five locomotion modes was in the range of 95 ~ 100%. The misclassification of all classifiers mainly occurred between the level walking and ramp ascend.

4.2. Transient-State Recognition Performance. The 50 ms sEMG signals of 4 channels were collected from the heel contact (i.e., beginning of stance phase) and toe off (i.e., beginning of swing phase) moment, respectively. The features of SSC, WAMP, WL, and LogVAR were extracted for classification recognition. After repeated verification, the feature dimensionality reduction could not significantly improve the classification accuracy. Consequently, classifiers were directly trained by utilizing the features data, which

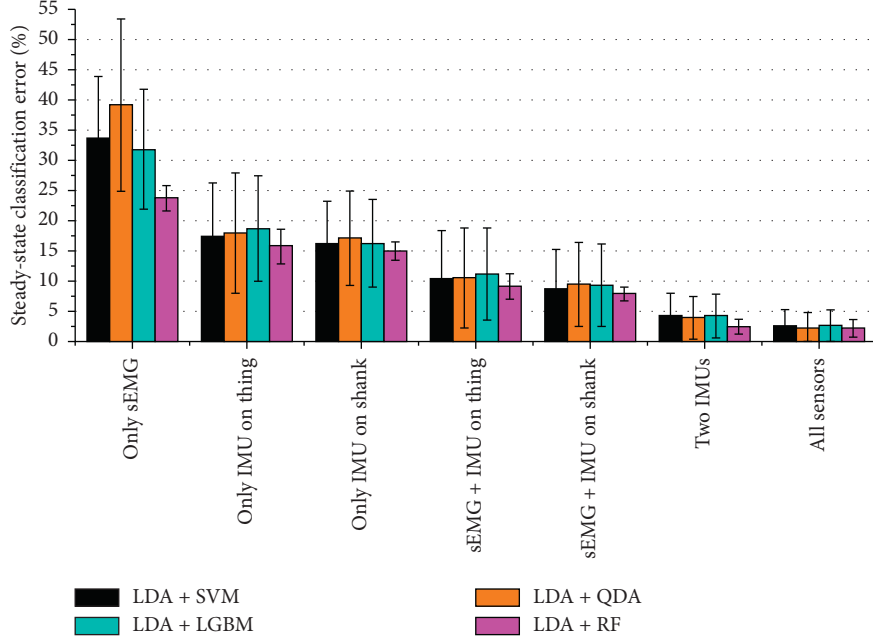


FIGURE 5: Contributions of different sensors and sensor combinations. Error bars represent \pm SD.

would obviously shorten the data processing time. Using 10-fold cross validation, the average prediction accuracy and prediction time are shown in Table 2. Whether in the preswing or the prestance, the recognition accuracy of the LightGBM algorithm was higher than that of other classifiers. Even though the QDA algorithm had the shortest prediction time, the accuracy was the lowest. On account of gradient-based one-side sampling (GOSS) and exclusive feature bundling (EFB) techniques, LightGBM has a faster speed without lowering its accuracy. The average classification accuracy of different types of locomotion in transient state is shown in Figures 7 and 8. On the whole, the ramp ascend movement can be most accurately detected. When moving up or down the stairs, the sEMG signal at the prestance phase could be distinguished more easily, probably because the muscle stretching and contraction are more obvious in this stage.

Furthermore, we take the confusion matrix of LightGBM recognition in preswing as an example to analyze the distribution of misidentification. The confusion matrix is described in Table 3. It can be seen that stair descent movement had the highest classification error in the preswing phase, 23% of which were misclassified as level walking, 3% as ramp ascend, and 8% as stair ascend. Compared with the prestance phase, it is much more difficult to detect stair descend movement at the preswing.

4.3. Results and Deficiencies of Locomotion Prediction. A simplified HMM model was constructed combining the prior knowledge of steady-state and transient-state observation, and it was defined as follows:

$$\lambda = (S, O, \Pi, A, B), \quad (15)$$

where S denotes the set of hidden states, which refers to the five states of LW, RA, RD, SA, and SD. O is observation sequence, which is obtained from transient-state classifier. $\Pi = [\pi_1, \pi_2, \pi_3, \pi_4, \pi_5]$, where π_i represents the initial probability of the i th (1-LW, 2-RA, 3-RD, 4-SA, 5-SD) state. π_i is set to 0.2 at the beginning of a new cycle, but reset according to steady-state recognition in continuous gaits. $A = [a_{ij}]_{1 \leq i, j \leq 5}$ is the state transition probability matrix which can be generated from Table 1. $B = [b_{ij}]_{1 \leq i, j \leq 5}$ is the state observation probability matrix which can be inferred from Table 3. Next, we use Viterbi algorithm to solve the predictive problem of HMM. In this paper, the locomotion prediction only depends on the prior state and transient state, so the algorithm is simplified as the following two steps.

Step 1:

$$\delta_1(i) = \pi_i b_{ij}. \quad (16)$$

Step 2:

$$\begin{aligned} \delta_2(i) &= \max_{1 \leq j \leq 5} [\delta_1(a_{ji}) b_{ij}], \\ \psi &= \operatorname{argmax}_{1 \leq j \leq 5} [\delta_1(j) a_{ji}], \end{aligned} \quad (17)$$

where i denotes the number of states, j is the observation state, $\delta_{(i)}$ represents the probability of i th locomotion mode, and ψ refers to the upcoming gait with a value of 1, 2, 3, 4, 5, which is the final result for locomotion prediction.

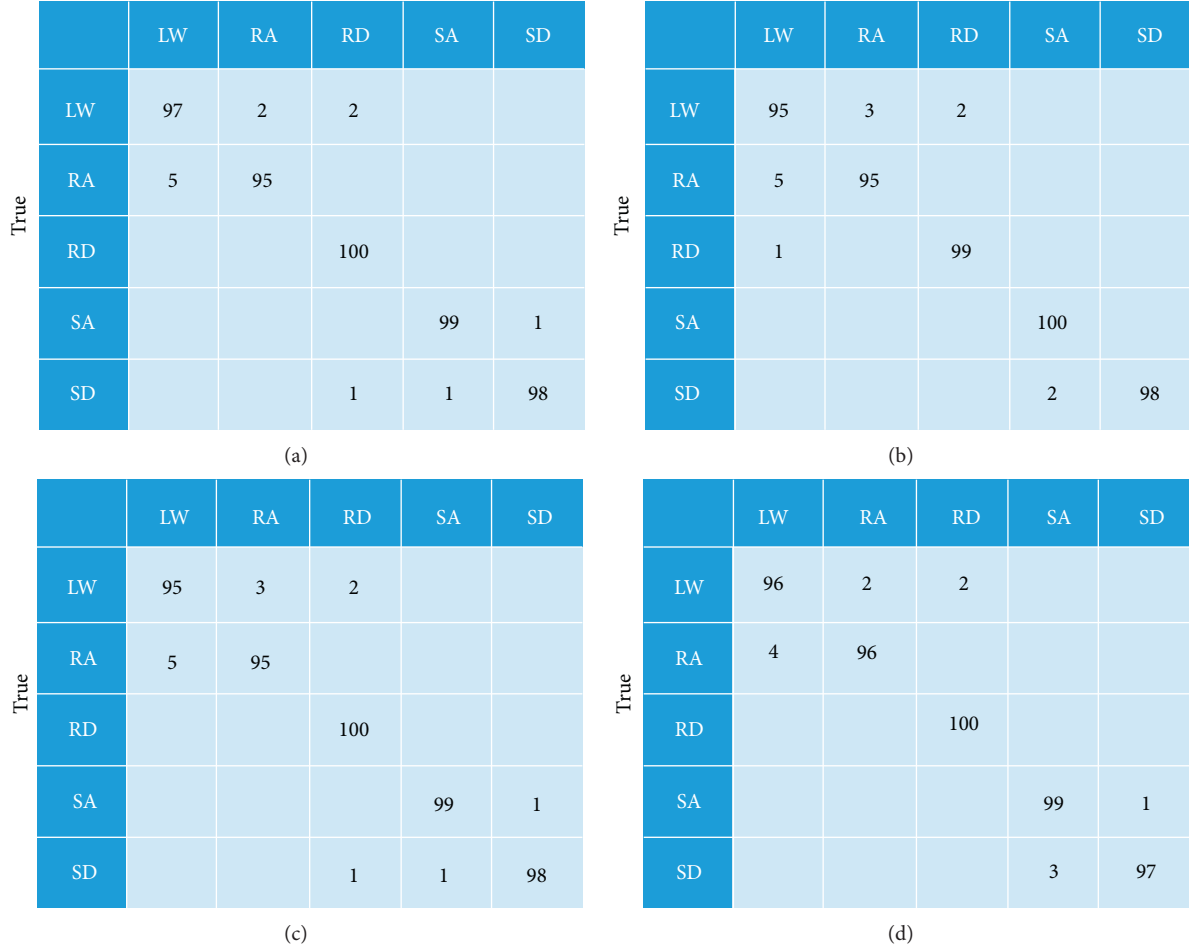


FIGURE 6: Confusion matrix for steady-state recognition from different classifiers. (a) LDA+SVM confusion matrix estimation. (b) LDA+QDA confusion matrix estimation. (c) LDA+LGBM confusion matrix estimation. (d) LDA+RF confusion matrix estimation.

TABLE 2: Average accuracy and consumption time of transient-state recognition (unit: % (ms)).

	SVM	QDA	LGBM	RF
Preswing	72.8 (42.3)	49.6 (1.0)	85.1 (14.3)	83.9 (39.7)
Prestance	75.4 (25.6)	49.4 (0.7)	83.5 (11.6)	82.5 (34.1)

By observing the inconsistency between the observed value and the predicted value, we can judge whether the observed value is wrong or not. Combined with the results of steady gait analysis, this method can effectively reduce the misclassification of Table 3. The final predictive accuracy is shown in Table 4, and it improves the accuracy of transient-state recognition from 84.2% to 91.2%.

In addition, because the transition probabilities between level walking to other locomotion modes are all equal to 20%, the prediction accuracy rate for level walking seems not as useful as the other modes. Similarly, the probability of other movements switching to level walking is 50%; if the movement is misclassified as level walking, the error cannot

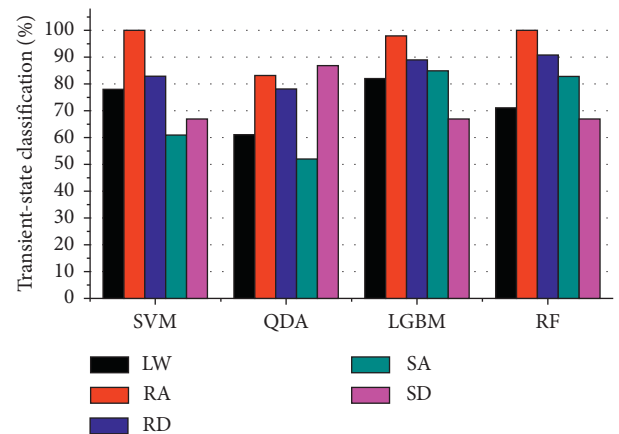


FIGURE 7: Transient-state recognition at preswing phase.

be corrected by using this method. The above two cases are inherent deficiencies of this method.

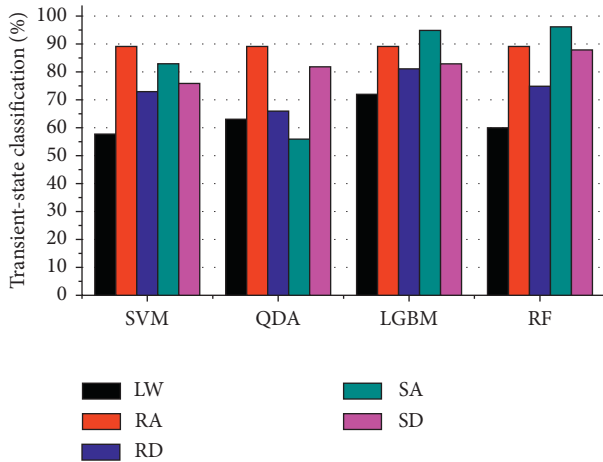


FIGURE 8: Transient-state recognition at prestance phase.

TABLE 3: The confusion matrix of LightGBM classifier in preswing transient state.

	LW	RA	RD	SA	SD	Accuracy (%)
LW	82	2	2	4	10	82
RA	0	98	2	0	0	98
RD	4	2	89	4	2	89
SA	0	2	0	85	13	85
SD	23	3	0	8	67	67
Total	—	—	—	—	—	84.2

TABLE 4: The confusion matrix of LightGBM classifier in transitional state.

	LW	RA	RD	SA	SD	Accuracy (%)
LW	82	2	2	4	10	82
RA	0	100	0	0	0	100
RD	4	0	96	0	0	96
SA	0	0	0	100	0	100
SD	23	0	0	0	78	78
Total	—	—	—	—	—	91.2

5. Conclusions

In this paper, a multilevel classifier fusion strategy based on steady gait recognition and transient-state recognition is proposed to realize human motion intention and prediction under complex environment. The steady-state recognition based on inertial and sEMG fusion as prior gait intention recognition, as well as the LDA + RF classifier, produced over 97.8% accuracy. The transient-state recognition only based on 50 ms sEMG signal as upcoming movement observation and LightGBM classifier outperformed other traditional methods in the accuracy and prediction time, 85.1% accuracy/14.3 ms in the prestance phase and 83.5% accuracy/11.6 ms in the preswing phase. A simplified HMM model that combined prior knowledge and observation was constructed to predict upcoming locomotion with over 91% accuracy. The prediction time could be guaranteed within 70 ms, which ensures that the lower limb prosthesis can switch locomotion in time. The method proposed in this

paper can be used in the control of lower limb prosthesis in the future to improve the movement coordination of amputees. In future work, we will consider more application scenarios combined with multisensor information fusion technology.

Data Availability

No data were used to support the study.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This study was supported by the Science and Technology Planning Project of Guangdong Province under grant no. 2016A020220003, Provincial Key Platforms and Major Scientific Research Projects of Guangdong Universities under grant no. 2017KTSCX208, and Science and Technology Planning Project of Zhongshan under grant no. 2019B2066.

References

- [1] Z. Li, Y. Yuan, L. Luo et al., "Hybrid brain/muscle signals powered wearable walking exoskeleton enhancing motor ability in climbing stairs activity," *IEEE Transactions on Medical Robotics and Bionics*, vol. 1, no. 4, pp. 218–227, 2019.
- [2] H. Su, S. Ertug Ovrur, Z. Li et al., "Internet of things (iot)-based collaborative control of a redundant manipulator for teleoperated minimally invasive surgeries," in *Proceedings of the 2020 International Conference on Robotics and Automation (ICRA)*, pp. 9737–9742, IEEE, Paris, France, May–August 2020.
- [3] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, "Robot learning system based on adaptive neural control and dynamic movement primitives," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, pp. 777–787, 2018.
- [4] Y. Liu, W. Su, Z. Li et al., "Motor-imagery-based teleoperation of a dual-arm robot performing manipulation tasks," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 11, pp. 414–424, 2018.
- [5] H. Lin, T. Zhang, Z. Chen, H. Song, and C. Yang, "Adaptive fuzzy Gaussian mixture models for shape approximation in robot grasping," *International Journal of Fuzzy Systems*, vol. 21, no. 4, pp. 1026–1037, 2019.
- [6] M. Windrich, M. Grimmer, O. Christ, S. Rinderknecht, and P. Beckerle, "Active lower limb prosthetics: a systematic review of design issues and solutions," *BioMedical Engineering OnLine*, vol. 15, p. 140, 2016.
- [7] Z. Li, B. Huang, A. Ajoudani, C. Yang, C.-Y. Su, and A. Bicchi, "Asymmetric bimanual control of dual-arm exoskeletons for human-cooperative manipulations," *IEEE Transactions on Robotics*, vol. 34, pp. 264–271, 2017.
- [8] H. Huang, T. A. Kuiken, R. D. Lipschutz et al., "A strategy for identifying locomotion modes using surface electromyography," *IEEE Transactions on Biomedical Engineering*, vol. 56, pp. 65–73, 2008.
- [9] Z. Li, J. Li, S. Zhao, Y. Yuan, Y. Kang, and C. P. Chen, "Adaptive neural control of a kinematically redundant exoskeleton robot using brain-machine interfaces," *IEEE*

- Transactions on Neural Networks and Learning Systems*, vol. 30, pp. 3558–3571, 2018.
- [10] K. Deok-Hwan, C. Chi-Young, and R. Jaehwan, “Real-time locomotion mode recognition employing correlation feature analysis using emg pattern,” *ETRI Journal*, vol. 36, pp. 99–105, 2014.
 - [11] D. Joshi, B. H. Nakamura, and M. E. Hahn, “High energy spectrogram with integrated prior knowledge for emg-based locomotion classification,” *Medical Engineering & Physics*, vol. 37, no. 5, pp. 518–524, 2015.
 - [12] F. Peng, W. Peng, C. Zhang, and D. Zhong, “Iot assisted kernel linear discriminant analysis based gait phase detection algorithm for walking with cognitive tasks,” *IEEE Access*, vol. 7, pp. 68240–68249, 2019.
 - [13] J. Li, J. Wang, S. Wang et al., “Neural approximation-based model predictive tracking control of non-holonomic wheel-legged robots,” *International Journal of Control, Automation and Systems*, pp. 1–10, 2020.
 - [14] H. Su, Y. Hu, H. R. Karimi, A. Knoll, G. Ferrigno, and E. De Momi, “Improved recurrent neural network-based manipulator control with remote center of motion constraints: experimental results,” *Neural Networks*, vol. 131, pp. 291–299, 2020.
 - [15] J. Li, J. Wang, S. Wang et al., “Parallel structure of six wheel-legged robot trajectory tracking control with heavy payload under uncertain physical interaction,” *Assembly Automation*, vol. 40, no. 5, pp. 675–687, 2020.
 - [16] Z. Liu, W. Lin, Y. Geng, and P. Yang, “Intent pattern recognition of lower-limb motion based on mechanical sensors,” *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 651–660, 2017.
 - [17] B.-Y. Su, J. Wang, S.-Q. Liu, M. Sheng, J. Jiang, and K. Xiang, “A cnn-based method for intent recognition using inertial measurement units and intelligent lower limb prosthesis,” *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 27, no. 5, pp. 1032–1042, 2019.
 - [18] E. C. Wentink, V. G. H. Schut, E. C. Prinsen, J. S. Rietman, and P. H. Veltink, “Detection of the onset of gait initiation using kinematic sensors and emg in transfemoral amputees,” *Gait & Posture*, vol. 39, no. 1, pp. 391–396, 2014.
 - [19] Q. Ai, Y. Zhang, W. Qi, Q. Liu, and a. K. Chen, “Research on lower limb motion recognition based on fusion of semg and accelerometer signals,” *Symmetry*, vol. 9, no. 8, p. 147, 2017.
 - [20] A. J. Young, T. A. Kuiken, and L. J. Hargrove, “Analysis of using EMG and mechanical sensors to enhance intent recognition in powered lower limb prostheses,” *Journal of Neural Engineering*, vol. 11, Article ID 056021, 2014.
 - [21] H. Su, S. E. Ovrur, X. Zhou, W. Qi, G. Ferrigno, and E. De Momi, “Depth vision guided hand gesture recognition using electromyographic signals,” *Advanced Robotics*, pp. 1–13, 2020.
 - [22] H. Huang, T. Zhang, C. Yang, and C. P. Chen, “Motor learning and generalization using broad learning adaptive neural control,” *IEEE Transactions on Industrial Electronics*, vol. 67, pp. 8608–8617, 2019.
 - [23] H. Huang, F. Zhang, L. J. Hargrove, Z. Dou, D. R. Rogers, and K. B. Englehart, “Continuous locomotion-mode identification for prosthetic legs based on neuromuscular–mechanical fusion,” *IEEE Transactions on Biomedical Engineering*, vol. 58, pp. 2867–2875, 2011.
 - [24] J. A. Spanias, A. M. Simon, S. B. Finucane, E. J. Perreault, and L. J. Hargrove, “Online adaptive neural control of a robotic lower limb prosthesis,” *Journal of Neural Engineering*, vol. 15, Article ID 016015, 2018.
 - [25] M. Meng, Z. Luo, Q. She, and Y. Ma, “Automatic recognition of gait mode from emg signals of lower limb,” in *Proceedings of the 2010 The 2nd International Conference on Industrial Mechatronics and Automation*, vol. 1, pp. 282–285, Beijing, China, May 2010.
 - [26] C. Yang, C. Chen, N. Wang, Z. Ju, J. Fu, and M. Wang, “Biologically inspired motion modeling and neural control for robot learning from demonstrations,” *IEEE Transactions on Cognitive and Developmental Systems*, vol. 11, pp. 281–291, 2018.
 - [27] J. Li, J. Wang, H. Peng, L. Zhang, Y. Hu, and H. Su, “Neural fuzzy approximation enhanced autonomous tracking control of the wheel-legged robot under uncertain physical interaction,” *Neurocomputing*, vol. 410, pp. 342–353, 2020.
 - [28] X. Wu, Z. Li, Z. Kan, and H. Gao, “Reference trajectory reshaping optimization and control of robotic exoskeletons for human-robot co-manipulation,” *IEEE Transactions on Cybernetics*, vol. 50, no. 8, pp. 3740–3751, 2020.
 - [29] A. Salman, E. G. Allstot, A. Y. Chen, A. M. Dixon, D. Gangopadhyay, and D. J. Allstot, “Compressive sampling of emg bio-signals,” in *Proceedings of the 2011 IEEE International Symposium of Circuits and Systems (ISCAS)*, pp. 2095–2098, IEEE, Rio de Janeiro, Brazil, May 2011.
 - [30] L. Manoni, C. Turchetti, L. Falaschetti, and P. Crippa, “A comparative study of computational methods for compressed sensing reconstruction of emg signal,” *Sensors*, vol. 19, no. 16, p. 3531, 2019.
 - [31] K. Huang, X. Guo, Y. Guo, and G. Wang, “Heterogeneous bayesian compressive sensing for sparse signal recovery,” *IET Signal Processing*, vol. 8, no. 9, pp. 1009–1017, 2014.
 - [32] H. Su, Y. Hu, Z. Li, A. Knoll, G. Ferrigno, and E. De Momi, “Reinforcement learning based manipulation skill transferring for robot-assisted minimally invasive surgery,” in *Proceedings of the 2020 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2203–2208, Paris, France, May–August 2020.
 - [33] Z. Li, C.-Y. Su, G. Li, and H. Su, “Fuzzy approximation-based adaptive backstepping control of an exoskeleton for human upper limbs,” *IEEE Transactions on Fuzzy Systems*, vol. 23, no. 3, pp. 555–566, 2015.
 - [34] K. Huang, S. Tan, Y. Luo, X. Guo, and G. Wang, “Enhanced radio tomographic imaging with heterogeneous bayesian compressive sensing,” *Pervasive and Mobile Computing*, vol. 40, pp. 450–463, 2017.
 - [35] M. E. Tipping, “Sparse bayesian learning and the relevance vector machine,” *Journal of Machine Learning Research*, vol. 1, pp. 211–244, 2001.
 - [36] S. Ji, Y. Xue, and L. Carin, “Bayesian compressive sensing,” *IEEE Transactions on Signal Processing*, vol. 56, no. 6, pp. 2346–2356, 2008.
 - [37] Z. Li, C. Xu, Q. Wei, C. Shi, and C.-Y. Su, “Human-inspired control of dual-arm exoskeleton robots with force and impedance adaptation,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 12, pp. 5296–5305, 2018.
 - [38] H. Su, C. Yang, G. Ferrigno, and E. De Momi, “Improved human-robot collaborative control of redundant robot for teleoperated minimally invasive surgery,” *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1447–1453, 2019.
 - [39] W. Qi, H. Su, and A. Aliverti, “A smartphone-based adaptive recognition and real-time monitoring system for human activities,” *IEEE Transactions on Human-Machine Systems*, vol. 50, no. 5, pp. 414–423, 2020.
 - [40] H. A. Varol, F. Sup, and M. Goldfarb, “Multiclass real-time intent recognition of a powered lower limb prosthesis,” *IEEE*

- Transactions on Biomedical Engineering*, vol. 57, no. 3, pp. 542–551, 2010.
- [41] L. J. Hargrove, G. Li, K. B. Englehart, and B. S. Hudgins, “Principal components analysis preprocessing for improved classification accuracies in pattern-recognition-based myoelectric control,” *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 5, pp. 1407–1414, 2009.
- [42] H. Su, W. Qi, C. Yang, J. Sandoval, G. Ferrigno, and E. D. Momi, “Deep neural network approach in robot tool dynamics identification for bilateral teleoperation,” *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2943–2949, 2020.
- [43] K. H. Ha, H. A. Varol, and M. Goldfarb, “Volitional control of a prosthetic knee using surface electromyography,” *IEEE Transactions on Biomedical Engineering*, vol. 58, no. 1, pp. 144–151, 2011.
- [44] C. Yang, Y. Jiang, J. Na, Z. Li, L. Cheng, and C.-Y. Su, “Finite-time convergence adaptive fuzzy control for dual-arm robot with unknown kinematics and dynamics,” *IEEE Transactions on Fuzzy Systems*, vol. 27, pp. 574–588, 2018.

Research Article

Adaptive Visually Servoed Tracking Control for Wheeled Mobile Robot with Uncertain Model Parameters in Complex Environment

Fujie Wang ¹, Yi Qin ¹, Fang Guo,¹ Bin Ren,¹ and John T. W. Yeow^{1,2}

¹School of Electrical Engineering and Intelligentization, Dongguan University of Technology, Dongguan 523000, China

²System Design Engineering, University of Waterloo, Waterloo, ON, Canada

Correspondence should be addressed to Yi Qin; qinyidee@163.com

Received 7 September 2020; Revised 4 November 2020; Accepted 17 November 2020; Published 4 December 2020

Academic Editor: Ning Wang

Copyright © 2020 Fujie Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper investigates the stabilization and trajectory tracking problem of wheeled mobile robot with a ceiling-mounted camera in complex environment. First, an adaptive visual servoing controller is proposed based on the uncalibrated kinematic model due to the complex operation environment. Then, an adaptive controller is derived to provide a solution of uncertain dynamic control for a wheeled mobile robot subject to parametric uncertainties. Furthermore, the proposed controllers can be applied to a more general situation where the parallelism requirement between the image plane and operation plane is no more needed. The overparameterization of regressor matrices is avoided by exploring the structure of the camera-robot system, and thus, the computational complexity of the controller can be simplified. The Lyapunov method is employed to testify the stability of a closed-loop system. Finally, simulation results are presented to demonstrate the performance of the suggested control.

1. Introduction

In recent decades, the wheeled mobile robots (WMRs) have received increasing attention due to their promising applications in transportation, health care, security, and so on, which promotes the research of high-accuracy tracking control and stability analysis of the WMRs [1–4]. Particularly, WMR belongs to the nonholonomic mechanical system which is unable to be stabilized at one equilibrium by means of continuous and static state feedback controller [5–7], leading to the great complexity of the study about WMRs. A significant direction of the motion control of WMR is to employ various kinds of sensors in a closed-loop controller. The visual sensor, one of the typical noncontact sensors, has particular advantages such as abundant visual information and high efficiency; hence, visual servoing control of WMR has become a vigorous research field worldwide.

Numerous scientific achievements have been reported on visual servoing and vision-based manipulations [8, 9]. Just like the robot manipulators, the vision system in a mobile robot can be formed by two kinds of configurations,

namely, eye-in-hand configuration [10, 11] and fixed-camera configuration [12, 13], respectively. For the first category configuration, the camera is mounted on the end-effector. In contrast, the camera is called a static-camera or fixed-camera configuration when the camera is located on the ceiling. Till now, there has been a plethora of prominent literature concerning the visual servoing of nonholonomic mobile robots. To mention a few, in [14], position-based visual servoing (PBVS) was employed for visual tracking between a WMR and a multi-DOF crane. In [15], a visual servoing scheme was presented for a nonholonomic mobile robot to combine the merits of PBVS and image-based visual servoing (IBVS). In [16], a novel strategy was proposed for visual servoing of a mobile robot and the difficult issue of the automatic extrinsic calibration was addressed. It should be noted that the above-mentioned works require the camera mounted on the end-effector to be tediously calibrated beforehand. Unfortunately, the controllers are very sensitive to camera calibration errors which may give rise to reduced accuracy. To obviate this limitation, the uncalibrated camera system has emerged as a valid tool for practical systems. In [17], two independent uncalibrated cameras were used to

accomplish person tracking for a vision-based mobile robot subject to nonholonomic constraint. The authors in [18] addressed a visual servo regulation approach which can work well without the perfectly calibrated camera. To deal with the imperfect calibration of the camera, the visual servoing of nonholonomic mobile robots was proposed in [19], considering both unknown extrinsic parameters and unknown depth from the camera to the motion plane. In [20], without calibrating the camera, the eye-in-hand visual trajectory tracking control strategy was constructed to ensure that the WMR is able to track the desired trajectory.

The aforesaid papers mainly discuss the visual servoing of nonholonomic mobile robots with eye-in-hand configuration. The fixed-camera configuration has the global sight and it enables the camera system to keep the observed object always in the field of view. Therefore, many researchers also devote themselves to the solutions of a WMR with the fixed uncalibrated camera. For instance, in [21], the unified tracking and regulation WMR visual servoing control was studied and the state information can be utilized to formulate the WMR kinematic model. In [22], a monocular camera with a fixed position and orientation was used to track the desired trajectory for a WMR and the controller does not require the camera to be mounted. Taking the limited velocity of a WMR into account, the control scheme for tracking a moving target by a WMR was presented in [23]. Despite the significant progress of visual servoing with the fixed uncalibrated camera, the adaptability of these controllers is unsatisfactory since the camera plane is always required to be parallel to the motion plane of the robots. It means that the controllers in [21–23] are no longer effective when the camera is fixed at a general orientation on the ceiling. To overcome this drawback, the authors in [24, 25] proposed the visual servoing of a mobile robot without the parallelism requirement. By employing an adaptive image-based visual servoing approach, the camera image plane and the motion plane of WMRs are free from position constraint. However, all these methods suffer from the overparameterization in the process of the decoupled linear transformation. In addition, the previous controllers are developed via a kinematics-based model and the nonlinear dynamics are not taken into consideration in controller design.

Dynamic model-based control methods [26–29] reflect the motion of real mobile robots with significant dynamics characterized by mass and inertia as well as friction, which are otherwise not considered in kinematics-based model control. The nonlinear dynamics of the mobile robot usually contain uncertain and time-varying parameters. Consequently, the nonlinear dynamic controllers to deal with unmodeled robot dynamics diverse further research. Control methodologies such as adaptive control technique [6], sliding mode control technique [27], and neural network control technique [28] have been developed on dynamic model with uncertain parameters of mobile robots. By far, visual servoing control for mobile robots at the dynamic level can be found in [8, 30–32]. In [32], position/orientation tracking control of WMRs via an uncalibrated camera was considered and the adaptive controller was designed to

compensate for the dynamic and the camera system uncertainties. It is noteworthy that the preceding studies are confined to visual servoing of mobile robots based on dynamic model, and these methods are invalid in a more general situation where the uncalibrated camera is fixed at an arbitrary position. Additionally, overparameterization limits the applicability of these controllers to a great extent.

In this paper, the stabilization and trajectory tracking problems of a wheeled mobile robot in complex environments are studied. The main contributions of this paper are threefold:

- (1) Two visual servoing controllers are proposed to stabilize a wheeled mobile robot with a ceiling-mounted camera and the desired trajectory tracking can be realized. First, an adaptive visual servoing controller is proposed based on the kinematic model. Then, an adaptive controller is derived to provide a solution of uncertain dynamic wheeled mobile robot subject to parametric uncertainties related to the camera system.
- (2) An uncalibrated visual servoing control strategy is proposed to realize trajectory tracking of a WMR, whose major superiority lies in the avoidance of both the requirement that the camera plane must be parallel to the motion plane of the robots and the overparameterization as in [24, 25]. Such a solution allows the controllers to be applied in a more general situation with a simpler structure and higher efficiency.
- (3) In comparison with the existing works for visual servoing mobile robot control in [19, 33], the camera parameters, including the intrinsic and extrinsic parameters, are unnecessary to be well calibrated, and the tracking control can be ensured in the presence of uncertain dynamics.

2. Preliminaries and System Descriptions

Throughout this paper, a typical setup for the visually servoed wheeled mobile robot is considered, as shown in Figure 1, where the camera is mounted on the ceiling to observe the movement of feature point labeled on the mobile robot. Let $O_b X_b Y_b Z_b$ be the base coordinate frame, $O_c X_c Y_c Z_c$ be the camera coordinate frame, and $O_m X_m Y_m Z_m$ be the mobile robot coordinate frame, respectively. Furthermore, let O_m be the center of mass of wheeled mobile robot, P be the feature point, and d be the distance from O_m to P along the positive direction of axis X_m . Without loss of generality, it is assumed that the robot moves in a specific plane. Note that both the image-based kinematic and dynamic control are fully considered in this paper.

2.1. Kinematics Model of Nonholonomic Mobile Robot in Task Space. Let us firstly review the kinematics model of a mobile robot. Denote the task-space position of wheeled mobile robot with respect to the base coordinate frame by

$[x_B, 0]^T = [x_b, y_b, 0]^T$ and the orientation by θ , whose forward rotation direction is set to counterclockwise from axis X_b . Then, the kinematic model of the mobile robot can be written as [32, 34]

$$\dot{x}_B = \begin{bmatrix} \dot{x}_b \\ \dot{y}_b \end{bmatrix} = \begin{bmatrix} \nu \cos \theta \\ \nu \sin \theta \end{bmatrix} \quad (1)$$

$$\dot{\theta} = \omega, \quad (2)$$

where ν and ω denote the linear velocity and angular velocity of wheeled mobile robot in task space, respectively. From [32], the nonholonomic constraint of wheeled mobile robot can be formulated as follows:

$$\dot{x}_b \sin \theta - \dot{y}_b \cos \theta = 0. \quad (3)$$

This nonholonomic constraint indicates that the velocity along the connected direction between the left and right driving wheels is restricted to be zero; that is, the wheeled mobile robot will not slip during task execution. Combining the definition of θ and mobile robot kinematics, the task-space position of P with respect to the base coordinate frame can be described as [24]

$$\begin{bmatrix} x \\ 0 \end{bmatrix} = \begin{bmatrix} x_p \\ y_p \\ 0 \end{bmatrix} = \begin{bmatrix} x_b + d \cos \theta \\ y_b + d \sin \theta \\ 0 \end{bmatrix}. \quad (4)$$

Differentiating (4) with respect to time gives rise to

$$\dot{x} = \begin{bmatrix} \cos \theta & -d \sin \theta \\ \sin \theta & d \cos \theta \end{bmatrix} \begin{bmatrix} \nu \\ \omega \end{bmatrix}. \quad (5)$$

2.2. Transformation from Task Space to Image Space. Let $y \in \mathbb{R}^2$ be the position of feature point P on the image plane. Via the perspective projection model [8, 35], the mapping relation of P from task space to image space is given by

$$\begin{bmatrix} Y \\ 1 \end{bmatrix} = \frac{1}{z} D \begin{bmatrix} x \\ 0 \\ 1 \end{bmatrix}, \quad (6)$$

where z is the depth information of feature point, $D = \Omega T \in \mathbb{R}^{3 \times 4}$ is the so-called *perspective projection matrix* (see [8]), $T \in \mathbb{R}^{4 \times 4}$ denotes the homogenous transformation matrix from the base frame to camera frame, and $\Omega \in \mathbb{R}^{3 \times 4}$ denotes the internal transformation matrix of camera. It should be noted that Ω and T depend on the intrinsic and extrinsic parameters, respectively. In addition, the depth information is defined as

$$z = D_3^T \begin{bmatrix} x \\ 0 \\ 1 \end{bmatrix}, \quad (7)$$

where D_3^T denotes the 3rd row of matrix D . Differentiating (6) and utilizing the definition of depth, we can obtain

$$\begin{aligned} \dot{y} &= \frac{1}{z} \left(\overline{D}_{2 \times 2} - y \overline{D}_3^T \right) \dot{x} \\ &= \frac{1}{z} \underbrace{\left(\overline{D}_{2 \times 2} - y \overline{D}_3^T \right)}_{N(y, \theta)} \begin{bmatrix} \cos \theta & -d \sin \theta \\ \sin \theta & d \cos \theta \end{bmatrix} \underbrace{\begin{bmatrix} \nu \\ \omega \end{bmatrix}}_{\tau_K}, \end{aligned} \quad (8)$$

where τ_K can be interpreted as the kinematic control input and $\overline{D} = [\overline{D}_{2 \times 2}^T, \overline{D}_3^T]^T \in \mathbb{R}^{3 \times 2}$ is the left 3×2 submatrix of D . Note that $N(y, \theta) \in \mathbb{R}^{2 \times 1}$ depends on both the intrinsic and extrinsic parameters of the visual model. In addition, $N(y, \theta)$ is called *the depth-independent interaction matrix* since the depth information z is separated. By exploiting the structure of $N(y, \theta)$, we can further obtain

$$\begin{aligned} N(y, \theta) &= \left(\overline{D}_{2 \times 2} - y \overline{D}_3^T \right) \begin{bmatrix} \cos \theta & -d \sin \theta \\ \sin \theta & d \cos \theta \end{bmatrix} \\ &= \underbrace{\overline{D}_{2 \times 2}}_{N_a(\theta)} \begin{bmatrix} \cos \theta & -d \sin \theta \\ \sin \theta & d \cos \theta \end{bmatrix} - y \underbrace{\overline{D}_3^T}_{N_b(\theta)} \begin{bmatrix} \cos \theta & -d \sin \theta \\ \sin \theta & d \cos \theta \end{bmatrix}. \end{aligned} \quad (9)$$

Similarly, the time differential of depth information z can be written as

$$\dot{z} = \overline{D}_3^T \dot{x} = \overline{D}_3^T \underbrace{\begin{bmatrix} \cos \theta & -d \sin \theta \\ \sin \theta & d \cos \theta \end{bmatrix}}_{N_b(\theta)} \tau_K. \quad (10)$$

The linearization properties, which are important to simplify the control design, are given as follows [34, 35].

Property 1. The products of $N_a(\theta)\zeta$ and $\Xi N_b(\theta)\zeta$ can be linearly decomposed and recombined as

$$N_a(\theta)\zeta = E_{K,a}(\theta, \zeta)\phi_{k,a}, \quad (11)$$

$$\Xi N_b(\theta)\zeta = E_{K,b}(\theta, \zeta, \xi)\phi_{k,b}, \quad (12)$$

where $\zeta \in \mathbb{R}^{2 \times 1}$ is a constant vector, $\Xi = \text{diag}(\xi) \in \mathbb{R}^{2 \times 2}$ is a diagonal matrix with $\xi = [\xi_1, \xi_2]^T$, $\phi_{k,a} \in \mathbb{R}^{e_1 \times 1}$ and $\phi_{k,b} \in \mathbb{R}^{e_2 \times 1}$ are visual model parameter vector, and $E_{K,a}(\theta, \zeta) \in \mathbb{R}^{2 \times e_1}$ and $E_{K,b}(\theta, \zeta, \xi) \in \mathbb{R}^{2 \times e_2}$ are the regressor matrices without depending on the parameter vectors $\phi_{k,a}$ and $\phi_{k,b}$. Specifically, by observing (7), (9), and (10), it can be further obtained as follows:

$$\dot{\zeta} \xi = \Xi N_b(\theta)\tau_K = E_{K,b}(\theta, \tau_K, \xi)\phi_{k,b}, \quad (13)$$

$$z \xi = \overline{E}_{K,b}(\theta, x, \xi)\phi_{k,b}, \quad (14)$$

where $\overline{E}_{K,b}(\theta, x, \xi) \in \mathbb{R}^{2 \times e_2}$ is the depth regressor matrix. Note that the vector $\phi_{k,b}$ should involve all the depth parameters. By employing (9), (11), and (12), we have

$$N(y, \theta)\zeta = E_{K,a}(\theta, \zeta)\phi_{k,a} + E_{K,b}(\theta, \zeta, -y)\phi_{k,b}. \quad (15)$$

Remark 1. In this paper, parameter uncertainties of visual servoing robot system are addressed, which means that the real parameter values $\phi_{k,a}$ and $\phi_{k,b}$ in (11)–(15) are unknown in the control design. Moreover, the image depth is not required to be consistent during robot operation as in [25, 36]; that is, the fixed-camera image plane can be not parallel to the operation plane, where a more realistic scenario is considered in both kinematic and dynamic control. In addition, the distance d between the feature point and the origin of the coordinate system $O_m X_m Y_m Z_m$ is assumed to be uncalibrated, which, together with the above parameter uncertainties, imposes great complexity and challenge in visual tracking control.

Throughout this paper, the following assumptions hold.

Assumption 1. The feature point P can always be detected throughout the entire robot workspace such that the image position is continuously available. Moreover, the orientation θ of mobile robot can be measured by the encoders or other optical sensors mounted on the actuators.

2.3. Dynamics Model of Nonholonomic Mobile Robot. The dynamic behavior of wheeled mobile robot can be expressed by the Euler-Lagrangian equation as follows [6, 37]:

$$M(\theta)\ddot{q} + V(\theta, \dot{\theta})\dot{q} + G = B(\theta)\tau_D + A^T(\theta)\lambda, \quad (16)$$

where $q = [x_b, y_b, \theta]^T \in \mathbb{R}^{3 \times 1}$, $M(\theta) \in \mathbb{R}^{3 \times 3}$ is the symmetric and positive-definite inertia matrix, $V(\theta, \dot{\theta}) \in \mathbb{R}^{3 \times 3}$ is the Coriolis and centrifugal matrix, $G \in \mathbb{R}^{3 \times 1}$ denotes the gravitational force, $B(\theta) \in \mathbb{R}^{3 \times 2}$ denotes the input transformation matrix, $\tau_D \in \mathbb{R}^{2 \times 1}$ represents the dynamic input torque, $A(\theta) \in \mathbb{R}^{1 \times 3}$ is the so-called constraint vector with λ being the constraint force, and the constraint form can be further represented as

$$A(\theta)\dot{q} = 0. \quad (17)$$

It must be noted that matrices $M(\theta)$, $V(\theta, \dot{\theta})$, G , $B(\theta)$, and $A(\theta)$ do not depend on the actual position of x_b and y_b (more details of the robotic dynamics model can be referred to [37]). Based on the kinematics (1) and (2), the following holds:

$$\dot{q} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \tau_K. \quad (18)$$

$S(\theta)$

Differentiating both sides of (18) and then substituting into the robot dynamics (16) and premultiplying both sides by $S^T(\theta)$, we have

$$\overline{M}(\theta)\dot{\tau}_K + \overline{V}(\theta, \dot{\theta})\tau_K + \overline{G}(\theta) = \overline{B}(\theta)\tau_D, \quad (19)$$

where (17) is utilized in the process of formula simplification and $\overline{M}(\theta) = S^T(\theta)M(\theta)S(\theta)$, $\overline{V}(\theta, \dot{\theta}) = S^T(\theta)(M(\theta)\dot{S}(\theta) + V(\theta, \dot{\theta})S(\theta))$, $\overline{G}(\theta) = S^T(\theta)G(\theta)$, and $\overline{B}(\theta) = S^T(\theta)B(\theta)$, respectively. To facilitate the control scheme, the dynamics properties of WMR are employed [37].

Property 2. The inertia matrix $\overline{M}(\theta)$ is symmetric and positive definite, which also satisfies

$$\mu_1 \|\theta\|^2 \leq \theta^T \overline{M}(\theta) \theta \leq \mu_2 \|\theta\|^2, \quad (20)$$

where μ_1 and μ_2 are positive constants and $\|\cdot\|$ denotes the standard Euclidean norm.

Property 3. The matrix $\dot{\overline{M}}(\theta) - 2\overline{V}(\theta, \dot{\theta})$ is skew-symmetric such that

$$\aleph^T [\dot{\overline{M}}(\theta) - 2\overline{V}(\theta, \dot{\theta})] \aleph = 0, \quad (21)$$

with $\aleph \in \mathbb{R}^{2 \times 1}$ being a constant vector.

Property 4. The dynamic equation (19) can be linearly restructured as

$$\overline{M}(\theta)\dot{\rho} + \overline{V}(\theta, \dot{\theta})\rho + \overline{G}(\theta) = E_D(\theta, \dot{\theta}, \rho, \dot{\rho})\phi_d, \quad (22)$$

where $\rho \in \mathbb{R}^{2 \times 1}$ is a differentiable vector, $\phi_d \in \mathbb{R}^{\epsilon_3 \times 1}$ denotes the constant parameter vector of dynamics and is unknown in the control design, and $E_D(\theta, \dot{\theta}, \rho, \dot{\rho}) \in \mathbb{R}^{2 \times \epsilon_3}$ is the regressor matrix of dynamics.

Remark 2. Via observing (8) and (19), it can be found that the kinematic control and the dynamic control are related by τ_K and τ_D , respectively. If the designed kinematic input τ_K is actually achievable in the task execution without any time delay, the visual tracking control can be conveniently realized by the kinematic loop. However, in most state-of-the-art researches on wheeled mobile robot control [19, 32, 33, 38], it is stressed that the motors assembled on the left and right wheels may not respond fast enough with the result that the actual kinematic control values τ_K may lag behind the design values. Thus, in this paper, the dynamics control for visual servoing WMR is also addressed, simultaneously taking the mechanical parameter uncertainties into consideration; that is, the precise parameter values (e.g., robot mass, inertia, and friction) are not required to be exactly measured.

2.4. Problem Statement. Based on the above system model and assumptions, the control problems from two different perspectives, namely, the kinematic and dynamic control, are addressed. Given a continuous desired trajectory $y_d, \dot{y}_d, \ddot{y}_d \in \mathbb{R}^{2 \times 1}$ on the image plane, this paper aims to solve the following problems:

P1L: assuming that the WMR responds fast enough, design an adaptive visual servoing kinematic controller (AVSKC) τ_K such that the precise trajectory tracking performance can be obtained in the absence of calibrated camera model; that is,

$$\lim_{t \rightarrow \infty} y - y_d \rightarrow 0. \quad (23)$$

P2: when the kinematic input τ_K is not always achievable, design an adaptive visual servoing dynamic controller (AVSDC) τ_D such that (23) holds,

simultaneously taking into account the uncalibrated camera-robot model.

3. Adaptive Visual Servoing Kinematic Control for Wheeled Mobile Robot under Uncalibrated Visual Model

In this part, we focus on the adaptive visual servoing kinematic control scheme for wheeled mobile robot with uncalibrated camera model, where the projection plane of camera does not need to be parallel to the operation plane during the execution of the mission, and the dynamic control will be exhibited in the next section. Since the parameters of the visual model are unknown, adaption laws are presented to estimate the real parameter values, and based on the estimated parameters, AVSKC is developed to realize the asymptotic image trajectory tracking.

3.1. Controller Design. Let $\hat{N}(y, \theta)$, \hat{z} , and $\hat{\dot{z}}$ be the estimated values of $N(y, \theta)$, z , and \dot{z} by replacing the unknown parameters $\phi_{k,a}$ and $\phi_{k,b}$ in $N(y, \theta)$, z , and \dot{z} with the estimations $\hat{\phi}_{k,a}$ and $\hat{\phi}_{k,b}$, respectively, and the estimations are offered by the adaption laws. Define $\Delta y = y - y_d$ as the image error. Then, inspired by [25], the AVSKC is designed as

$$\tau_K = \hat{N}^{-1}(y, \theta) \left[\hat{z} \dot{y}_d - \left(\alpha + \frac{1}{2} \hat{\dot{z}} \right) \Delta y \right], \quad (24)$$

where α is a positive constant. In (24), the estimated visual model rather than the calibrated model is utilized, and the estimated depth and its differential are also introduced to compensate the model error since the image plane and the operation plane are nonparallel. Now, we can further analyze the closed-loop kinematics with depth information as follows:

$$\begin{aligned} z \Delta \dot{y} + \frac{1}{2} \dot{z} \Delta y &= N(y, \theta) \tau_K - z \dot{y}_d + \frac{1}{2} \dot{z} \Delta y \\ &= [N(y, \theta) \tau_K - \hat{N}(y, \theta) \tau_K] + \hat{N}(y, \theta) \tau_K \\ &\quad + [z \dot{y}_d - \hat{z} \dot{y}_d] + \hat{z} \dot{y}_d + \frac{1}{2} [z \Delta y - \hat{z} \Delta y] + \frac{1}{2} \hat{\dot{z}} \Delta y \\ &= E_{K,a}(\theta, \tau_K) \Delta \phi_{k,a} + E_{K,b}(\theta, \tau_K, -y) \Delta \phi_{k,b} \\ &\quad + \left(\bar{E}_{K,b}(\theta, x, \tau_K) \Delta \phi_{k,b} + E_{K,b} \left(\theta, \tau_K, \frac{1}{2} \Delta y \right) \Delta \phi_{k,b} \right) \\ &\quad + \hat{N}(y, \theta) \tau_K + \hat{z} \dot{y}_d + \frac{1}{2} \hat{\dot{z}} \Delta y, \end{aligned} \quad (25)$$

where $\Delta \phi_{k,a} = \phi_{k,a} - \hat{\phi}_{k,a}$ and $\Delta \phi_{k,b} = \phi_{k,b} - \hat{\phi}_{k,b}$, and Property 1 is used. Substituting the AVSKC (24) into (25) gives rise to

$$\begin{aligned} z \Delta \dot{y} + \frac{1}{2} \dot{z} \Delta y &= E_{K,a}(\theta, \tau_K) \Delta \phi_{k,a} - \alpha \Delta y \\ &\quad + \left[E_{K,b} \left(\theta, \tau_K, -\frac{y + y_d}{2} \right) + \bar{E}_{K,b}(\theta, x, \tau_K) \right] \Delta \phi_{k,b}. \end{aligned} \quad (26)$$

3.2. Unknown Parameter Estimation. By observing (24), it is obvious that the estimation of $N(y, \theta)$ is employed, which requires that the parameters $\hat{\phi}_a$ and $\hat{\phi}_b$ are updated online. The kinematic parameter updating laws are presented as

$$\dot{\hat{\phi}}_a = -\Phi_a^{-1} E_{K,a}^T(\theta, \tau_K) \Delta y, \quad (27)$$

$$\dot{\hat{\phi}}_b = -\Phi_b^{-1} \left[E_{K,b}^T \left(\theta, \tau_K, -\frac{y + y_d}{2} \right) + \bar{E}_{K,b}^T(\theta, x, \tau_K) \right] \Delta y, \quad (28)$$

where $\Phi_a \in \mathbb{R}^{e_1 \times e_1}$ and $\Phi_b \in \mathbb{R}^{e_2 \times e_2}$ are the positive-definite diagonal matrices. Thus, by integrating (27) and (28), $\hat{N}(y, \theta)$, \hat{z} , and $\hat{\dot{z}}$ in (24) are then available.

3.3. Stability Analysis. At this point, we are going to formulate the first theorem.

Theorem 1. *Consider the visual servoing wheeled mobile robot represented by (1), (2), (4), (6), and (8) satisfying the assumption that the estimated interaction matrix $\hat{N}(y, \theta)$ is nonsingular. In the case that the design kinematic input τ_K is actually achievable in the task execution, the adaptive visual servoing kinematic controller (AVSKC) given by (24) together with the visual parameter adaption laws (27) and (28) ensures the global stability of (26) and the asymptotical convergence of Δy to zero such that $\lim_{t \rightarrow \infty} y - y_d \rightarrow 0$.*

Proof. Construct the kinematic-based Lyapunov function candidate as

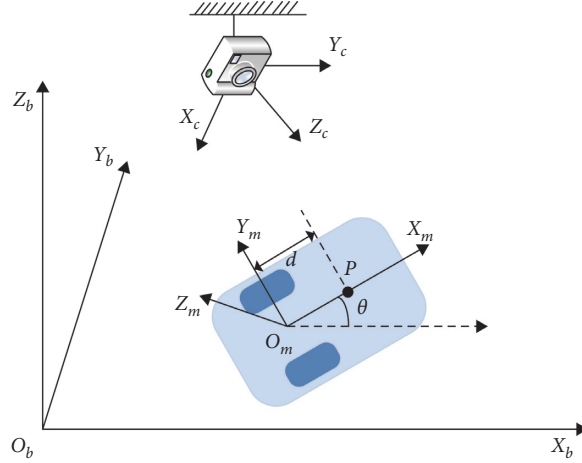


FIGURE 1: Visually servoed wheeled mobile robot system and coordinate representation.

$$V_k = \frac{1}{2} z \Delta y^T \Delta y + \frac{1}{2} \Delta \phi_{k,a}^T \Phi_a \Delta \phi_{k,a} + \frac{1}{2} \Delta \phi_{k,b}^T \Phi_b \Delta \phi_{k,b}. \quad (29)$$

Differentiating V_k with respect to time yields

$$\dot{V}_k = \Delta y^T \left[z \Delta \dot{y} + \frac{1}{2} \dot{z} \Delta y \right] + \Delta \phi_{k,a}^T \Phi_a \dot{\hat{\phi}}_{k,a} + \Delta \phi_{k,b}^T \Phi_b \dot{\hat{\phi}}_{k,b}. \quad (30)$$

Substituting the closed-loop kinematics (26) and the parameter updating laws (27) and (28) into (30), the derivative of V_k can be denoted as

$$\begin{aligned} \dot{V}_k &= \Delta y^T E_{K,a}(\theta, \tau_K) \Delta \phi_{k,a} - \Delta \phi_{k,a}^T E_{K,a}^T(\theta, \tau_K) \Delta y - \Delta y^T \alpha \Delta y \\ &\quad + \Delta y^T \left[E_{K,b} \left(\theta, \tau_K, -\frac{y+y_d}{2} \right) + \bar{E}_{K,b}(\theta, x, \tau_K) \right] \Delta \phi_{k,b} \\ &\quad - \Delta \phi_{k,b}^T \left[E_{K,b}^T \left(\theta, \tau_K, -\frac{y+y_d}{2} \right) + \bar{E}_{K,b}^T(\theta, x, \tau_K) \right] \Delta y \\ &= -\Delta y^T \alpha \Delta y \leq 0. \end{aligned} \quad (31)$$

Since $V_K \geq 0$ and $\dot{V}_K(t) \leq 0$, we can obtain that $V_K(t)$ is bounded; that is, Δy , $\Delta \phi_{k,a}$, and $\Delta \phi_{k,b}$ are bounded, which directly implies that $\hat{\phi}_{k,a}$ and $\hat{\phi}_{k,b}$ are both bounded since $\phi_{k,a}$ and $\phi_{k,b}$ are constants. Thus, $\hat{N}(y, \theta)$, \hat{z} , and $\hat{\tau}_K$ are all bounded, giving rise to the boundness of τ_K from (24), which means that $\dot{y}, \Delta \dot{y} \in \mathcal{L}_\infty$ from (8) and the boundness of (26) is guaranteed. From the result of (29) and (38), we have $\Delta y \in \mathcal{L}_2 \cap \mathcal{L}_\infty$. Therefore, we can obtain that $\lim_{t \rightarrow \infty} y - y_d \rightarrow 0$. Thus, the proof is completed.

From the result in [34], it has been proven that the matrix $N(y, \theta)$ is always nonsingular. Thus, if the parameters of $\hat{N}(y, \theta)$ in $\hat{\phi}_{k,a}$ and $\hat{\phi}_{k,b}$ are updated properly, it can be ensured that $\hat{N}(y, \theta)$ is full rank by modifying the parameter adaption laws. In this paper, the so-called parameter projection [39] is introduced to avoid nonsingularity of $\hat{N}(y, \theta)$. The adaption laws for visual kinematic parameters are presented as

$$\mathcal{U}_a = -\Phi_a^{-1} E_{K,a}^T(\theta, \tau_K) \Delta y, \quad (32)$$

$$\hat{\phi}_{a,i} = \text{proj}\{\mathcal{U}_{a,i}\}, \quad (33)$$

$$\mathcal{U}_b = -\Phi_b^{-1} \left[E_{K,b}^T \left(\theta, \tau_K, -\frac{y+y_d}{2} \right) + \bar{E}_{K,b}^T(\theta, x, \tau_K) \right] \Delta y, \quad (34)$$

$$\hat{\phi}_{b,i} = \text{proj}\{\mathcal{U}_{b,i}\}, \quad (35)$$

where $\hat{\phi}_{a,i}$, $\hat{\phi}_{b,i}$, $\mathcal{U}_{a,i}$, and $\mathcal{U}_{b,i}$ are the i th element of $\hat{\phi}_a$, $\hat{\phi}_b$, \mathcal{U}_a , and \mathcal{U}_b . Furthermore, the projection function is given as [39, 40]

$$\text{proj}\{\mathcal{U}_{a,i}\} = \begin{cases} \mathcal{U}_{a,i}, & \text{if } \hat{\phi}_{a,i} > \underline{\phi}_{a,i} \\ \mathcal{U}_{a,i}, & \text{if } \hat{\phi}_{a,i} = \underline{\phi}_{a,i}, \\ & \mathcal{U}_{a,i} \geq 0, \\ \mathcal{U}_{a,i}, & \text{if } \hat{\phi}_{a,i} = \bar{\phi}_{a,i}, \\ \mathcal{U}_{a,i}, & \mathcal{U}_{a,i} \leq 0, \\ \mathcal{U}_{a,i}, & \text{if } \hat{\phi}_{a,i} < \bar{\phi}_{a,i}, \\ 0, & \text{otherwise,} \end{cases} \quad (36)$$

$$\text{proj}\{\mathcal{U}_{b,i}\} = \begin{cases} \mathcal{U}_{b,i}, & \text{if } \hat{\phi}_{b,i} > \underline{\phi}_{b,i}, \\ \mathcal{U}_{b,i}, & \text{if } \hat{\phi}_{b,i} = \underline{\phi}_{b,i}, \\ & \mathcal{U}_{b,i} \geq 0, \\ \mathcal{U}_{b,i}, & \text{if } \hat{\phi}_{b,i} = \bar{\phi}_{b,i}, \\ \mathcal{U}_{b,i}, & \mathcal{U}_{b,i} \leq 0, \\ \mathcal{U}_{b,i}, & \text{if } \hat{\phi}_{b,i} < \bar{\phi}_{b,i}, \\ 0, & \text{otherwise,} \end{cases} \quad (37)$$

where $(\bar{\cdot})$ and $(\underline{\cdot})$ denote the upper and lower bounds of parameter (\cdot) . In this way, if the condition $(\cdot) \leq (\bar{\cdot}(0)) \leq (\bar{\cdot})$ is satisfied, then the parameter $(\bar{\cdot})$ will locate in the region

$[(\cdot), (\bar{\cdot})]$. Then, we are ready to state the following proposition.

Proposition 1. *Consider the visual servoing wheeled mobile robot represented by (1), (2), (4), (6), and (8). In the case that τ_K is achievable, the adaptive visual servoing kinematic controller (AVSKC) given by (24) together with the visual parameter adaption laws (33) and (35) ensures the global stability of (26) and the asymptotical convergence of image errors such that $\lim_{t \rightarrow \infty} y - y_d \rightarrow 0$.*

Proof. Choose the same Lyapunov function candidate V_k (29), whose time derivative can be written as

$$\begin{aligned} \dot{V}_k &= -\Delta y^T \alpha \Delta y + \Delta \phi_{k,a}^T \Phi_a \left(\hat{\phi}_{k,a} - \mathcal{U}_a \right) + \Delta \phi_{k,b}^T \Phi_b \left(\hat{\phi}_{k,b} - \mathcal{U}_b \right) \\ &\leq -\Delta y^T \alpha \Delta y. \end{aligned} \quad (38)$$

Thus, the proof of Proposition 1 can be referred to that of Theorem 1.

Remark 3. It is noted that the considerations on visual kinematic model in this paper are similar to that in [19, 24, 25, 32, 33, 36], where the visual model parameters are uncalibrated in kinematic control design. However, the projection plane is set to be parallel to the operation plane in [32]. Also, note that eye-in-hand configuration is addressed in [19, 33] rather than the eye-to-hand setup in this paper, and in addition, only partial camera intrinsic parameters are taken into consideration in [33], and only the extrinsic camera parameters are considered in [19], respectively. Furthermore, the overparametrization problem is still unresolved in [24, 25], where a 2×14 regressor matrix needs to be determined. Extra two particular feature points are introduced in [36] for the purpose of preventing the direct use of image Jacobian matrix from control design. The major difference between the proposed AVSKC scheme and the uncalibrated visual tracking control scheme [19, 24, 25, 32, 33, 36] is that the parameters in image projection matrix, including the intrinsic and extrinsic camera parameters, are estimated by adaption laws while avoiding the overparametrization problem. This is realized by exploiting the structure of image Jacobian matrix inspired by [35] (see Property 1). Specifically, the salient features of the proposed AVSKC scheme lie in (I) the structurally simple implementation of control law τ_K (24); (II) the inexpensive obtainment of regressor matrices $E_{K,a}$ and $E_{K,b}$ in adaption laws (27) and (28) (or using (33) and (35)); (III) the nonsingularity property of $\hat{N}(y, \theta)$ and the receptivity of uncertain parameters processed by adaption laws (33) and (35).

Remark 4. If the actuators of WMR perform effectively, the visual tracking on image plane can be conveniently realized by the proposed AVSKC scheme. However, it is well recognized that the presence of dynamic uncertainties will cause a great negative impact on control performance. In the next section, we will pertinently propose the dynamic control scheme for visual servoing WMR together with the handling of dynamic uncertainties.

4. Adaptive Visual Servoing Dynamic Control for Wheeled Mobile Robot with Uncalibrated Visual Model and Dynamics

The focus in this section is extending the wheeled mobile robot kinematic control to dynamic control in the presence of uncalibrated visual model and uncertain dynamics; that is, both the kinematics and dynamics parameters are not required to be measured accurately. Note that, in this case, the designed controller becomes the dynamic input torque τ_D , in which case a deeper control loop is considered.

4.1. Controller Design. Define a referenced image velocity as

$$\dot{y}_r = \dot{y}_d - \gamma \Delta y, \quad (39)$$

where γ is a positive constant. Then, the reference errors of image velocity can be denoted as

$$r_y = \dot{y} - \dot{y}_r = \Delta \dot{y} + \gamma \Delta y. \quad (40)$$

Thus, the reference errors r_y contain both the image errors and image velocity errors. Furthermore, define a kinematic auxiliary variable

$$\tau_r = \hat{N}^{-1}(y, \theta) \hat{z} \dot{y}_r. \quad (41)$$

Differentiating (41) with respect to time, we have

$$\begin{aligned} \dot{\tau}_r &= -\hat{N}^{-1}(y, \theta) [\dot{\hat{N}}(y, \theta)] \hat{N}^{-1}(y, \theta) \\ &\quad + \hat{N}^{-1}(y, \theta) \dot{\hat{z}} \dot{y}_r + \hat{N}^{-1}(y, \theta) \hat{z} r_y. \end{aligned} \quad (42)$$

The purpose of designing this auxiliary variable is to connect the kinematic control variable τ_K , which can be specified as

$$r_\tau = \tau_K - \tau_r. \quad (43)$$

Note that, in (41), τ_K is a real response due to the evolution of robot dynamics (19) rather than a designed input. The relation between r_τ and r_y can be interestingly derived as

$$\begin{aligned} r_\tau^T \hat{N}^T(y, \theta) \beta_y \hat{z} r_y &= [\hat{N}(y, \theta) \tau_K - \hat{N}(y, \theta) \tau_r]^T \beta_y \hat{z} r_y \\ &= [\hat{N}(y, \theta) \tau_K - N(y, \theta) \tau_K + z \dot{y} - \hat{z} \dot{y} + \hat{z} \dot{y} - \hat{z} \dot{y}_r]^T \beta_y \hat{z} r_y, \end{aligned} \quad (44)$$

where β_y is a positive constant. Using Property 1, we can further obtain $\hat{N}(y, \theta) \tau_K - N(y, \theta) \tau_r = -E_{K,a}(\theta, \tau_K) \Delta \phi_{k,a} + E_{K,b}(\theta, \tau_K, y) \Delta \phi_{k,b}$ and $z \dot{y} - \hat{z} \dot{y} = \bar{E}_{K,b}(\theta, x, \dot{y}) \Delta \phi_{k,b}$. Thus, (44) can be rewritten as

$$\begin{aligned}
& r_\tau \widehat{N}^T(y, \theta) \beta_y \widehat{z} r_y \\
&= -\Delta \phi_{k,a}^T E_{K,a}^T(\theta, \tau_K) \beta_y \widehat{z} r_y + r_y^T \widehat{z} \beta_y \widehat{z} r_y \\
&\quad + \Delta \phi_{k,b}^T \left[\overline{E}_{K,b}^T(\theta, x, y) + E_{K,b}^T(\theta, \tau_K, y) \right] \beta_y \widehat{z} r_y.
\end{aligned} \tag{45}$$

Based on the kinematic control errors r_τ and reference errors r_y , the adaptive visual servoing dynamic controller (AVSDC) is proposed as

$$\tau_D = \overline{B}^{-1}(\theta) \left[E_D(\theta, \dot{\theta}, \tau_r, \dot{\tau}_r) \widehat{\phi}_d - \beta_\tau r_\tau - \widehat{N}^T(y, \theta) \beta_y \widehat{z} r_y \right]. \tag{46}$$

where β_τ is a positive constant.

Remark 5. Since $\overline{B}(\theta) = S^T(\theta)B(\theta)$ is related to the actuator dynamics rather than the robot dynamics, in this paper, we assume that the exact structure and parameters of $\overline{B}(\theta)$ are known. In particular, $\overline{B}(\theta)$ is defined as the $I_{2 \times 2}$ identity matrix in [6, 38] if the actuators are free of operation faults.

Remark 6. It seems interesting that, in the proposed AVSDC scheme, both image errors and velocity errors ($r_y = \Delta \dot{y} + \gamma \Delta y$) are introduced in the control law, which will strengthen the tracking performance and robustness of the

dynamic closed-loop system. As will be shown in the stability analysis, the asymptotical convergence of r_y leads to the asymptotical convergence of both $\Delta \dot{y}$ and Δy .

By substituting the AVSDC (46) into (19) and employing Property 4, one can obtain the closed-loop dynamics:

$$\overline{M}(\theta) \dot{r}_\tau + \overline{V}(\theta, \dot{\theta}) r_\tau = E_D(\theta, \dot{\theta}, \tau_r, \dot{\tau}_r) \Delta \phi_d - \beta_\tau r_\tau - \widehat{N}^T(y, \theta) \beta_y \widehat{z} r_y. \tag{47}$$

4.2. Unknown Parameter Estimation. In the AVSDC design (46), the estimated dynamics and visual kinematics are employed, and the estimated parameters are updated online by

$$\dot{\widehat{\phi}}_d = -\Phi_d^{-1} E_D^T(\theta, \dot{\theta}, \tau_r, \dot{\tau}_r) r_\tau, \tag{48}$$

$$\dot{\widehat{\phi}}_{a,i} = \text{proj}\{\overline{\mathcal{U}}_{a,i}\}, \tag{49}$$

$$\dot{\widehat{\phi}}_{b,i} = \text{proj}\{\overline{\mathcal{U}}_{b,i}\}, \tag{50}$$

where $\Phi_d \in \mathbb{R}^{e_3 \times e_3}$ is a positive-definite diagonal matrix, the projection operation is given in (36) and (37), and

$$\overline{\mathcal{U}}_a = \Phi_a^{-1} E_{K,a}^T(\theta, \tau_K) \beta_y \widehat{z} r_y, \overline{\mathcal{U}}_b = -\Phi_b^{-1} \left[\overline{E}_{K,b}^T(\theta, x, y) + E_{K,b}^T(\theta, \tau_K, y) \right] \beta_y \widehat{z} r_y. \tag{51}$$

4.3. Stability Analysis. Based on the above system analysis of closed-loop dynamics, we have the following result.

Theorem 2. Consider the visually servoed WMR system consisting of the uncalibrated kinematics (1), (2), (4), (6), and (8) and uncertain dynamics (19), under the control of AVSDC (46) with parameter updating laws (48), (49), and (50). Then, the closed-loop dynamics system of WMR is globally bounded, and the image errors are convergent to zero asymptotically.

Proof. Similarly, consider the Lyapunov function candidate

$$V_D = \frac{1}{2} r_\tau^T \overline{M}(\theta) r_\tau + \frac{1}{2} \Delta \phi_{k,a}^T \Phi_a \Delta \phi_{k,a} + \frac{1}{2} \Delta \phi_{k,b}^T \Phi_b \Delta \phi_{k,b} + \frac{1}{2} \Delta \phi_d^T \Phi_b \Delta \phi_d. \tag{52}$$

Taking the derivative of V_D yields

$$\dot{V}_D = -r_\tau^T \beta_\tau r_\tau - r_y^T \widehat{z} \beta_y \widehat{z} r_y + \Delta \phi_{k,a}^T \Phi_a \left[\dot{\widehat{\phi}}_{k,a} - \overline{\mathcal{U}}_a \right] + \Delta \phi_{k,b}^T \Phi_b \left[\dot{\widehat{\phi}}_{k,b} - \overline{\mathcal{U}}_b \right] \leq -r_\tau^T \beta_\tau r_\tau - r_y^T \widehat{z} \beta_y \widehat{z} r_y. \tag{55}$$

As $V_D \geq 0$ and $\dot{V}_D \leq 0$ are simultaneously established, we can obtain that $V_D(t)$ must be bounded; that is, r_τ , $\overline{M}(\theta)$, $\Delta \phi_{k,a}$, $\Delta \phi_{k,b}$, and $\Delta \phi_d$ are all bounded, giving rise to the

$$\dot{V}_D = r_\tau^T \overline{M}(\theta) \dot{r}_\tau + \frac{1}{2} r_\tau^T \dot{\overline{M}}(\theta) r_\tau + \Delta \phi_{k,a}^T \Phi_a \dot{\widehat{\phi}}_{k,a} \tag{53}$$

$$\Delta \phi_{k,b}^T \Phi_b \dot{\widehat{\phi}}_{k,b} + \Delta \phi_d^T \Phi_b \dot{\widehat{\phi}}_d.$$

Premultiplying both sides of (47) by r_τ^T and then substituting it into (53), we have

$$\begin{aligned}
\dot{V}_D &= r_\tau^T E_D(\theta, \dot{\theta}, \tau_r, \dot{\tau}_r) \Delta \phi_d - r_\tau^T \beta_\tau r_\tau - r_\tau^T \widehat{N}^T(y, \theta) \beta_y \widehat{z} r_y \\
&\quad + \Delta \phi_{k,a}^T \Phi_a \dot{\widehat{\phi}}_{k,a} + \Delta \phi_{k,b}^T \Phi_b \dot{\widehat{\phi}}_{k,b} + \Delta \phi_d^T \Phi_b \dot{\widehat{\phi}}_d,
\end{aligned} \tag{54}$$

where Property 3 is used. Subsequently, substituting (45) and the parameter updating laws (48), (49), and (50) into (54), we can obtain the following result:

boundedness of θ , $\widehat{\phi}_{k,a}$, $\dot{\widehat{\phi}}_{k,a}$, $\widehat{\phi}_{k,b}$, and $\widehat{\phi}_d$ since $\phi_{k,a}$, $\phi_{k,b}$, and ϕ_d are all constants. Moreover, $\widehat{N}(y, \theta)$ is bounded and nonsingular, and \widehat{z} , \dot{y}_r , and $r_y \in \mathcal{L}_\infty$ by observing (40) and

TABLE 1: Visual model parameters.

	f	u_k	v_k	u_o	v_o	ε
Camera parameters	0.04	2000	2000	300	300	$\pi/2$
Transformation matrix in parallel camera case	$\begin{bmatrix} \text{Rot}(x, \pi) & 1 \\ & 1 \\ & -3 \\ 0_{1 \times 3} & 1 \end{bmatrix}^{-1}$					
Transformation matrix in unparallel camera case	$\begin{bmatrix} \text{Rot}(x, \pi - \pi/10) & 1 \\ & 1 \\ & 3 \\ 0_{1 \times 3} & 1 \end{bmatrix}^{-1}$					

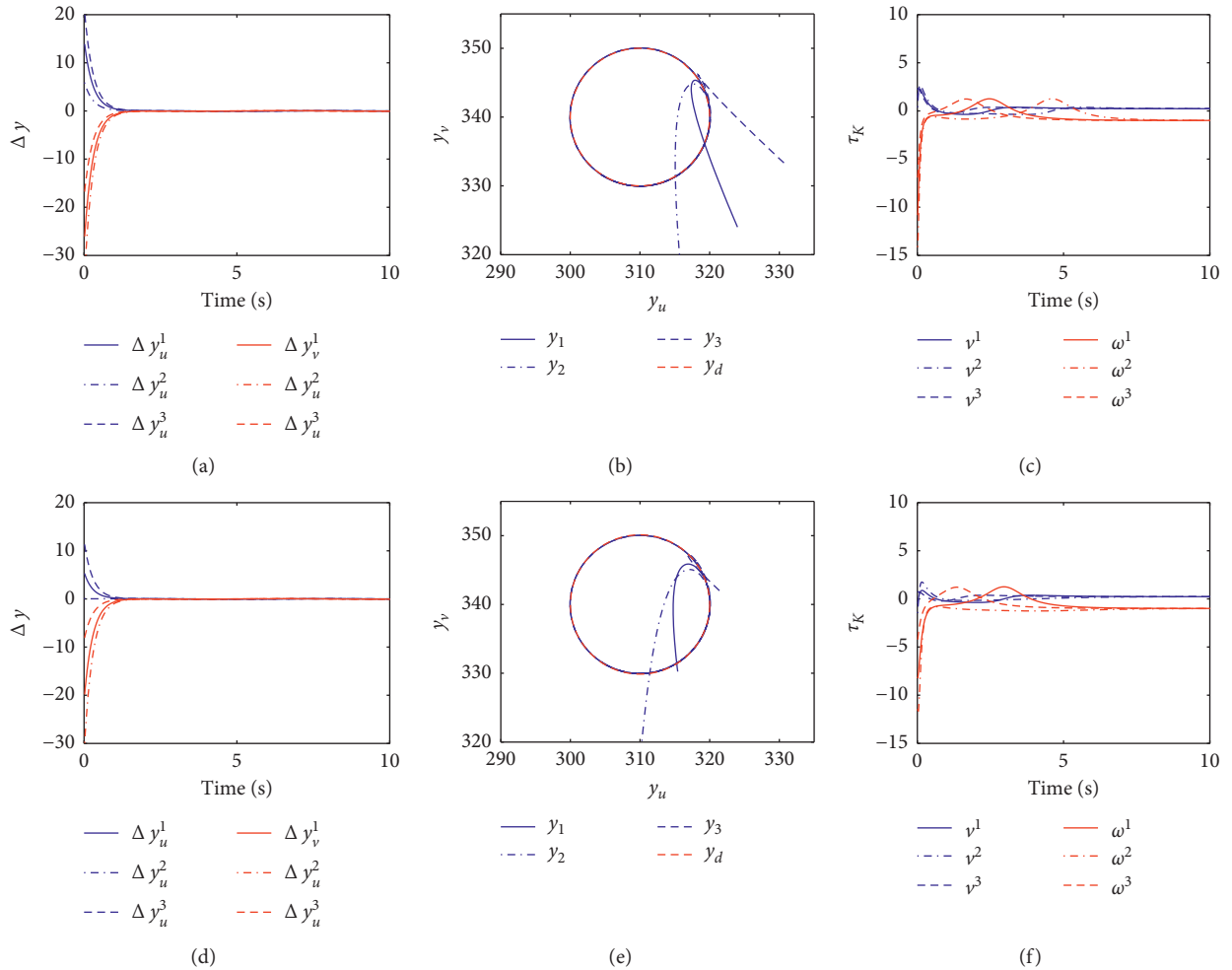


FIGURE 2: Control responses of AVSKC schemes. (a) Position errors on image plane in PCC. (b) The real and desired trajectory in PCC. (c) Control input in PCC. (d) Image errors in UPCC. (e) The real and desired trajectory in UPCC. (f) Control input in UPCC.

(43). From (42), we have $\dot{r}_r \in \mathcal{L}_\infty$, leading to $\tau_D \in \mathcal{L}_\infty$ from (46). According to the robot dynamics (19), we have $\dot{r}_K \in \mathcal{L}_\infty$, which directly implies that $\dot{r}_\tau \in \mathcal{L}_\infty$. Thus, the closed-loop dynamics of WMR in (47) is globally bounded.

Furthermore, from the result of (52) and (55), we get $r_\tau \in \mathcal{L}_\infty \cap \mathcal{L}_2$ and $\hat{z}r_y \in \mathcal{L}_\infty \cap \mathcal{L}_2$. Differentiating (8) with respect to time leads to $\dot{y} = (\dot{N}(y, \theta)\tau_K + N(y, \theta)\dot{r}_K)/z - N(y, \theta)\tau_K \dot{z}/z^2 \in \mathcal{L}_\infty$. Additionally, from the above

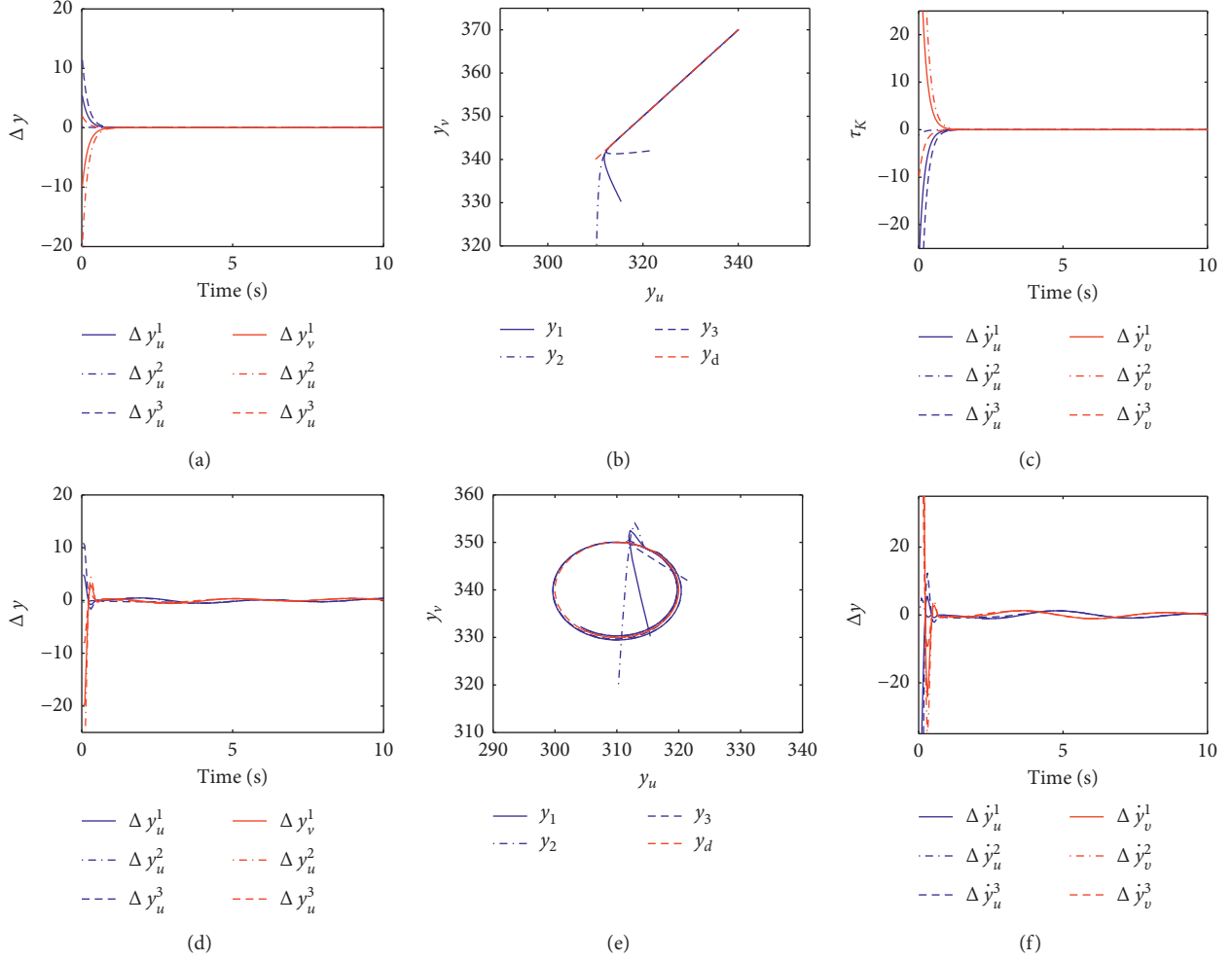


FIGURE 3: Control responses of AVSDC schemes. (a) Position errors on image plane in TLC. (b) The real and desired trajectory in TLC. (c) Velocity errors on image plane in TLC. (d) Position errors in TCC. (e) The real and desired trajectory in TCC. (f) Velocity errors on image plane in TCC.

analysis, $\ddot{y}_r = \ddot{y}_d - \gamma \Delta \dot{y} \in \mathcal{L}_\infty$, which thus results in $\hat{z}r_y + \hat{z}r_{\dot{y}} \in \mathcal{L}_\infty$. Therefore, we have $\lim_{t \rightarrow \infty} \hat{z}r_y \rightarrow 0$, since $\hat{z} \neq 0$ is defined by the projection function, which finally indicates that the image errors converge to zero such that $\lim_{t \rightarrow \infty} y - y_d \rightarrow 0$ and $\lim_{t \rightarrow \infty} \dot{y} - \dot{y}_d \rightarrow 0$.

Remark 7. Compared with the recent work on handling uncertain parameters for WMR [19, 33, 34, 36], where only kinematic uncertainties are addressed, this paper extends the uncalibrated visual servoing control to a dynamic control loop in the presence of parameter uncertainties and varying depth. It can be seen in AVSDC (46) that both the reference image errors r_y and kinematic control errors r_r , which contain velocity errors $\Delta \dot{y}$ and \dot{y}_r , are concurrently employed, giving rise to asymptotical convergence of both $\Delta \dot{y}$ and Δy by comparing with the AVSKC (24). Furthermore, the parameter uncertainties of visual kinematics and dynamics are adaptively compensated by the parameter

updating laws (48), (49), and (50). Actually, the AVSDC (46), to some extent, can be potentially regarded as containing the kinematic control by designing the kinematic auxiliary variable τ_r .

5. Numerical Simulations

In order to demonstrate the tracking performance of AVSKC and AVSDC schemes, simulation studies are carried out. As in Figure 1, a two-wheeled mobile robot with a camera in a fixed place is considered.

5.1. Trajectory Tracking for AVSKC. In the simulation task, we firstly address the AVSKC scheme under parallel and unparallel camera case. Assume that one feature point is marked on the WMR, and the distance d is set to be 0.3m. The simulated parameters of the perspective projection matrix in [8, 25] are given in Table 1, where f denotes the

focal length of the camera, u_k and v_k denote the scalar factors of two-dimensional axes on image plane, u_o and v_o are the positions of principal point, ε is the included angle between the coordinate axis and is assumed to be known, and the rotation matrix in transformation matrix is set as $\text{Rot}^{-1}(x, \pi)$ in parallel camera case and $\text{Rot}^{-1}(x, \pi - \pi/10)$ in unparallel camera case, respectively. Note that these parameters are only used to construct the simulated model, but are unavailable in the control design. For simplicity, the detailed expression of system model in (6) and (7) and Property 1 can be referred to in [8, 35]. The designed control gains and adaption gains are set as $\alpha = 10$ and $\Phi_a = \Phi_b = 10000$, respectively. The upper and lower bounds in parameter projection are designed as $\underline{\phi}_{a,i} = 0.6\phi_{a,i}$, $\bar{\phi}_{a,i} = 1.3\phi_{a,i}$, $\underline{\phi}_{b,i} = 0.6\phi_{b,i}$, and $\bar{\phi}_{b,i} = 1.3\phi_{b,i}$, and the initial parameters are $\hat{\phi}_{a,i}(0) = 0.8\phi_{a,i}$ and $\hat{\phi}_{b,i}(0) = 0.8\phi_{b,i}$, respectively. The initial states of WMR are $\theta^1(0) = \theta^2(0) = \theta^3(0) = 0$, where the superscript i denotes i th initial condition in simulation task, and the referenced trajectory is given as

$$y_d = \begin{bmatrix} 10 * \sin(t) + 310 \\ 10 * \cos(t) + 340 \end{bmatrix} \text{pixel.} \quad (56)$$

5.1.1. Parallel Camera Case (PCC). In this case, the image plane is parallel to the operation plane. The initial states of WMR are set as $x_b^1(0) = y_b^1(0) = 0.1\text{m}$, $x_b^2(0) = y_b^2(0) = 0.1\text{m}$, and $x_b^3(0) = y_b^3(0) = 0.1\text{m}$, respectively. The graphs in Figures 2(a)–2(c) demonstrate the corresponding simulation results, from which we can observe that the real trajectory converges to the referenced trajectory in about 2.5 s. Note that, in this case, the desired velocity \dot{y}_d is time varying; however, smoothly real trajectory and bounded states are still achievable.

5.1.2. Unparallel Camera Case (UPCC). In this case, the camera is placed in a position unparallel to the operation plane. The initial states of WMR are the same as in PCC. The simulation results are depicted in Figures 2(d)–2(f). Note that, in Figures 2(e) and 2(b), the initial points on image plane are noncoincident since the camera parameters are chosen in different values. Moreover, the image errors asymptotically converge to zero as expected, verifying the effectiveness of AVSKC scheme.

5.2. Trajectory Tracking for AVSDC. In this subsection, we will test the tracking performance of AVSDC scheme under tracking line case and tracking circle case. Due to the limitation of space, the parametric dynamics model of WMR is omitted, whose detailed expressions are given in [38], and the initial value for $\hat{\phi}$ is set as $\hat{\phi}_d(0) = 0.8\phi_d$. The designed gains are set as $\gamma = 5$, $\beta_\tau = 5$, $\beta_y = 10$, and $\Phi_d = 1000$; apart from this, all the simulated model and system parameters are given in UPCC.

5.2.1. Tracking Line Case (TLC). The reference line is given as

$$y_d = \begin{bmatrix} 3t + 310 \\ 3t + 340 \end{bmatrix} \text{pixel.} \quad (57)$$

In this case, the desired velocity is constant. Based on the theoretical analysis in Theorem 2, the real trajectory on the image plane asymptotically converges to the desired trajectory in the sense of position and velocity, confirmed by the simulation results in Figures 3(a)–3(c).

5.2.2. Tracking Circle Case (TCC). In this case, the desired trajectory in this case is chosen as in UPCC, and the external disturbance $f = [3 \sin(t), 3 \cos(t)]^T$ is applied to the robot dynamics such that

$$M(\theta)\ddot{q} + V(\theta, \dot{\theta})\dot{q} + G = B(\theta)\tau_D + A^T(\theta)\lambda + f. \quad (58)$$

The time histories of the corresponding results are plotted in Figures 3(d)–3(f). As predicted by Theorem 2, both Δy and $\Delta \dot{y}$ asymptotically converge to zero in about 1 s even under the influence of external interference. Furthermore, faster responses are expectedly obtained as compared with the simulation results in AVSKC scheme (see Figures 2(d) and 3(d)).

6. Conclusions

Two uncalibrated visual servoing control schemes for the wheeled mobile robot were developed from different perspectives, namely, the kinematic control and dynamic control. By utilizing the linearization characteristics of visual kinematics and robot dynamics, image-based tracking control laws (i.e., ASVKC and ASVDC) together with the parameter adaption algorithms were proposed to realize asymptotical convergence of image errors without the knowledge of visual model robot parameters. Furthermore, the overparametrization problem is avoided by exploiting the structure of depth-independent interaction matrix, giving less dimensional regressor matrices and simple configuration of parameter adaption laws. It was proven by the Lyapunov theory that both ASVKC and ASVDC schemes are capable of achieving global stability of closed-loop system. Lastly, numerical simulations were carried out to confirm the performance of ASVKC and ASVDC.

In this paper, we assume that the image trajectory is given in advance, and the external forces of robot system are not considered in complex environment. Furthermore, the applicability of visual servoing WMR control is worth further exploring. Thus, the further work encompasses the deterministic learning and accurate identification of system dynamics [41], the applicability of WMR with actuator constraint [42], and obstacle avoidance [43–45].

Data Availability

No underlying data are included.

Conflicts of Interest

The authors declare that they have no financial and personal relationships with other people or organizations that can inappropriately influence their work; there is no professional or other personal interest of any nature or kind in any product, service, and/or company that could be construed as influencing the position presented in, or the review of, the paper.

Acknowledgments

This work was supported in part by the Research start-up funds of DGUT (GC300501-113 and GC300501-111), in part by the Innovation Talent Program for Young Scholars of Guangdong Province (2018KQNCX252 and 2018KTSCX226), in part by the General program of Guangdong Natural Science Foundation (2019A1515010493), and in part by the Basic and Applied Basic Program of Guangdong Province (2019A1515110477, 2019A1515110476, and 2019B1515120076).

References

- [1] J. Luo, Z. Lin, Y. Li, and C. Yang, "A teleoperation framework for mobile robots based on shared control," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 377–384, 2020.
- [2] M. Wu, S.-L. Dai, and C. Yang, "Mixed reality enhanced user interactive path planning for omnidirectional mobile robot," *Applied Sciences*, vol. 10, no. 3, p. 1135, 2020.
- [3] J. R. García-Sánchez, R. Silva-Ortigoza, S. Tavera-Mosqueda et al., "Tracking control for mobile robots considering the dynamics of all their subsystems: experimental implementation," *Complexity*, vol. 2017, 2017.
- [4] C. Y. Chen, T. H. S. Li, and Y. C. Yeh, "Ep-based kinematic control and adaptive fuzzy sliding-mode dynamic control for wheeled mobile robots," *Information Sciences*, vol. 179, no. 1–2, p. 180, 2009.
- [5] H. Chen, S. Xu, L. Chu, F. Tong, and L. Chen, "Finite-time switching control of nonholonomic mobile robots for moving target tracking based on polar coordinates," *Complexity*, vol. 2018, 2018.
- [6] T. Fukao, H. Nakagawa, and N. Adachi, "Adaptive tracking control of a nonholonomic mobile robot," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 5, pp. 609–615, 2000.
- [7] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, "Robot learning system based on adaptive neural control and dynamic movement primitives," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 777–787, 2019.
- [8] Y.-H. Liu, H. Wang, C. Wang, and K. K. Lam, "Uncalibrated visual servoing of robots using a depth-independent interaction matrix," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 22, no. 4, pp. 804–817, 2006.
- [9] G. D. Hager, "Special section on vision-based control of robot manipulators," *IEEE Trans Robotics Automation*, vol. 12, pp. 651–670, 1996.
- [10] A. V. Kudryavtsev, M. T. Chikhaoui, A. Liadov et al., "Eye-in-hand visual servoing of concentric tube robots," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 2315–2321, 2018.
- [11] G. Chesi, K. Hashimoto, D. Prattichizzo, and A. Vicino, "Keeping features in the field of view in eye-in-hand visual servoing: a switching approach," *IEEE Transactions on Robotics*, vol. 20, no. 5, pp. 908–913, 2004.
- [12] J. A. Piepmeyer, G. V. McMurray, and H. Lipkin, "Uncalibrated dynamic visual servoing," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 1, pp. 143–147, 2004.
- [13] Y. Shen, D. Sun, Y. H. Liu, and K. Li, "Asymptotic trajectory tracking of manipulators using uncalibrated visual feedback," *IEEE/ASME Transactions on Mechatronics*, vol. 8, no. 1, pp. 87–98, 2003.
- [14] J. Ling, Y. Li, and Y. Jia, "Visual tracking strategy of a wheeled mobile robot and a multi-dof crane equipped with a camera," in *Proceedings of the 32nd Chinese Control Conference*, pp. 5489–5493, New York, NY, USA, 2013.
- [15] X. Zhang, Y. Fang, and X. Liu, "Visual servoing of non-holonomic mobile robots based on a new motion estimation technique," in *Proceedings of the 48th IEEE Conference on Decision and Control (CDC) Held Jointly with 2009 28th Chinese Control Conference*, pp. 8428–8433, New York, NY, USA, 2009.
- [16] R. S. Sharma, S. Shukla, L. Behera, and V. K. Subramanian, "Position-based visual servoing of a mobile robot with an automatic extrinsic calibration scheme," *Robotica*, vol. 38, no. 5, pp. 831–844, 2020.
- [17] H. Kwon, Y. Yoon, J. B. Park, and A. C. Kak, "Person tracking with a mobile robot using two uncalibrated independently moving cameras," in *Proceedings of the IEEE International Conference on Robotics & Automation*, pp. 2877–2883, New York, NY, USA, 2005.
- [18] B. Li, Y. Fang, and X. Zhang, "Visual servo regulation of wheeled mobile robots with an uncalibrated onboard camera," *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 5, pp. 2330–2342, 2016.
- [19] X. Zhang, Y. Fang, B. Li, and J. Wang, "Visual servoing of nonholonomic mobile robots with uncalibrated camera-to-robot parameters," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 1, pp. 390–400, 2017.
- [20] K. Zhang, J. Chen, G. Yu, X. Zhang, and Z. Li, "Visual trajectory tracking of wheeled mobile robots with uncalibrated camera extrinsic parameters," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 23, pp. 1–10, 2020.
- [21] W. Mackunis, N. Gans, A. Parikh, and W. E. Dixon, "Unified tracking and regulation visual servo control for wheeled mobile robots," *Asian Journal of Control*, vol. 16, no. 3, pp. 669–678, 2014.
- [22] J. Chen and D. M. Dawson, "Monocular camera visual servo control of wheeled mobile robots," *International Journal of Robotics & Automation*, vol. 26, no. 1, pp. 26–34, 2011.
- [23] H. Loulin, "Control approach for tracking a moving target by a wheeled mobile robot with limited velocities," *Iet Control Theory & Applications*, vol. 3, no. 12, pp. 1565–1577, 2009.
- [24] X. Liang, H. Wang, and W. Chen, "Adaptive image-based visual servoing of wheeled mobile robots with fixed camera configuration," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, p. 6199, New York, NY, USA, 2014.
- [25] X. Liang, H. Wang, W. Chen, D. Guo, and T. Liu, "Adaptive image-based trajectory tracking control of wheeled mobile robots with an uncalibrated fixed camera," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 6, pp. 2266–2282, 2015.
- [26] N. Sarkar, X. Yun, and V. Kumar, "Control of mechanical systems with rolling constraints application to dynamic control of mobile robots," *International Journal of Robotics Research*, vol. 13, no. 1, pp. 55–69, 1992.

- [27] G. Corradini, "Robust tracking control of mobile robots in the presence of uncertainties in the dynamical model," *Journal of Robotic Systems*, vol. 18, no. 6, pp. 317–323, 2001.
- [28] M. K. Bugeja, S. G. Fabri, and L. Camilleri, "Dual adaptive dynamic control of mobile robots using neural networks," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 1, pp. 129–141, 2009.
- [29] C. Yang, Y. Jiang, W. He, J. Na, Z. Li, and B. Xu, "Adaptive parameter estimation and control design for robot manipulators with finite-time convergence," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 10, pp. 8112–8123, Oct. 2018.
- [30] C. Wang, Z. Liang, and Y. Liu, "Dynamic feedback robust regulation of nonholonomic mobile robots based on visual servoing," in *Proceedings of the Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, pp. 4384–4389, London, UK, 2009.
- [31] F. Yang and C.-L. Wang, "Adaptive stabilization for uncertain nonholonomic dynamic mobile robots based on visual servoing feedback," *Acta Automatica Sinica*, vol. 37, no. 7, pp. 857–864, 2011.
- [32] W. E. Dixon, D. M. Dawson, E. Zergeroglu, and A. Behal, "Adaptive tracking control of a wheeled mobile robot via an uncalibrated camera system," *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 31, no. 3, pp. 341–352, 2001.
- [33] K. Zhang, J. Chen, Y. Li, and Y. Gao, "Unified visual servoing tracking and regulation of wheeled mobile robots with an uncalibrated camera," *IEEE/ASME Transactions on Mechatronics*, vol. 23, no. 4, pp. 1728–1739, 2018.
- [34] X. Liang, H. Wang, Y.-H. Liu, W. Chen, G. Hu, and J. Zhao, "Adaptive task-space cooperative tracking control of networked robotic manipulators without task-space velocity measurements," *IEEE Transactions on Cybernetics*, vol. 46, no. 10, pp. 2386–2398, 2016.
- [35] H. Wang, "Adaptive visual tracking for robotic systems without image-space velocity measurement," *Automatica*, vol. 55, pp. 294–301, 2015.
- [36] X. Liang, H. Wang, Y.-H. Liu, W. Chen, and Z. Jing, "Image-based position control of mobile robots with a completely unknown fixed camera," *IEEE Transactions on Automatic Control*, vol. 63, no. 9, pp. 3016–3023, 2018.
- [37] P. Coelho and U. Nunes, "Path-following control of mobile robots in presence of uncertainties," *IEEE Transactions on Robotics*, vol. 21, no. 2, pp. 252–261, 2005.
- [38] Z.-G. Hou, A.-M. Zou, L. Cheng, and M. Tan, "Adaptive control of an electrically driven nonholonomic mobile robot via backstepping and fuzzy approach," *IEEE Transactions on Robotics*, vol. 17, no. 4, pp. 803–815, 2009.
- [39] W. E. Dixon, "Adaptive regulation of amplitude limited robot manipulators with uncertain kinematics and dynamics," *IEEE Transactions on Automatic Control*, vol. 52, no. 3, pp. 488–493, 2007.
- [40] Y.-C. Liu and M.-H. Khong, "Adaptive control for nonlinear teleoperators with uncertain kinematics and dynamics," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 5, pp. 2550–2562, 2015.
- [41] N. Wang, C. Chen, and A. Di Nuovo, "A framework of hybrid force/motion skills learning for robots," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 1, 2020.
- [42] F. Wang, Z. Liu, C. L. P. Chen, and Y. Zhang, "Synchronized adaptive control for coordinating manipulators with time-varying actuator constraint and uncertain dynamics," *International Journal of Robust and Nonlinear Control*, vol. 29, no. 12, pp. 4149–4167, 2019.
- [43] H. Kong, C. Yang, G. Li, and S.-L. Dai, "A semg-based shared control system with no-target obstacle avoidance for omnidirectional mobile robots," *IEEE Access*, vol. 8, pp. 26030–26040, 2020.
- [44] H. Lin, T. Zhang, Z. Chen, H. Song, and C. Yang, "Adaptive fuzzy Gaussian mixture models for shape approximation in robot grasping," *International Journal of Fuzzy Systems*, vol. 21, no. 4, pp. 1026–1037, 2019.
- [45] Y. Kong, R. Zhang, Y. Jiang, and X. Xia, "A repeatable optimization for kinematic energy system with its mobile manipulator application," *Complexity*, vol. 2019, 2019.

Research Article

Disturbance-Observer-Based Fuzzy Control for a Robot Manipulator Using an EMG-Driven Neuromusculoskeletal Model

Longbin Zhang ¹, Wen Qi,² Yingbai Hu ³ and Yue Chen⁴

¹Department of Engineering Mechanics, KTH Royal Institute of Technology, SE-100 44 Stockholm, Sweden

²Dipartimento Di Elettronica, Informazione e Bioingegneria, Politecnico Di Milano, 20133 Milano, Italy

³Department of Informatics, Technical University of Munich, Munich 85748, Germany

⁴Department of Mechanical Engineering, University of Arkansas, Fayetteville 72701, AR, USA

Correspondence should be addressed to Longbin Zhang; longbin@kth.se

Received 18 September 2020; Revised 1 November 2020; Accepted 17 November 2020; Published 30 November 2020

Academic Editor: Ning Wang

Copyright © 2020 Longbin Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Robot manipulators have been extensively used in complex environments to complete diverse tasks. The teleoperation control based on human-like adaptivity in the robot manipulator is a growing and challenging field. This paper developed a disturbance-observer-based fuzzy control framework for a robot manipulator using an electromyography- (EMG-) driven neuromusculoskeletal (NMS) model. The motion intention (desired torque) was estimated by the EMG-driven NMS model with EMG signals and joint angles from the user. The desired torque was transmitted into the desired velocity for the robot manipulator system through an admittance filter. In the robot manipulator system, a fuzzy logic system, utilizing an integral Lyapunov function, was applied for robot manipulator systems subject to model uncertainties and external disturbances. To compensate for the external disturbances, fuzzy approximation errors, and nonlinear dynamics, a disturbance observer was integrated into the controller. The developed control algorithm was validated with a 2-DOFs robot manipulator in simulation. The results indicate the proposed control framework is effective and crucial for the applications in robot manipulator control.

1. Introduction

Robotic manipulators are increasingly used in welding automation, robotic surgery [1], and space, as they are able to complete diverse tasks in complex environments, such as uncertain system dynamics, time-vary delays, and unknown external disturbances. The robot manipulator may work within dangerous environments for unfriendly tasks, such as handling radioactive material and searching, and the teleoperation control of the robot manipulator has been widely utilized into the controller design. Yang et al. [2] proposed an admittance-adaptation-based methodology for robot manipulators when interacting with unknown environments and guaranteed trajectory tracking performance. Recently, there is a growing demand for the natural interface between the robotic manipulator and the user [3]. Specifically, the robotic manipulator is teleoperated by the user with human-like

characteristics. A human-like learning controller was proposed to optimally adapt interaction with unknown environments [4], and the human-like adaptivity was shown well by the robot manipulator in stable and unstable tasks.

The development of robot manipulator controller with human-like characteristics (the user's intention) requires accurate and robust decoding of motor function. Muscle electromyographic (EMG) signals from the central nervous system (CNS) [5, 6] are widely used in the user's intention detection as EMG signals are relatively easy to acquire and process and provide essential information on human motion. EMG-based modelling methodologies have been utilized into various human-machine control algorithms for robot manipulators. Ryu et al. [7] developed a continuous position-based strategy for a robot manipulator with EMG signals and the manipulator could replicate the movements from the user well, thereby improving the control strategy

for teleoperated robot manipulators. Artemiadis et al. [3] proposed a teleportation methodology for a robot manipulator based on EMG signals and position feedback and realized a good master-slave manipulator system with human-like adaptivity. Bu et al. [8] proposed a Bayesian-network-based model to predict occurrence probabilities of the motions with the given information of the previous motion and classify hybrid motions with EMG signals. Their results demonstrated that the EMG-based Bayesian network model could improve the robustness and stability for motion classification. Therefore, integrating an EMG-driven neuromusculoskeletal (NMS) model with human-like characteristics into the robot manipulator controller is valuable and could be crucial in the robot manipulator control field [9].

Diverse control strategies for robot manipulators have been developed in complex environments [10–12], as these environments could degrade the performance of the robot manipulator system [13, 14]. Lin et al. [15] developed fuzzy Gaussian mixture models to approximate the objects' shape for robot manipulator grasping tasks under unknown environments, and the model has a good grasp quality. He et al. [16] proposed a disturbance-observer-based control strategy to approximate unknown parameters and disturbance for multimanipulator robots and validated on a dual-arm cooperative robot (Baxter). Their results showed that the controller has an accurate control performance and was able to compensate for the errors due to the model uncertainties and external disturbances. Yang et al. [17] proposed a disturbance-observer-based impedance control for uncertain robot manipulators in unknown environments. Fuzzy logic system and disturbance observer are two commonly used techniques in the control system to compensate for unknown functions, parameters, and disturbance [18–20]. Therefore, in this study, a disturbance-observer-based fuzzy algorithm was integrated into the control framework for a robot manipulator, subjected to practical problems including external disturbances and model uncertainties.

The objective of this study was to design a disturbance-observer-based fuzzy controller for a 2-DOFs robot manipulator using an EMG-driven NMS model. The main contributions of this paper are as follows: (1) a disturbance-observer-based fuzzy control framework is developed that fully incorporates a robot manipulator system with an EMG-driven NMS model and could be applied in controlling the movement of robot manipulators with human-like characteristics; (2) the user's motion intention was able to be well-predicted by the EMG-driven model and transmitted into the desired velocity through an admittance filter, which allows the robot manipulator system behave with human-like adaptivity; (3) in the current paper, the control framework we proposed in general is useful for obtaining the human-like characteristics, as well as simulating the control strategy in the robot manipulator system, and thus be of large benefit for teleoperation robot manipulator applications.

2. Methods

2.1. Controller Framework. A disturbance-observer-based fuzzy control framework (Figure 1) was developed for a robot manipulator system in complex environments. In the robot manipulator controller design, to complete tasks with human-like characteristics, an EMG-driven NMS model through the EMG signals and joint angles from the user was used to estimate the desired torque. The desired torque was transmitted into the desired velocity through an admittance filter. A disturbance-observer-based adaptive controller with fuzzy compensation was developed for the robot manipulator with unknown model dynamics.

2.2. EMG-Driven Neuromusculoskeletal Model. A previously developed EMG-driven NMS model [21] was used in this study. It reproduces the transformations from EMG signal generation and joint angles to musculotendon forces and joint torques. The NMS model consists of four components: musculotendon kinematics, muscle contraction dynamics, muscle activation dynamics, and joint dynamics [22].

The musculotendon kinematics component used the 3D joint angles to calculate musculotendon lengths and moment arms of individual musculotendon units (MTUs) through a musculoskeletal model. The muscle activation dynamics component calculated muscle activation based on filtered EMG signals. The relation between neural activation $u(t)$ and filtered EMG signal $e(t)$ was represented by a recursive filter [23] as shown in the following equation:

$$u_i(t) = \mu e_i(t - d) - \beta_1 u_i(t - 1) - \beta_2 u_i(t - 2), \quad (1)$$

where $e_i(t)$ is the linear envelope of the EMG signal of i^{th} muscle; $u_i(t)$ is the neural activation of i^{th} muscle; d is the electromechanical delay; μ is the muscle gain coefficient; and β_1 and β_2 are the recursive coefficients and are subject to the following constraints to obtain a stable solution [21, 23, 24]: $\beta_1 = C_1 + C_2$, $\beta_2 = C_1 \cdot C_2$, where $|C_1| < 1$, $|C_2| < 1$, and $\mu - \beta_1 - \beta_2 = 1$.

The relationship from neural activation to muscle activation is nonlinear and was formulated to describe muscle activation dynamics as follows:

$$a_i(t) = \frac{e^{A_i u_i(t)} - 1}{e^{A_i} - 1}, \quad (2)$$

where $a_i(t)$ is the i^{th} muscle activation and A_i is the nonlinear shape factor of i^{th} muscle and subjected to the interval $(-3, 0)$ with zero representing a linear relationship and negative values introduce a nonlinear relationship [24, 25].

Musculotendon forces were computed through muscle contraction dynamics, based on a 3-element Hill-type muscle model; a series elastic element (SE), a contractile element (CE), and a parallel elastic element (PE). Each MTU's force (F^{mt}) could be represented as a function of

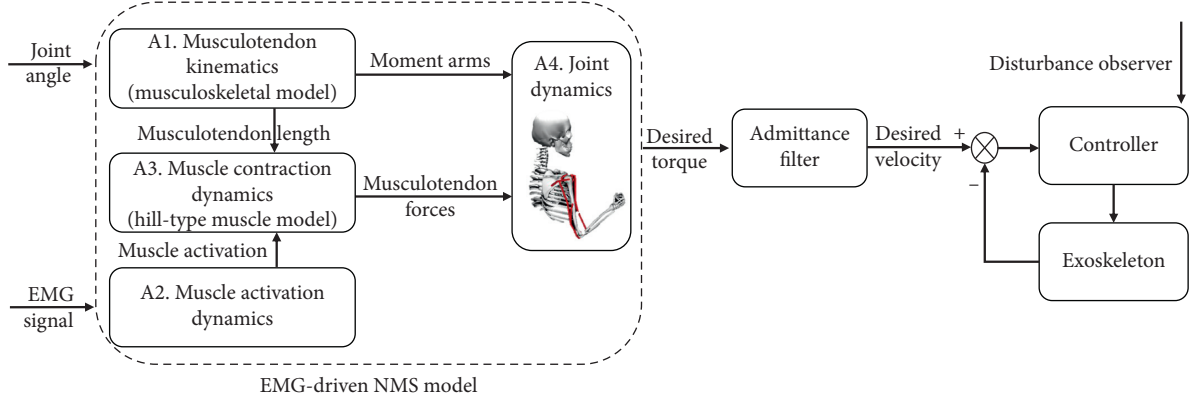


FIGURE 1: Schematic structure of the controller framework based on an EMG-driven NMS model. The NMS model consists of four components: (A1) the model's musculotendon kinematics were used to compute musculotendon lengths and moment arms; (A2) muscle activation dynamics were employed to calculate the level of muscle activation involved in the processed EMG signals; (A3) muscle contraction dynamics according to a Hill-type muscle model were applied to predict musculotendon force, using calculated musculotendon length and muscle activation as inputs; (A4) joint dynamics was used to compute joint torques, using the calculated musculotendon forces and moment arms as inputs. The EMG signals and joint angles from the user were used to estimate the motion intention (desired torque) through the EMG-driven NMS model. Then, the desired torque was transmitted to the desired velocity through an admittance filter. A disturbance-observer-based adaptive fuzzy controller was developed for the robot manipulator system with model uncertainties.

muscle activation and muscle kinematics as shown in the following equation:

$$F^{mt} = F_0^m [f_a(\tilde{l}_m) \cdot f_v(\tilde{v}_m) \cdot a + f_p(\tilde{l}_m) + d_m \tilde{v}_m] \cos(\phi), \quad (3)$$

where F_0^m is the maximum isometric muscle force; $f_a(\tilde{l}_m)$ is the active force-length relationship that describes the ability of muscle fibres to generate forces at different lengths; \tilde{l}_m is the fibre length normalized with the optimal fibre length; $f_v(\tilde{v}_m)$ is the force-velocity relationship that represents the muscle fibre force contribution of the fibres' contraction velocity (\tilde{v}_m), and the velocity was normalized with maximum contraction velocity and optimal fibre length; $f_p(\tilde{l}_m)$ is the passive force-length relationship that expresses the force response to the fibres to strain; d_m is the muscle damping coefficient which represents the muscle damping characteristics; and ϕ is the pennation angle of the fibres.

Joint torques were then estimated by the product of musculotendon forces F_{mt} and moment arms r_{mt} through the joint dynamics component as shown in the following equation:

$$M = r_{mt} \times F^{mt}. \quad (4)$$

2.3. Admittance Filter. To control the robot manipulator based on the user's adaptivity (desired torque estimated from EMG signals), an admittance filter was used to transform the joint torque into the desired angular velocity [26, 27].

$$\frac{\Delta\tau(s)}{Ms^2 + Bs + K} = \Delta q(s), \quad (5)$$

where $\Delta\tau$ and Δq are the torque and position regulation and M , B and K are the mass, damping, and stiffness parameters of the admittance filter, respectively.

2.4. Robot Manipulator Dynamic Modelling. The dynamics of a robot manipulator was generally modeled as follows [28]:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + f_{dis} = \tau, \quad (6)$$

where $M(q)$ is inertial matrix; q is the joint angle; $C(q, \dot{q})$ is centripetal and Coriolis force; $G(q)$ is the gravitational force; τ is the joint torque; and f_{dis} is the unknown disturbance.

We rewrite this robot manipulator dynamics as follows:

$$\begin{cases} \mathbf{y} = \mathbf{x}_1, \\ \dot{\mathbf{x}}_2 = \mathbf{B}^{-1}(\mathbf{x})[\mathbf{U}(\mathbf{x}) + \mathbf{r} + \boldsymbol{\tau}], \\ \dot{\mathbf{x}}_1 = \mathbf{x}_2, \end{cases} \quad (7)$$

where $\mathbf{x}_1 = [q_1, q_2, \dots, q_m]^T$, $\mathbf{x}_2 = [\dot{q}_1, \dot{q}_2, \dots, \dot{q}_m]^T$, $\mathbf{B}(\mathbf{x}) = M(q)$, $\mathbf{U}(\mathbf{x}) = -C(q, \dot{q})\dot{q} - G(q)$, $\mathbf{r} = -f_{dis}$, and $\boldsymbol{\tau} = [\tau_1, \dots, \tau_m]^T$.

We aim to design an adaptive controller that could ensure the robot manipulator system has satisfactory tracking performance under input nonlinearity and uncertain model dynamics.

The $\mathbf{B}(\mathbf{x})$ could be divided as follows:

$$\mathbf{B}(\mathbf{x}) = \mathbf{B}_d(\mathbf{x}) + \Delta_{\mathbf{B}}, \quad (8)$$

where $\mathbf{B}_d(\mathbf{x})$ component matrix is diagonal and $\Delta_{\mathbf{B}}$ component is unknown.

With (7), we could obtain

$$\begin{aligned} \mathbf{B}_d(\mathbf{x})\dot{\mathbf{x}}_2 &= (\mathbf{I} - \Delta_{\mathbf{B}}\mathbf{B}^{-1}(\mathbf{x}))\mathbf{U}(\mathbf{x}) + \boldsymbol{\tau} \\ &+ (\mathbf{I} - \Delta_{\mathbf{B}}\mathbf{B}^{-1}(\mathbf{x}))\mathbf{r} - \Delta_{\mathbf{B}}\mathbf{B}^{-1}(\mathbf{x})\boldsymbol{\tau}, \quad (9) \\ &= \mathcal{F}(\mathbf{x}) + \mathcal{R} + \boldsymbol{\tau} + \boldsymbol{p}, \end{aligned}$$

where $\mathcal{F}(\mathbf{x}) = (\mathbf{I} - \Delta_{\mathbf{B}}\mathbf{B}^{-1}(\mathbf{x}))\mathbf{U}(\mathbf{x}) \in R^m$, $\mathcal{R} = (\mathbf{I} - \Delta_{\mathbf{B}}\mathbf{B}^{-1}(\mathbf{x}))\mathbf{r} \in R^m$, and $\boldsymbol{p} = -\Delta_{\mathbf{B}}\mathbf{B}^{-1}(\mathbf{x})\boldsymbol{\tau} \in R^m$.

The tracking errors \mathbf{e}_i are represented as

$$\mathbf{s}_i = \dot{e}_i + \lambda_i \mathbf{e}_i, \quad (10)$$

$$\mathbf{e}_i = \mathbf{y}_i - \mathbf{y}_{di}, \quad i = 1, \dots, m, \quad (11)$$

where $\lambda_1, \lambda_2, \dots, \lambda_m$ are controlled parameters.

Therefore, combining (7) with (10), we can obtain

$$\dot{\mathbf{s}} = \mathbf{B}_d^{-1}(\mathbf{x})[\mathcal{F}(\mathbf{x}) + \mathcal{R} + \tau + p] + \mathbf{v}, \quad (12)$$

where $\mathbf{s} = [\mathbf{s}_1, \dots, \mathbf{s}_m]^T$ and $\mathbf{v} = [\nu_1, \dots, \nu_m]^T$ with $\nu_i = -\mathbf{y}_{di}^{(2)} + \lambda_i \dot{e}_i$, $i = 1, \dots, m$.

2.5. Integral Lyapunov Analysis. An integral Lyapunov function is modeled as

$$\mathbf{V}_1 = \mathbf{s}^T \mathbf{B}_9 \mathbf{s}, \quad (13)$$

where

$$\mathbf{B}_9 = \int_0^1 \vartheta \mathbf{B}_\alpha d\vartheta = \text{diag} \left[\int_0^1 \vartheta \mathbf{B}_{\alpha ii}(\bar{\mathbf{x}}, t\vartheta \mathbf{n}_s, q + h\nu_i) d\vartheta \right], \quad (14)$$

with $\mathbf{B}_\alpha = \mathbf{B}_d(\bar{\mathbf{x}}, \vartheta \mathbf{s}_i + \mathbf{v}_i) \alpha = \text{diag}[b_{dii}(\bar{\mathbf{x}}, \vartheta \mathbf{s}_i + \mathbf{v}_i) \alpha_{ii}]_{m \times m}$, $\bar{\mathbf{x}} = \mathbf{x}_1$, $\alpha \in R^{m \times m}$, and $\alpha_{11} = \dots = \alpha_{mm}$. $\mathbf{v} = \dot{y}_d - \xi$ where $\xi = [\xi_1, \xi_2, \dots, \xi_m]^T \in R^m$ with $\xi_i = \lambda_i \mathbf{e}_i$, $i = 1, 2, \dots, m$.

According to (14), then (13) is remodelled as a high-dimensional Lyapunov function

$$\mathbf{V}_1 = \sum_{i=1}^m \mathbf{s}_i^2 \int_0^1 \vartheta \mathbf{B}_{\alpha ii}(\bar{\mathbf{x}}, \vartheta \mathbf{s}_i + \mathbf{v}_i) d\vartheta. \quad (15)$$

Based on the definition of \mathbf{B}_α , \mathbf{B}_α has minimum and maximum eigenvalues $\lambda_{\min}(\mathbf{B}_\alpha)$ and $\lambda_{\max}(\mathbf{B}_\alpha)$,

$$0 \leq \lambda_{\min}(\mathbf{B}_\alpha) \mathbf{s}^T \mathbf{s} \leq \mathbf{s}^T \mathbf{B}_\alpha \mathbf{s} \leq \lambda_{\max}(\mathbf{B}_\alpha) \mathbf{s}^T \mathbf{s}. \quad (16)$$

Considering ϑ component in \mathbf{B}_α function, which is independent of \mathbf{s} , $\bar{\mathbf{x}}$, and \mathbf{v} , we can obtain

$$0 \leq \mathbf{s}^T \left(\int_0^1 \vartheta \mathbf{B}_\alpha d\vartheta \right) \mathbf{s} \leq \left(\int_0^1 \vartheta \lambda_{\max}(\mathbf{B}_\alpha) d\vartheta \right) \mathbf{s}^T \mathbf{s}. \quad (17)$$

Thus, $\mathbf{V}_1 \geq 0$ is proved.

As \mathbf{B}_α and \mathbf{B}_9 are symmetric, the derivative of \mathbf{V}_1 is modeled as

$$\dot{\mathbf{V}}_1 = 2\mathbf{s}^T \mathbf{B}_9 \dot{\mathbf{s}} + \mathbf{s}^T \left(\frac{\partial \mathbf{B}_9}{\partial \mathbf{s}} \dot{\mathbf{s}} \right) \mathbf{s} + \mathbf{s}^T \left(\frac{\partial \mathbf{B}_9}{\partial \bar{\mathbf{x}}} \dot{\bar{\mathbf{x}}} \right) \mathbf{s} + \mathbf{s}^T \left(\frac{\partial \mathbf{B}_9}{\partial \mathbf{v}} \dot{\mathbf{v}} \right) \mathbf{s}, \quad (18)$$

with

$$\begin{aligned} \frac{\partial \mathbf{B}_9}{\partial \mathbf{s}} \dot{\mathbf{s}} &= \text{diag} \left[\int_0^1 \vartheta \frac{\partial \mathbf{B}_{\alpha ii}}{\partial \mathbf{s}_i} \dot{\mathbf{s}}_i d\vartheta \right], \\ \frac{\partial \mathbf{B}_9}{\partial \bar{\mathbf{x}}} \dot{\bar{\mathbf{x}}} &= \text{diag} \left[\int_0^1 \vartheta \sum_{j=1}^m \frac{\partial \mathbf{B}_{\alpha ii}}{\partial \bar{\mathbf{x}}_j} \dot{\bar{\mathbf{x}}}_j d\vartheta \right], \\ \frac{\partial \mathbf{B}_9}{\partial \mathbf{v}} \dot{\mathbf{v}} &= \text{diag} \left[\int_0^1 \vartheta \frac{\partial \mathbf{B}_{\alpha ii}}{\partial \mathbf{v}_i} \dot{\mathbf{v}}_i d\vartheta \right], \quad i = 1, \dots, m. \end{aligned} \quad (19)$$

Considering the following equations,

$$\frac{\partial \mathbf{B}_9}{\partial \mathbf{s}} \mathbf{s} = \text{diag} \left[\int_0^1 \vartheta \frac{\partial \mathbf{B}_{\alpha ii}}{\partial \mathbf{s}_i} \mathbf{s}_i d\vartheta \right] = \int_0^1 \vartheta^2 \frac{\partial \mathbf{B}_\alpha}{\partial \vartheta} d\vartheta, \quad (20)$$

we have

$$\begin{aligned} \mathbf{s}^T \left(\frac{\partial \mathbf{B}_9}{\partial \mathbf{s}} \dot{\mathbf{s}} \right) \mathbf{s} &= \mathbf{s}^T \left(\left[\vartheta^2 \mathbf{B}_\alpha \right]_0^1 - 2 \int_0^1 \vartheta \mathbf{B}_\alpha d\vartheta \right) \dot{\mathbf{s}} \\ &= \mathbf{s}^T \mathbf{B}_\alpha \dot{\mathbf{s}} - 2\mathbf{s}^T \mathbf{B}_9 \dot{\mathbf{s}}. \end{aligned} \quad (21)$$

As ϑ is a scalar, and $\sigma = \vartheta \mathbf{s}_i$, we have $(\partial \mathbf{B}_9 / \partial \mathbf{v}) \mathbf{s} = \text{diag} \left[\int_0^1 \vartheta (\partial \mathbf{B}_{\alpha ii} / \partial \mathbf{v}_i) \mathbf{s}_i d\vartheta \right] = \int_0^1 \vartheta (\partial \mathbf{B}_\alpha / \partial \vartheta) d\vartheta$ and $\nu = -\dot{\mathbf{v}}$. Then, we have

$$\begin{aligned} \mathbf{s}^T \left(\frac{\partial \mathbf{B}_9}{\partial \mathbf{v}} \dot{\mathbf{v}} \right) \mathbf{s} &= \mathbf{s}^T \left(- \int_0^1 \vartheta \frac{\partial \mathbf{B}_\alpha}{\partial \vartheta} d\vartheta \right) \mathbf{v} \\ &= -\mathbf{s}^T \mathbf{B}_\alpha \mathbf{v} + \mathbf{s}^T \int_0^1 \mathbf{B}_\alpha \mathbf{v} d\vartheta. \end{aligned} \quad (22)$$

Combining (21) and (22), (18) can be rewritten as

$$\dot{\mathbf{V}}_1 = \mathbf{s}^T \mathbf{B}_\alpha \dot{\mathbf{s}} - \mathbf{s}^T \mathbf{B}_\alpha \mathbf{s} + \mathbf{s}^T \left[\left(\frac{\partial \mathbf{B}_9}{\partial \bar{\mathbf{x}}} \dot{\bar{\mathbf{x}}} \right) \mathbf{s} + \int_0^1 \mathbf{B}_\alpha \mathbf{v} d\vartheta \right]. \quad (23)$$

Considering (12), we rewritten (23) as

$$\begin{aligned} \dot{\mathbf{V}}_1 &= \mathbf{s}^T \mathbf{B}_\alpha \mathbf{B}_d^{-1}(\mathbf{x}) [\mathcal{F}(\mathbf{x}) + \tau + \mathcal{R} + p] \\ &\quad + \mathbf{s}^T \left[\left(\frac{\partial \mathbf{B}_9}{\partial \bar{\mathbf{x}}} \dot{\bar{\mathbf{x}}} \right) \mathbf{s} + \int_0^1 \mathbf{B}_\alpha \mathbf{v} d\vartheta \right]. \end{aligned} \quad (24)$$

Since \mathbf{B}_d , α , and $\mathbf{B}_d \alpha$ are symmetric, we have

$$\mathbf{B}_\alpha \mathbf{B}_d^{-1}(\mathbf{x}) = \mathbf{B}_d(\mathbf{x}) \alpha \mathbf{B}_d^{-1}(\mathbf{x}) = \alpha, \quad (25)$$

and rewrite (24) as

$$\dot{\mathbf{V}}_1 = \mathbf{s}^T \alpha [\mathcal{F}(\mathbf{x}) + \tau + \mathcal{R} + p] + \mathbf{s}^T \left[\left(\frac{\partial \mathbf{B}_9}{\partial \bar{\mathbf{x}}} \dot{\bar{\mathbf{x}}} \right) \mathbf{s} + \int_0^1 \mathbf{B}_\alpha \mathbf{v} d\vartheta \right]. \quad (26)$$

Considering (14) and $\mathbf{B}_\alpha = \mathbf{B}_d \alpha$,

$$\begin{aligned} \dot{\mathbf{V}}_1 &= \mathbf{s}^T \alpha [\mathcal{F}(\mathbf{x}) + \tau + \mathcal{R} + p] \\ &\quad + \mathbf{s}^T \left[\int_0^1 \vartheta \left(\frac{\partial \mathbf{B}_\alpha}{\partial \bar{\mathbf{x}}} \dot{\bar{\mathbf{x}}} \right) \mathbf{s} d\vartheta + \int_0^1 \mathbf{B}_\alpha \mathbf{v} d\vartheta \right] \\ &= \mathbf{s}^T \alpha [\mathcal{F}(\mathbf{x}) + \Phi + \tau + \mathcal{R} + p]. \end{aligned} \quad (27)$$

where

$$\Phi = \int_0^1 \vartheta \left(\frac{\partial \mathbf{B}_d}{\partial \bar{\mathbf{x}}} \dot{\bar{\mathbf{x}}} \right) \mathbf{s} d\vartheta + \int_0^1 \mathbf{B}_d \mathbf{v} d\vartheta, \quad (28)$$

$$\frac{\partial \mathbf{B}_d}{\partial \bar{\mathbf{x}}} \dot{\bar{\mathbf{x}}} = \text{diag} \left[\sum_{j=1}^m \frac{\partial b_{dii}}{\partial \bar{\mathbf{x}}_j} \dot{\bar{\mathbf{x}}}_j \right], \quad i = 1, \dots, m.$$

2.6. Disturbance-Observer-Based Fuzzy Controller. As the accurate and complete dynamics of a robot manipulator is difficult to obtain, especially in complex environments with unknown external disturbance, a fuzzy logic system is adopted for the estimation of unknown functions in the robot manipulator. Generally, the fuzzy rules are given as

$$\Lambda_i: \text{if } Z_1 \text{ is } K_1^i \text{ and, } \dots, \text{ and } Z_l \text{ is } K_l^i, \text{ then } h \text{ is } h_i, \quad (29)$$

where h_i represents i_{th} fuzzy rule.

$$h(Z) = \frac{\sum_{l=1}^m h^l \left(\prod_{j=1}^n \varphi(Z_j) \right)}{\sum_{l=1}^m \left(\prod_{j=1}^n \varphi(Z_j) \right)} = \Theta^T S(Z), \quad (30)$$

where $Z = [Z_1, Z_2, \dots, Z_m] \in R^n$ is the input variable; $S_i = ((\prod_{j=1}^n \varphi(Z_j)) / (\sum_{l=1}^m (\prod_{j=1}^n \varphi(Z_j))))$ are the fuzzy basis functions; m is the number of fuzzy rules; and W denotes the adaptable weight parameters.

In this study, the uncertain term $\mathcal{F}(\mathbf{x})$ is estimated as

$$\mathcal{F}(\mathbf{x}) = -\Theta^{*T} S(Z) - \varsigma, \quad (31)$$

where $Z = [\mathbf{x}_1^T, \mathbf{x}_2^T]^T$ and ς represents the approximate error and $\|\varsigma\| \leq \varsigma^*$ ($\varsigma^* > 0$).

Then, we rewritten (9) as

$$\mathbf{B}_d(\mathbf{x})\dot{x}_2 = -\Theta^{*T} S(Z) - \varsigma + \mathcal{R} + \tau + p. \quad (32)$$

To eliminate the effect of the external disturbance and unknown terms in the system, a disturbance observer is utilized in the controller design. We define $D = \mathcal{R} + p - \varsigma$, and (32) can be represented as

$$\mathbf{B}_d(\mathbf{x})\dot{x}_2 = -\Theta^{*T} S(Z) + \tau + D. \quad (33)$$

As in the robotic system, the actuator and input saturation exist [29]; the motor torque is normally assumed to be constrained with saturation in practical. Therefore, $p = -\Delta_{\mathbf{B}} \mathbf{B}^{-1}(\mathbf{x})\tau$ is bounded. In this study, the \mathbf{r} in this robot manipulator system is assumed as bounded with $\|\mathbf{r}\| \leq d_c$, where d_c is a positive constant. As the derivative of a differential continuous function with bounded constraints is also bounded [30], we can obtain $\|\dot{D}\| \leq \varrho$, where $\varrho > 0$.

For the disturbance observer design, an auxiliary variable Ω is defined as $\Omega = D - \Psi \mathbf{x}_2$ with $\Psi = \Psi^T > 0$. With (33), we can obtain

$$\begin{aligned} \dot{\Omega} &= \dot{D} - \Psi \dot{\mathbf{x}}_2 \\ &= \dot{D} - \Psi \mathbf{B}_d^{-1}(\mathbf{x}) \left[-\Theta^{*T} S(Z) + \tau + D \right]. \end{aligned} \quad (34)$$

Then, the estimate of $\dot{\Omega}$ is calculated as

$$\hat{\dot{\Omega}} = -\Psi \mathbf{B}_d^{-1}(\mathbf{x}) \left[\tau + \hat{D} - \hat{\Theta}^T S(Z) \right], \quad (35)$$

where $\hat{D} = \hat{\Omega} + \Psi \mathbf{x}_2$ and is the estimate of D . Then, we define $\tilde{D} = D - \hat{D}$ as the disturbance estimate error, and we have

$$\tilde{\Omega} = \Omega - \hat{\Omega} = D - \hat{D} = \tilde{D}. \quad (36)$$

Then, the derivative of \tilde{D} is computed as

$$\begin{aligned} \dot{\tilde{D}} &= \dot{\hat{\Omega}} = \dot{\Omega} - \dot{\hat{\Omega}} \\ &= \dot{D} - \Psi \mathbf{B}_d^{-1}(\mathbf{x}) \left[\hat{D} + \hat{\Theta}^T S(Z) \right], \end{aligned} \quad (37)$$

with $\tilde{\Theta} = \hat{\Theta} - \Theta^*$.

Based on the disturbance observer [31] and fuzzy logic rules, the control law of this robot manipulator is developed as

$$\tau = \hat{\Theta}^T S(Z) - \Phi - K_1 \alpha \mathbf{s} - \hat{D}. \quad (38)$$

The updated law of the fuzzy controller is developed as

$$\dot{\hat{\Theta}}_i = -\Gamma_i \left[S_i(Z) \alpha_{ii} \mathbf{s}_i + \varepsilon \hat{\Theta}_i \right], \quad (39)$$

where Γ_i and ε are positive constants.

Theorem 1. *According to the designed controller in (38) and the disturbance-observer-based fuzzy controller, all closed-loop signals are uniformly bounded in this complex environment with unknown disturbances and model uncertainties (proof is shown in Appendix A).*

3. Simulation

The EMG-driven NMS model was implemented in OpenSim 3.3 through the calibrated EMG-informed NMS modelling toolbox (CEINMS) [21]. OpenSim was also used to calculate musculotendon lengths and moment arms using joint angles through the scaled musculoskeletal model [32]. CEINMS was then employed to calibrate a subject-specific EMG-driven NMS model to predict joint torques. A generic reaching task was simulated with a musculoskeletal arm model (Figure 2), and the data were acquired from a publicly available database (https://simtk.org/frs/index.php?group_id=657) [33]. The input data include EMG signals and joint kinematics of right arm shoulder and elbow. The EMG signals of 6 muscles of the right arm were simulated through static optimization tool (resolving the net joint moments into individual muscle forces by minimizing the sum of squared muscle activations) in OpenSim, including triceps long, triceps lateral, triceps medial, biceps long, biceps short, and brachialis.

Before using the EMG-driven NMS model to predict joint torque, a calibration process was applied to obtain a subject-specific EMG-driven NMS model with individual's muscle-tendon properties. To perform the calibration, inverse dynamics was used to calculate the joint moment as measured joint moments. Inverse dynamics calculated the forces and torques of joints by solving dynamic equation of motion of the user as shown in the following equation:

$$M(q)\ddot{q} + C(q, \dot{q}) + G(q) = \tau, \quad (40)$$

where q, \dot{q}, \ddot{q} are the position, velocity, and acceleration of the generalised coordinates; $M(q)$ is the mass matrix; $C(q, \dot{q})$ is the centripetal and Coriolis forces matrix; $G(q)$ is the gravitational forces matrix; and τ is the vector of unknown generalised forces.



FIGURE 2: The reaching task illustration of the manipulator.

During the calibration of the subject-specific EMG-driven NMS model, optimal fibre length l_0^m and tendon slack length l_s^t of each MTU were bounded within $\pm 15\%$ from their initial values, and muscle activation dynamics parameters A, C_1, C_2 were calibrated globally. The shape factor A was bounded between -3 and 0 and coefficients C_1, C_2 were bounded between -1 and 1 . A strength coefficient constrained between 0.5 and 2.5 was assigned to each MTU and was used to calibrate maximum isometric force. During the calibration, these subject-specific parameters were refined by an optimization algorithm to minimize the error between estimated and measured/actual ankle joint torques [21].

Correlation analysis of joint torque estimated via inverse dynamics and EMG-driven NMS model was carried out across all trials using MATLAB (MatlabR2018a, MathWorks Inc., Natick, MA, USA). A significance level of 0.05 was set for all statistical tests.

For the robot manipulator, our aim was to design a controller that could complete tasks with human-like characteristics as to follow the desired velocity. In the system, we consider 2-DOFs shoulder and elbow manipulator, and the dynamic modelling of the shoulder and elbow manipulator is modeled as

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + f_{\text{dis}} = \tau, \quad (41)$$

where τ is the joint torque and $q = [q_1, q_2]^T$. We model $M(q)$, $C(q, \dot{q})$, $G(q)$, and f_{dis} as

$$\begin{aligned} M(q) &= \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}, \\ C(q, \dot{q}) &= \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}, \\ G(q) &= \begin{bmatrix} G_1 \\ G_2 \end{bmatrix}, \\ f_{\text{dis}} &= \begin{bmatrix} D_1 \\ D_2 \end{bmatrix}, \end{aligned} \quad (42)$$

where $M_{11} = m_1 l_{c1}^2 + m_2 (l_{c1}^2 + l_{c2}^2 + 2l_1 l_{c2} \cos q_2) + I_1 + I_2$, $M_{12} = m_2 (l_{c2}^2 + l_1 l_{c2} \cos q_2) + I_2$, $M_{21} = m_2 (l_{c2}^2 + l_1 l_{c2} \cos q_2) + I_2$, $M_{22} = m_2 l_{c2}^2 + I_2$; $C_{11} = -m_2 l_1 l_{c2} \dot{q}_2 \sin q_2$, $C_{12} = -m_2 l_1 l_{c2} (\dot{q}_1 + \dot{q}_2) \sin q_2$, $C_{21} = m_2 l_1 l_{c2} \dot{q}_1 \sin q_2$, $C_{22} = 0$; $G_1 = (m_1 l_{c2} + m_2 l_1) g \cos q_1 + m_2 l_{c2} g \cos (q_1 + q_2)$, $G_2 = m_2 l_{c2} g \cos (q_1 + q_2)$; $D_1 = 0.21 \cos q_2^2 + 0.04 \sin (0.3q_2 t)$; and $D_2 = 0.12 \sin q_1^2 + 0.03 \sin (0.2q_1 t)$.

The robotic system parameters are modeled as $m_1 = 2$ kg, $m_2 = 1.5$ kg, $l_1 = 0.35$ m, $l_2 = 0.2$ m, $g = 9.81$ m/s², $I_1 = (1/4)m_1 l_1^2$, and $I_2 = (1/2)m_2 l_2^2$, and l_c is the center of mass with $l_c = (l/2)$.

We transform (41) into the model we used in (7), which is

$$\begin{cases} \mathbf{y} = \mathbf{x}_1, \\ \dot{\mathbf{x}}_2 = \mathbf{B}^{-1}(\mathbf{x})[\mathbf{U}(\mathbf{x}) + \mathbf{r} + \boldsymbol{\tau}], \\ \dot{\mathbf{x}}_1 = \mathbf{x}_2, \end{cases} \quad (43)$$

where $\mathbf{x}_1 = [q_1, q_2]^T$, $\mathbf{x}_2 = [\dot{q}_1, \dot{q}_2]^T$, $\mathbf{B}(x) = M(q)$, $\mathbf{U}(\mathbf{x}) = -C(q, \dot{q})\dot{q}_2 - G(q)$, $\mathbf{r} = -f_{\text{dis}}$, and $\boldsymbol{\tau}$ is the joint torque.

Let $\hat{\theta}_i(0) = [0, \dots, 0]^T$ be the initial values of the adaptive law (39). The design parameters' values were set to $\alpha = I_{2 \times 2}$, $B_d(x) = \text{diag}[\cos x_1(2) + 2, 1 + 0.5 \sin x_1(1)]$, $\lambda_{11} = \lambda_{21} = 24$, $\Gamma_1 = \Gamma_2 = \Gamma_{\tau 1} = \Gamma_{\tau 2} = 1$, and $\sigma = \sigma_\tau = 9$.

The joint torque (normalized by the arm mass) estimated via inverse dynamics and EMG-driven NMS model is shown in Figures 3 and 4. The results of the correlation analysis showed that there was a significant correlation of joint torque estimated via inverse dynamics and EMG-driven NMS model ($p < 0.01$). The Pearson coefficients $r_1 = 0.991$ and $r_2 = 0.507$ were observed at the shoulder and elbow flexion/extension DOFs in the arm reaching movement, respectively. The root mean square error (RMSE) of shoulder flexion/extension torque between inverse dynamics and EMG-driven NMS model was 0.048 Nm/kg. Similar RMSE 0.046 Nm/kg was observed at the elbow flexion/extension DOFs in the arm reaching movement. Figures 5 and 6 demonstrated the desired and actual joint angular velocity of the robot manipulator in the reaching task, and the results validated the effectiveness of the proposed controller which could complete the task with users-like adaptivity and had a good performance.

4. Discussion

We developed a disturbance-observer-based fuzzy control framework that fully incorporates a robot manipulator system with an EMG-driven NMS model and could be applied in the controlling of the movement of robot manipulators with human-like characteristics. In the developed framework, a reaching task for a robot manipulator was studied to test performance of the disturbance-observer-based fuzzy controller. We found that the user's intention was well-predicted by the EMG-driven NMS model and then transmitted to the desired velocities of the robot manipulator system, which allows the robot manipulator system behave with human-like adaptivity. In the robot manipulator system with external disturbances and model uncertainties, the proposed disturbance-observer-based fuzzy control was also able to provide a good performance of motion tracking. In the current paper, the control framework we proposed in general is useful for obtaining the human-like characteristics, as well as simulating the control strategy in the robot manipulator system, and thus be of

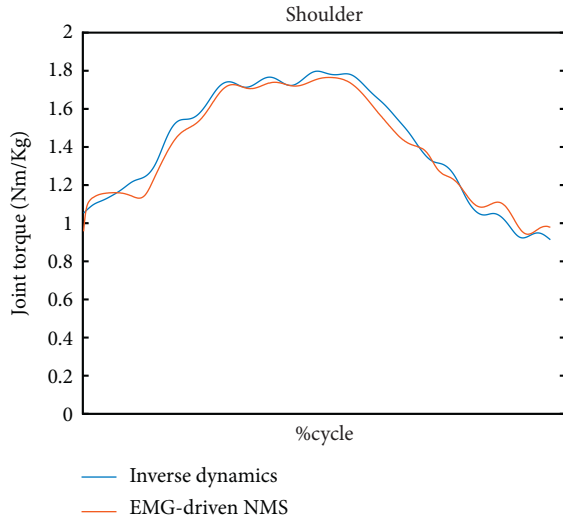


FIGURE 3: Shoulder joint torque (normalized by body mass) estimated via inverse dynamics and EMG-driven NMS model.

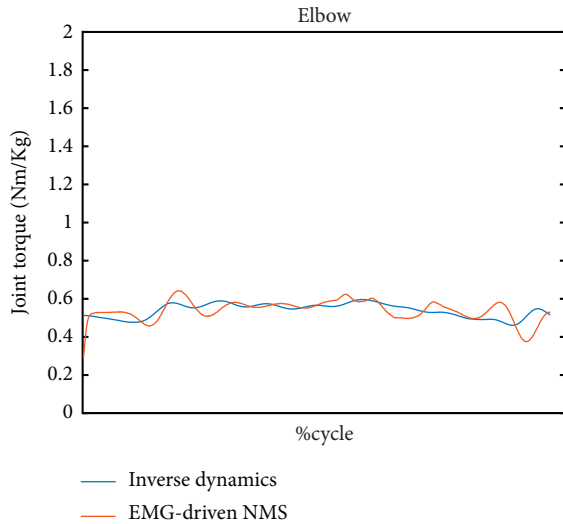


FIGURE 4: Elbow joint torque (normalized by body mass) estimated via inverse dynamics and EMG-driven NMS model.

large benefit for teleoperation robot manipulator applications.

EMG-driven NMS model is a popular method in users' motion intention (joint torque) estimation [34–36]. The EMG-driven NMS model uses mathematical equations to reproduce the transformations from EMG signal generation and joint angles to musculotendon forces and joint torques. To better predict joint torque, subject-specified EMG-driven model was calibrated with personalized musculoskeletal geometry such as moment arms and muscle characteristics. During the calibration, these subject-specific parameters were refined by an optimization algorithm to minimize the error between estimated and measured/actual ankle joint torques. With the mathematical equations approximately reproducing the transformations and optimization algorithm included in the calibration process, inevitable error existed in the EMG-driven NMS model when mapping EMG

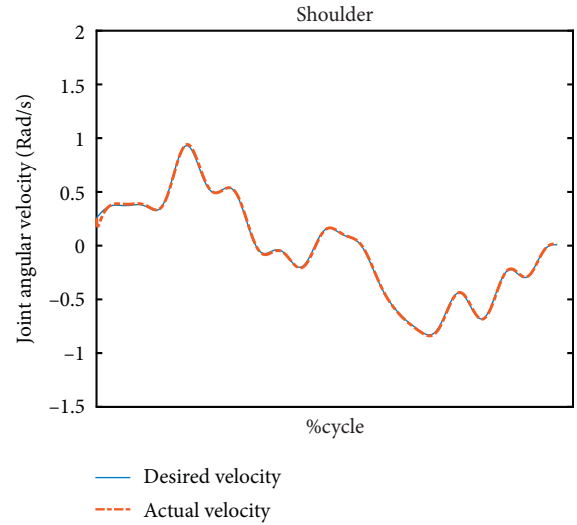


FIGURE 5: The desired and actual angular velocity of the shoulder.

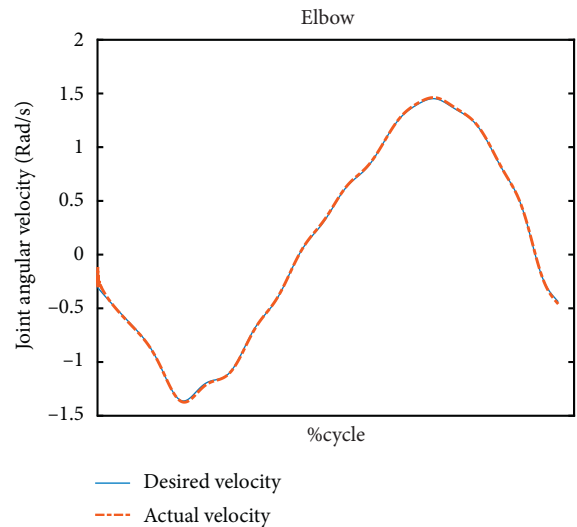


FIGURE 6: The desired and actual angular velocity of the elbow.

signals and angles into the joint torque. Despite this, the EMG-driven NMS model was able to estimate joint torque in close agreement to the reference data (with RMSE lower than 0.048 Nm/kg).

It is important to notice that unknown dynamics and external disturbance commonly exist in robotic manipulator systems. Fuzzy logic system [37], neural network [38], and disturbance observer [39] are commonly applied to solve these problems and maintain the system stability. Su et al. [40] integrated a fuzzy compensator into a teleoperation controller for a robot manipulator, and this controller was able to guarantee a safety-enhanced behavior in the null space. Li et al. [41] utilized a disturbance observer to compensate for the external disturbance in an admittance control for an upper robotic exoskeleton and guaranteed the robustness of the robotic arm. In the current study, a disturbance-observer-based controller with a high-dimensional integral-type Lyapunov function was developed for a robotic

manipulator with external disturbances and model uncertainties. The good tracking performance (Figures 5 and 6) demonstrate the effectiveness of the proposed control framework and the semiglobally uniformly ultimate boundedness of the closed-loop control system is also established.

One limitation of this study was that only reaching task at one speed was studied as an example in the proposed framework. Reaching task at different speeds as well as different situations and other daily activities such as grasp could be also tested in the proposed framework. Another limitation was that we only tested the control framework in a simulation environment with one subject data. Large sample size of subjects could be enrolled in the future study to see whether the proposed control framework would be able to generalize across subjects.

5. Conclusions

We developed an adaptive control framework that fully incorporates a robot manipulator system with an EMG-driven NMS model and use it to control the motion of the robot manipulator with human-like characteristics. In the developed framework, an example was studied to test performance of the disturbance-observer-based fuzzy controller, which was applied to a robot manipulator during a reaching task. We found that the EMG-driven NMS model was able to predict the user's intention (desired torque) and transmitted these human-like characteristics to the desired trajectories of the robot manipulator system. Moreover, external disturbances as well as model uncertainties were simulated in the robot manipulator system, and the proposed adaptive fuzzy controller integrated with a disturbance observer was able to provide a good performance of motion tracking. In the current paper, the control framework we proposed in general is useful for obtaining the human-like characteristics, as well as simulating the control strategy in the robot manipulator system that is subjected to realistic conditions, such as model uncertainties and external disturbances, and thus be of large benefit for teleoperation robot manipulator applications.

Appendix

Proof of the Proposed Lyapunov Function Stability

The Lyapunov function candidate is modeled as

$$\mathbf{V}_2 = \mathbf{V}_1 + \frac{1}{2}\tilde{D}^T\tilde{D} + \frac{1}{2}\sum_{i=1}^m\tilde{\Theta}_i^T\Gamma_i^{-1}\tilde{\Theta}_i. \quad (\text{A.1})$$

Based on (31), the derivative of \mathbf{V}_2 is calculated as

$$\begin{aligned} \dot{\mathbf{V}}_2 = & \mathbf{s}^T\alpha[-\Theta^*S(Z) - \zeta + \Phi + \tau + \mathcal{R} + p] \\ & + \tilde{D}^T\dot{\tilde{D}} + \sum_{i=1}^m\tilde{\Theta}_i^T\Gamma_i^{-1}\dot{\tilde{\Theta}}_i. \end{aligned} \quad (\text{A.2})$$

Considering (33), (38), and $D = \mathcal{R} + p - \zeta$, we have

$$\begin{aligned} \dot{\mathbf{V}}_2 = & \mathbf{s}^T\alpha[\tilde{\Theta}S(Z) + \tilde{D} - K_1\alpha\mathbf{s}] \\ & + \tilde{D}^T\dot{\tilde{D}} + \sum_{i=1}^m\tilde{\Theta}_i^T\Gamma_i^{-1}\dot{\tilde{\Theta}}_i. \end{aligned} \quad (\text{A.3})$$

Considering $\|\dot{\tilde{D}}\|$ is bounded, $\|S(Z)\| \leq \delta$, (37), (39), and the following terms

$$\begin{aligned} \mathbf{s}^T\alpha\tilde{D} & \leq \frac{\mathbf{s}^T\alpha\alpha\mathbf{s}}{2} + \frac{\tilde{D}^T\tilde{D}}{2}, \\ \tilde{D}^T\dot{\tilde{D}} & \leq \frac{\tilde{D}^T\tilde{D}}{2} + \frac{\|\dot{\tilde{D}}\|^2}{2}, \end{aligned} \quad (\text{A.4})$$

$$\sum_{i=1}^m\tilde{\Theta}_i^T S_i(Z)\mathbf{s}_i\alpha_{ii} = \mathbf{s}^T\alpha\tilde{\Theta}^T S(Z),$$

we have

$$\begin{aligned} \dot{\mathbf{V}}_2 \leq & -\mathbf{s}^T\alpha(K_1 - 0.5I_{m \times m})\alpha\mathbf{s} + \frac{\varrho^2}{2} \\ & - \tilde{D}^T(\Psi\mathbf{B}_d^{-1} - 2I_{m \times m})\tilde{D} + \frac{\varepsilon\|\Theta^*\|^2}{2}, \\ & - \frac{\varepsilon - \Psi\mathbf{B}_d^{-1}\delta}{2}\sum_{j=1}^m\tilde{\Theta}_j^T\tilde{\Theta}_j \end{aligned} \quad (\text{A.5})$$

where $\tilde{D}^T\Psi\mathbf{B}_d^{-1}\tilde{\Theta}^T S(Z) \leq ((\|\tilde{D}\|^2)/2) + ((\Psi\mathbf{B}_d^{-1}S(Z)\|\tilde{\Theta}\|^2)/2)$ and $-\varepsilon\tilde{\Theta}_i^T\tilde{\Theta}_i = -\varepsilon\|\tilde{\Theta}_i\|^2 - \varepsilon\tilde{\Theta}_i^T\tilde{\Theta}_i^* \leq -((\varepsilon\|\tilde{\Theta}_i\|^2)/2) + ((\varepsilon\|\tilde{\Theta}_i^*\|^2)/2)$.

If we choose K_1 , Ψ , and ε appropriately satisfying $\lambda_{\min}(\alpha(K_1 - 0.5I_{m \times m})\alpha) \geq \int_0^1 \vartheta \lambda_{\max}(\mathbf{B}_\alpha) d\vartheta$, $\Psi\mathbf{B}_d^{-1} - 2I_{m \times m} > 0$, and $\varepsilon - \Psi\mathbf{B}_d^{-1}\delta > 0$, we can obtain

$$\dot{\mathbf{V}}_2 \leq -\kappa\mathbf{V}_2 + C, \quad (\text{A.6})$$

where

$$\kappa = \min \left(\begin{array}{c} \lambda_{\min}(\Psi\mathbf{B}_d^{-1} - 2I_{m \times m}) \\ \frac{\varepsilon - \Psi\mathbf{B}_d^{-1}\delta}{\lambda_{\max}(\sum_{i=1}^m\Gamma_i^{-1})}, \quad 1 \end{array} \right), \quad (\text{A.7})$$

$$C = \frac{\varepsilon\|\Theta^*\|^2}{2} + \frac{\varrho^2}{2}.$$

By multiplying $e^{\kappa t}$ and integrating the both sides of inequality (A.6), we have

$$\mathbf{V}_2 \leq \left(\mathbf{V}_2(0) - \frac{C}{\kappa} \right) e^{-\kappa t} + \frac{C}{\kappa} \leq \mathbf{V}_2(0) + \frac{C}{\kappa}. \quad (\text{A.8})$$

Based on (A.8), \mathbf{V}_2 is ultimately bounded as $t \rightarrow \infty$; \tilde{D} , \mathbf{s} , and $\tilde{\Theta}$ are also bounded.

Data Availability

The data supporting this study is from a previously reported study, which has been cited at relevant place within the text as reference [33]. The dataset are available at (https://simtk.org/frs/index.php?group_id=657).

Conflicts of Interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflicts of interest.

References

- [1] H. Su, Y. Hu, H. R. Karimi, A. Knoll, G. Ferrigno, and E. De Momi, "Improved recurrent neural network-based manipulator control with remote center of motion constraints: experimental results," *Neural Networks*, vol. 131, pp. 291–299, 2020.
- [2] C. Yang, G. Peng, Y. Li, R. Cui, L. Cheng, and Z. Li, "Neural networks enhanced adaptive admittance control of optimized robot–environment interaction," *IEEE Transactions on Cybernetics*, vol. 49, no. 7, pp. 2568–2579, 2018.
- [3] P. K. Artemiadis and K. J. Kyriakopoulos, "Teleoperation of a robot manipulator using EMG signals and a position tracker," in *Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1003–1008, IEEE, Edmonton, Canada, August 2005.
- [4] H. Huang, T. Zhang, C. Yang, and C. P. Chen, "Motor learning and generalization using broad learning adaptive neural control," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 10, pp. 8608–8617, 2019.
- [5] W. Qi, H. Su, and A. Aliverti, "A smartphone-based adaptive recognition and real-time monitoring system for human activities," *IEEE Transactions on Human-Machine Systems*, vol. 50, no. 5, pp. 414–423, 2020.
- [6] Z. Li, Y. Yuan, L. Luo et al., "Hybrid brain/muscle signals powered wearable walking exoskeleton enhancing motor ability in climbing stairs activity," *IEEE Transactions on Medical Robotics and Bionics*, vol. 1, no. 4, pp. 218–227, 2019.
- [7] W. Ryu, B. Han, and J. Kim, "Continuous position control of 1 dof manipulator using EMG signals," in *Proceedings of the 2008 Third International Conference on Convergence and Hybrid Information Technology*, vol. 1, pp. 870–874, IEEE, Busan, South Korea, November 2008.
- [8] N. Bu, M. Okamoto, and T. Tsuji, "A hybrid motion classification approach for EMG-based human–robot interfaces using Bayesian and neural networks," *IEEE Transactions on Robotics*, vol. 25, no. 3, pp. 502–511, 2009.
- [9] C. Yang, J. Luo, C. Liu, M. Li, and S.-L. Dai, "Haptics electromyography perception and learning enhanced intelligence for teleoperated robot," *IEEE Transactions on Automation Science and Engineering*, vol. 16, no. 4, pp. 1512–1521, 2019.
- [10] Y. Hu, X. Wu, P. Geng, and Z. Li, "Evolution strategies learning with variable impedance control for grasping under uncertainty," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 10, pp. 7788–7799, 2018.
- [11] K. Yi, J. Han, X. Liang, and Y. He, "Contact transition control with acceleration feedback enhancement for a quadrotor," *ISA Transactions*, 2020.
- [12] G. Chen, H. Cao, J. Conrath, H. Tang, F. Rohrbein, and A. Knoll, "Event-based neuromorphic vision for autonomous driving: a paradigm shift for bio-inspired visual sensing and perception," *IEEE Signal Processing Magazine*, vol. 37, no. 4, pp. 34–49, 2020.
- [13] C. Yang, Y. Jiang, J. Na, Z. Li, L. Cheng, and C.-Y. Su, "Finite-time convergence adaptive fuzzy control for dual-arm robot with unknown kinematics and dynamics," *IEEE Transactions on Fuzzy Systems*, vol. 27, no. 3, pp. 574–588, 2019.
- [14] H. Su, W. Qi, C. Yang, J. Sandoval, G. Ferrigno, and E. D. Momi, "Deep neural network approach in robot tool dynamics identification for bilateral teleoperation," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2943–2949, 2020.
- [15] H. Lin, T. Zhang, Z. Chen, H. Song, and C. Yang, "Adaptive fuzzy Gaussian mixture models for shape approximation in robot grasping," *International Journal of Fuzzy Systems*, vol. 21, no. 4, pp. 1026–1037, 2019.
- [16] W. He, Y. Sun, Z. Yan, C. Yang, Z. Li, and O. Kaynak, "Disturbance observer-based neural network control of cooperative multiple manipulators with input saturation," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 31, no. 5, pp. 1735–1746, 2019.
- [17] R. Yang, C. Yang, M. Chen, and J. Na, "Adaptive impedance control of robot manipulators based on Q-learning and disturbance observer," *Systems Science & Control Engineering*, vol. 5, no. 1, pp. 287–300, 2017.
- [18] S. Qiu, Z. Li, W. He, L. Zhang, C. Yang, and C.-Y. Su, "Brain–machine interface and visual compressive sensing-based teleoperation control of an exoskeleton robot," *IEEE Transactions on Fuzzy Systems*, vol. 25, no. 1, pp. 58–69, 2016.
- [19] H. Su, C. Yang, G. Ferrigno, and E. De Momi, "Improved human-robot collaborative control of redundant robot for teleoperated minimally invasive surgery," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 1447–1453, 2019.
- [20] H. Su, W. Qi, Y. Hu, H. R. Karimi, G. Ferrigno, and E. De Momi, "An incremental learning framework for human-like redundancy optimization of anthropomorphic manipulators," *IEEE Transactions on Industrial Informatics*, 2021.
- [21] C. Pizzolato, D. G. Lloyd, M. Sartori et al., "CEINMS: a toolbox to investigate the influence of different neural control solutions on the prediction of muscle excitation and joint moments during dynamic motor tasks," *Journal of Biomechanics*, vol. 48, no. 14, pp. 3929–3936, 2015.
- [22] M. Sartori, M. Reggiani, D. G. Lloyd, and E. Pagello, "A neuromusculoskeletal model of the human lower limb: towards EMG-driven actuation of multiple joints in powered orthoses," in *Proceedings of the 2011 IEEE International Conference on Rehabilitation Robotics*, pp. 1–6, IEEE, Zurich, Switzerland, July 2011.
- [23] D. G. Lloyd and T. F. Besier, "An EMG-driven musculoskeletal model to estimate muscle forces and knee joint moments in vivo," *Journal of Biomechanics*, vol. 36, no. 6, pp. 765–776, 2003.
- [24] T. S. Buchanan, D. G. Lloyd, K. Manal, and T. F. Besier, "Neuromusculoskeletal modeling: estimation of muscle forces and joint moments and movements from measurements of neural command," *Journal of Applied Biomechanics*, vol. 20, no. 4, pp. 367–395, 2004.
- [25] H. X. Hoang, C. Pizzolato, L. E. Diamond, and D. G. Lloyd, "Subject-specific calibration of neuromuscular parameters enables neuromusculoskeletal models to estimate physiologically plausible hip joint contact forces in healthy adults," *Journal of Biomechanics*, vol. 80, pp. 111–120, 2018.

- [26] N. Lotti, M. Xiloyannis, G. Durandau et al., "Adaptive model-based myoelectric control for a soft wearable arm exosuit: a new generation of wearable robot control," *IEEE Robotics & Automation Magazine*, vol. 27, no. 1, pp. 43–53, 2020.
- [27] Q. Wu, X. Wang, B. Chen, and H. Wu, "Patient-active control of a powered exoskeleton targeting upper limb rehabilitation training," *Frontiers in Neurology*, vol. 9, p. 817, 2018.
- [28] Z. Li, Z. Huang, W. He, and C.-Y. Su, "Adaptive impedance control for an upper limb robotic exoskeleton using biological signals," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 2, pp. 1664–1674, 2016.
- [29] W. Sun, Z. Zhao, and H. Gao, "Saturated adaptive robust control for active suspension systems," *IEEE Transactions on Industrial Electronics*, vol. 60, no. 9, pp. 3889–3896, 2012.
- [30] L. Zhang, Z. Li, and C. Yang, "Adaptive neural network based variable stiffness control of uncertain robotic systems using disturbance observer," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 3, pp. 2236–2245, 2016.
- [31] D. Huang, C. Yang, Z. Ju, and S.-L. Dai, "Disturbance observer enhanced variable gain controller for robot teleoperation with motion capture using wearable armbands," *Autonomous Robots*, vol. 44, no. 7, pp. 1217–1231, 2020.
- [32] E. Ceseracciu, A. Mantoan, M. Bassa et al., "A flexible architecture to enhance wearable robots: integration of EMG-informed models," in *Proceedings of the 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4368–4374, IEEE, Hamburg, Germany, October 2015.
- [33] K. R. Saul, X. Hu, C. M. Goehler et al., "Benchmarking of dynamic simulation predictions in two software platforms using an upper limb musculoskeletal model," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 18, no. 13, pp. 1445–1458, 2015.
- [34] G. Durandau, D. Farina, and M. Sartori, "Robust real-time musculoskeletal modeling driven by electromyograms," *IEEE Transactions on Biomedical Engineering*, vol. 65, no. 3, pp. 556–564, 2017.
- [35] S. Yao, Y. Zhuang, Z. Li, and R. Song, "Adaptive admittance control for an ankle exoskeleton using an EMG-driven musculoskeletal model," *Frontiers in Neurobotics*, vol. 12, p. 16, 2018.
- [36] Y. Hu, Z. Li, G. Li, P. Yuan, C. Yang, and R. Song, "Development of sensory-motor fusion-based manipulation and grasping control for a robotic hand-eye system," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 47, no. 7, pp. 1169–1180, 2016.
- [37] H. Huang, C. Yang, and C. P. Chen, "Optimal robot-environment interaction under broad fuzzy neural adaptive control," *IEEE Transactions on Cybernetics*, pp. 1–12, 2020.
- [38] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, "Robot learning system based on adaptive neural control and dynamic movement primitives," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 777–787, 2019.
- [39] W. He, Z. Yan, C. Sun, and Y. Chen, "Adaptive neural network control of a flapping wing micro aerial vehicle with disturbance observer," *IEEE Transactions on Cybernetics*, vol. 47, no. 10, pp. 3452–3465, 2017.
- [40] H. Su, J. Sandoval, P. Vieyres, G. Poisson, G. Ferrigno, and E. De Momi, "Safety-enhanced collaborative framework for tele-operated minimally invasive surgery using a 7-dof torque-controlled robot," *International Journal of Control, Automation and Systems*, vol. 16, no. 6, pp. 2915–2923, 2018.
- [41] Z. Li, B. Huang, Z. Ye, M. Deng, and C. Yang, "Physical human-robot interaction of a robotic exoskeleton by admittance control," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 12, pp. 9614–9624, 2018.

Research Article

Load Parameter Identification for Parallel Robot Manipulator Based on Extended Kalman Filter

Shijie Song, Xiaolin Dai, Zhangchao Huang, and Dawei Gong 

School of Mechanical and Electrical Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

Correspondence should be addressed to Dawei Gong; pzhzhx@126.com

Received 12 September 2020; Revised 22 October 2020; Accepted 8 November 2020; Published 27 November 2020

Academic Editor: Ning Wang

Copyright © 2020 Shijie Song et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Load is the main external disturbance of a parallel robot manipulator. This disturbance will cause dynamic coupling among different degrees of freedom and make heaps of model-based control methods difficult to apply. In order to compensate this disturbance, it is crucial to obtain an accurate dynamic model of load. However, in practice, the load is always uncertain and its dynamic parameters are arduous to know a priori. To cope with this problem, this paper proposes a novel and simple approach to identify the dynamic parameters of load. Firstly, the dynamic model of the parallel robot manipulator with uncertain load is established and the dynamic coupling caused by load is also analyzed. Then, according to the dynamic model, the excitation signal is designed and a weak nonlinear dynamic model is derived. Furthermore, the identification model is presented and the identification algorithm based on the extended Kalman filter is designed. Lastly, numerical simulation results, obtained using a six-degree-of-freedom Gough–Stewart parallel manipulator, demonstrate the good estimation performance of the proposed method.

1. Introduction

Parallel robot manipulator (PRM) entails the advantages of higher precision, faster response, higher rigidity, and stronger carrying capacity over serial robots and, hence, is widely applied in many fields of industry [1, 2]. Currently, the standard industrial control technique applied in PRMs is PID control, which neglected the complex dynamic characteristics of robots, so that it can hardly meet the requirement of fast and accurate motion [3, 4]. During the past decade, several model-based control methods, such as computed torque control [5], dynamic feedforward control [6], and modal control [7, 8], are used in controlling PRMs because of the excellent control accuracy and dynamic performance of these methods. However, these methods are all based on the dynamic model to design, which makes the control performance of these methods strongly depend on the accurate knowledge of the dynamic model.

In practical applications, PRMs always have to hold an uncertain load to work, and the dynamic parameters of load

are hard to know a priori and cannot be measured directly, which makes several model-based control schemes unable to be applied [9]. Take modal control as an example, the key of the control strategy is the modal conversion matrix. The calculation of modal conversion needs the system mass matrix, but the uncertainty of the load will make the accurate system mass matrix impossible to obtain [10, 11]. Otherwise, the uncertain load, which is a main external disturbance of the system [12], has a significant impact on system dynamic performance; it will cause dynamic coupling among different degrees of freedom (DOFs) [13, 14]. Thus, in order to achieve the high-performance control of the system, the uncertain load disturbance is necessary to be estimated and compensated.

In parameter identification, three kinds of methods have been proposed to estimate the robot dynamic parameters. The first method is using computer-aided design (CAD) techniques to obtain the dynamic parameters of robots. The 3-dimensional (3D) model of robots generally provided by the robot manufacturer and the parameters, such as the

inertia tensor and centroid position, can be solved by any CAD software based on these models. However, the parameters obtained from the CAD techniques are not identical to the real robot because of the manufacturing and assembling error. In addition, for the PRMs with uncertain load, it is difficult to build a 3D model for load, which is usually not provided by the manufacturer and is made up of many complex parts. To avoid this problem, the method of the physical experiment is used to perform the determination of the dynamic parameter. This method can better estimate the mass, centroid position, and inertia tensor of the part. However, the necessity of disassembling the robot and measuring by special devices limits the application of this method. Moreover, this method is inappropriate for the measurement of large-sized parts, such as aircraft and submarine, which are common types of loads and are generally too big to be measured and too complex to be disassembled. The last method is the theoretical identification method, which can obtain better identification result, and does not require disassembly of robots and special devices for measurement compared to the above two methods. Until now, many identification algorithms have been presented to estimate the parameters of robots [15], such as the least squares method [16, 17], Kalman filtering method [18–20], maximum-likelihood method [21, 22]. In addition, among these identification algorithms, the most used method is the least squares method [23]. However, most of the research objects of these methods are serial manipulators, while for parallel manipulators, there is little research at present.

Compared to the serial manipulator, the research on parameter identification of PRMs started late, and too little work has been devoted to the load parameters identification. Chen derived the estimation equation in a linear form of identified parameters based on a new structured Boltzmann–Hamel–d’Alembert approach and used the least square method to identify the parameters [24]. Tian proposed an inertial parameter identification method based on sinusoidal vibrations of a six-degree-of-freedom parallel manipulator and used the least square method to identify the parameters [25]. Briot and Gautier used total least squares to identify the parallel robot dynamic parameters [26]. Wu et al. investigated the dynamic parameter identification of a redundantly actuated parallel manipulator and proposed a two-step identification approach based on the least squares method to identify the dynamic parameters of the system [27]. Thanh used the direct pattern search technique to do the dynamics identification for a redundant 3-(P) RRR manipulator [28].

Normally, the methods to identify the parameters of the PRM adopt the methods of the least squares method and the weighted least squares method. However, the least square method is not suitable for the identification of parallel manipulators. The main reasons are as follows:

- (1) The LMS needs to establish an inverse dynamic model that is linear with respect to the dynamic parameters. However, due to the complex and coupled dynamics of the parallel manipulators, it is

still not easy to rewrite the dynamic equation into a linear form that is suitable for using the parameter identification algorithm [29].

- (2) The LSM is sensitive to measurement noise. However, parallel manipulators are generally driven by hydraulic pressure, and the noise of driving force, displacement, speed, and acceleration is relatively large, which seriously limits the identification accuracy and convergence speed of the method [30, 31].
- (3) For the LSM, the observation matrix in the inverse dynamic model requires the value of displacement, speed, and acceleration in the task space. Since the displacement and speed of each leg of the parallel mechanism are relatively easy to measure, the displacement in the task space can be obtained by the positive kinematics solution, and the speed in the task space can be solved by the speed Jacobian matrix. But, for acceleration, it is not easy to be directly measured.

This paper proposed a method to identify the load of a parallel robot manipulator. Compared with the traditional least square method, the proposed identification approach does not require linearization of the dynamic model and optimization of the excitation trajectory, which are two complex problems to solve. Under the specific signals, the dynamic equation derived by the Newton–Euler method can be simplified and the simplified dynamic equation has two advantages: (1) this equation is weakly nonlinear with respect to the dynamic parameters of load and (2) the parameters to be identified are independent of each other in the equation and there is no product form. According to the simplified model, we designed the excitation trajectory, which has a simple form and can excite the dynamic parameters of the load. Importantly, the EKF algorithm is applied to estimate the load parameters, which is not sensitive to measurement noise and without acceleration measurements in the process of identification.

The organization of this paper is as follows. In Section 2, the dynamic model of PRM with uncertain load is established and the influence of load disturbance for the system is also analyzed. Section 3 presents the design of excitation trajectory and the process of load parameters identification based on the EKF algorithm. Section 4 shows the numerical simulation for verifying the proposed method. Finally, the main conclusions of this paper are presented in Section 5.

2. Mathematical Modeling

The parallel manipulator studied in this paper is a six-degree-of-freedom Gough–Stewart parallel manipulator, as shown in Figure 1(a). This manipulator is mainly composed of two platforms and six driving legs. The lower platform fixed on the ground is named static platform and the upper platform used to carry loads is named moving platform. The driving leg is the actuator to drive the moving platform to realize translation and rotation. In order to facilitate the

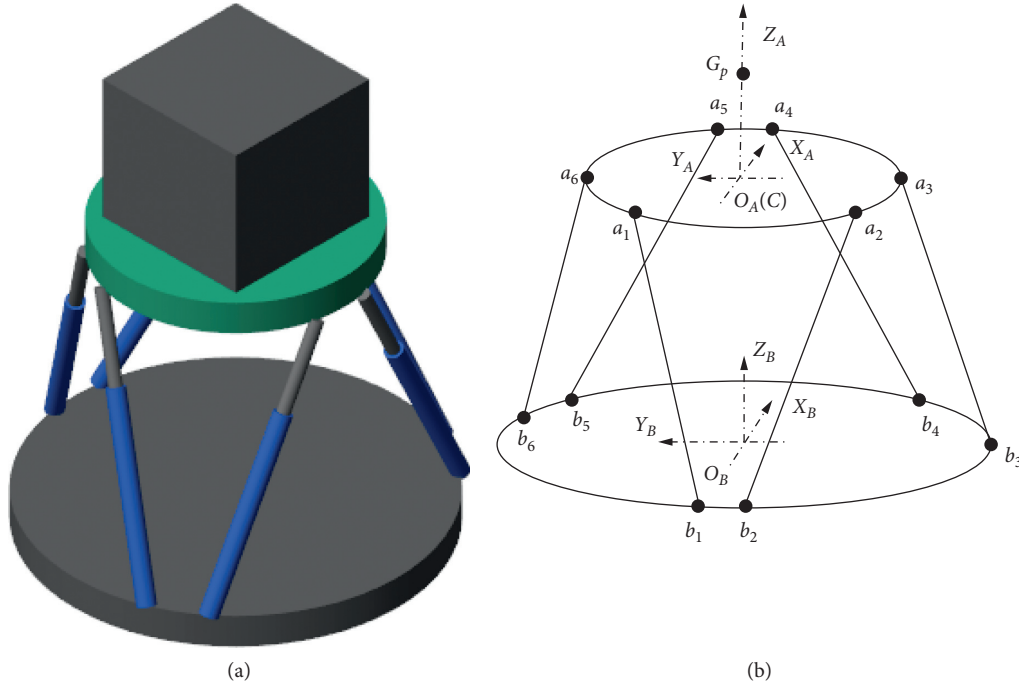


FIGURE 1: Schematic of the Stewart robot. (a) Simplified structure. (b) Mathematical representation.

analysis of the parallel manipulator, the coordinate system is established and shown in Figure 1(b). The static coordinate system $O_B - X_B Y_B Z_B$ is fixed on the static platform and the origin O_B is in the center of the lower platform. The moving coordinate $O_A - X_A Y_A Z_A$ is fixed on the upper platform and the origin O is in the center of the upper platform. Not only is O_A the centroid point of the upper platform, but also it is the control point C of the system. G_P is the centroid point of load. a_i and b_i are the upper and lower hinge points, respectively.

2.1. Kinematics Modeling. According to the space geometry theory, the length vector of the six legs can be expressed by

$$\mathbf{L} = [\mathbf{L}_1, \dots, \mathbf{L}_6] = \mathbf{R} \cdot \mathbf{A} + \mathbf{p} + \mathbf{p}_0 - \mathbf{B}, \quad (1)$$

where \mathbf{A} is the coordinate matrix of six lower hinges in the static coordinate. \mathbf{B} is the coordinate matrix of six upper hinges in the moving coordinate. \mathbf{R} denotes the transformation matrix from the static coordinate to the moving coordinate. $\mathbf{p} = [x \ y \ z]^T$ is a translation displacement vector. x , y and z are the displacement of platform mass center along the x -axis, y -axis, and z -axis, respectively. \mathbf{p}_0 is the initial height matrix composed of six initial height vectors, and each vector is $[0 \ 0 \ h_0]^T$. h_0 is the initial height when the upper platform motion table in the middle position.

By differentiating equation (1), one obtains

$$\dot{\mathbf{L}} = \frac{d\mathbf{L}}{dt} = \dot{\mathbf{p}} + \dot{\mathbf{R}} \cdot \mathbf{R}^T \cdot \mathbf{R} \cdot \mathbf{A} = \dot{\mathbf{p}} + \boldsymbol{\omega} \times \mathbf{R}^A \mathbf{A}. \quad (2)$$

The matrix form of equation (2) is

$$\dot{\mathbf{L}} = [\mathbf{L}_n^T (\mathbf{R} \cdot \mathbf{A} \times \mathbf{L}_n)^T] \cdot \dot{\mathbf{q}} = \mathbf{J}_v \dot{\mathbf{q}}, \quad (3)$$

where \mathbf{J}_v is velocity Jacobian matrix between the velocity of the upper platform and the leg velocity. \mathbf{L}_n is the unit direction vector matrix of the six legs' direction and $\mathbf{L}_n = [\mathbf{L}_1/|\mathbf{L}_1|, \dots, \mathbf{L}_6/|\mathbf{L}_6|]$. $\dot{\mathbf{q}} = \dot{\mathbf{q}} = [\dot{\mathbf{p}}, \boldsymbol{\omega}]^T$ is the upper platform velocity, in which $\boldsymbol{\omega} = [\omega_x \ \omega_y \ \omega_z]^T$ is the angular velocity of the moving platform and $\dot{\mathbf{L}}$ is the leg velocity. Importantly, matrix \mathbf{J} is the key to derive the dynamic equation of joint space.

In addition, kinematic analysis of the load is also required, such as centroid position and acceleration. Let the load eccentric position along three axes of the moving coordinate system be Δx , Δy , and Δz . The load eccentric position along three axes of the static coordinate can be expressed as

$$\begin{aligned} l_x &= c(q_5)c(q_6)\Delta x + [s(q_4)s(q_5)c(q_6)c(q_4)s(q_6)]\Delta y \\ &\quad + [c(q_4)s(q_5)c(q_6) + s(q_4)s(q_6)]\Delta z, \\ l_y &= c(q_5)s(q_6)\Delta x + [s(q_4)s(q_5)s(q_6) + c(q_4)c(q_6)]\Delta y \\ &\quad + [c(q_4)s(q_5)s(q_6) - s(q_4)c(q_6)]\Delta z, \\ l_z &= -s(q_5)\Delta x + s(q_4)c(q_5)\Delta y + c(q_4)c(q_5)\Delta z, \end{aligned} \quad (4)$$

where $c()$ and $s()$ are abbreviations for the trigonometric functions cosine and sine; q_4 , q_5 , and q_6 are the three Euler angles of the system; and l_x , l_y , and l_z are the load eccentric position along three main axes. From equation (4), it is easy to know that the l_x , l_y , and l_z are related to the Euler angles of the upper platform.

Let the load centroid acceleration in the static coordinate be $\ddot{p}_1 = [\ddot{x}_a \ddot{y}_a \ddot{z}_a]^T$. According to the rigid body dynamics theory, the relation between the load centroid acceleration and the upper platform centroid acceleration can be expressed by

$$\begin{aligned} \ddot{x}_a &= \ddot{x} + l_z \dot{\omega}_y - l_y \dot{\omega}_z, \ddot{y}_a = \ddot{y} + l_z \dot{\omega}_x - l_x \dot{\omega}_z, \ddot{z}_a \\ &= \ddot{z} + l_y \dot{\omega}_x - l_x \dot{\omega}_y. \end{aligned} \quad (5)$$

The relationship between \ddot{p}_1 and \ddot{p} is shown in Figure 2. Note that the $O_C - X_C Y_C Z_C$ coordinate system in Figure 1 is not the same as the $O_B - X_B Y_B Z_B$ coordinate system in Figure 2. The origin of the $O_C - X_C Y_C Z_C$ coordinate system is always coincident with the origin of the moving coordinate system, and the axes are always in the same direction as the axes in the static coordinate system. $O_C - X_C Y_C Z_C$ coordinate system is the coordinate system used to establish the dynamic equation. In addition, G_p is the position of the load center of mass.

2.2. Dynamic Modeling. First, we analyze the load dynamics. Based on Newton's second law and angular momentum theory, inertia force and moment of load can be described:

$$\mathbf{f}_L = m_l \ddot{p}_1 = m_l \ddot{p} - m_l \mathbf{Q} \dot{\omega}, \quad (6)$$

$$\mathbf{T}_L = {}^B I_L \dot{\omega} - m_l \mathbf{Q} \ddot{p}, \quad (7)$$

where m_l is the mass of load; ${}^B I_L$ is the inertia tensor of load in the static coordinate system; and \mathbf{Q} is the antisymmetric matrix composed of each element of the eccentric vector of the center of mass loaded in the static coordinate system and is as follows:

$$\mathbf{Q} = \begin{bmatrix} 0 & -l_z & l_y \\ l_z & 0 & -l_x \\ -l_y & l_x & 0 \end{bmatrix}. \quad (8)$$

Based on the Newton-Euler method, the dynamic model in task space can be obtained:

$$\sum_{i=1}^6 \mathbf{e}_i \cdot f_i = \mathbf{f}_L + m_p \ddot{p} + (m_p + m_l) \mathbf{g}, \quad (9)$$

$$\sum_{i=1}^6 (\mathbf{R} \cdot \mathbf{a}_i) \times (\mathbf{e}_i \cdot f_i) = \mathbf{T}_L + {}^B I_p \dot{\omega} + \omega \times ({}^B I_L + {}^B I_p) \omega, \quad (10)$$

where f_i is the force of hydraulic cylinder of i th leg; \mathbf{e}_i is the unit direction vector of the i th legs; \mathbf{f}_L is inertia force of load; \mathbf{T}_L is inertia moment of load; m_p is the mass of moving platform; and ${}^B I_p$ is the inertia tensor of moving platform in the static coordinate system. In addition, the third term on the right of equation (9) is the gravity of the system. The second term on the right of equation (9) represents the inertia force of the moving platform and the second term on the right of equation (10) represents the inertia force of the moving platform. Moreover, the third term on the right of equation (10) is Coriolis/centrifugal forces.

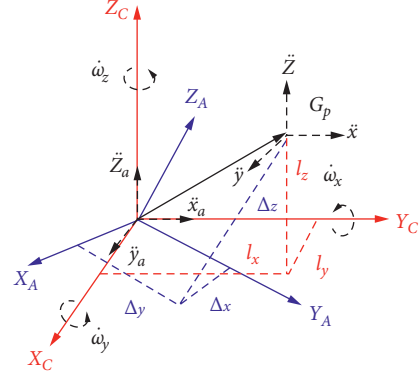


FIGURE 2: Kinematic relationship.

From equations (9) and (10), it can be obtained that

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{q}) \cdot \ddot{\mathbf{q}} + \mathbf{H}(\dot{\mathbf{q}}, \mathbf{q}) \cdot \dot{\mathbf{q}} + \mathbf{G}, \quad (11)$$

where $\boldsymbol{\tau}$ is the driving force vector in the task space; $\mathbf{H}(\dot{\mathbf{q}}, \mathbf{q})$ is the centrifugal/Coriolis term; \mathbf{G} is the gravity term; and $\mathbf{M}(\mathbf{q})$ is the mass of the system.

The relationship between the driving force vector in the task space and the driving force vector in the joint space can be expressed by

$$\boldsymbol{\tau} = \mathbf{J}^T \mathbf{f}. \quad (12)$$

From equations (11) and (12), the dynamic model of the robot in joint space can be rewritten as

$$\mathbf{f} = \mathbf{M}_L(\mathbf{l}) \cdot \ddot{\mathbf{l}} + \mathbf{H}_L(\mathbf{l}, \dot{\mathbf{l}}) \cdot \dot{\mathbf{l}} + \mathbf{G}_L, \quad (13)$$

where $\mathbf{M}_L(\mathbf{l})$ is the system mass matrix in joint space, $\mathbf{H}_L(\mathbf{l}, \dot{\mathbf{l}})$ is the centrifugal/Coriolis term in joint space, \mathbf{G}_L is the gravity term in joint space, and

$$\begin{aligned} \mathbf{M}_L(\mathbf{l}) &= \mathbf{J}^{-1T} \mathbf{M}(\mathbf{q}) \mathbf{J}^{-1}, \\ \mathbf{H}_L(\mathbf{l}, \dot{\mathbf{l}}) &= \mathbf{J}^{-1T} \left(\mathbf{M}(\mathbf{q}) \dot{\mathbf{J}}^{-1} + \mathbf{H}(\dot{\mathbf{q}}, \mathbf{q}) \mathbf{J}^{-1} \right), \end{aligned} \quad (14)$$

$$\mathbf{G}_L = \mathbf{J}^{-1T} \mathbf{G}_q.$$

The impact of the centrifugal/Coriolis term is small, which can be ignored, and equation (13) can be rewritten as

$$\mathbf{f} = \mathbf{M}_L(\mathbf{l}) \cdot \ddot{\mathbf{l}} + \mathbf{G}_L. \quad (15)$$

2.3. Dynamic Effects of Load. Before establishing the identification model and designing the excitation trajectory, it is necessary to analyze the impact of the load on the dynamics of the manipulator. Analyzing the dynamic equation (11), the equation mainly includes three items, Coriolis/centripetal force, gravity force, and inertial force. The load will affect the value of the Coriolis force/centripetal force, but this item of parallel manipulator is generally small and generally ignored. The gravity term is only related to the mass of the system. For parallel manipulators, the mass of the load is usually a constant value and can be easily measured, so it is not necessary to identify the load mass.

Therefore, we mainly analyze the effect of load on the inertial force of the system.

The inertial forces and torques of the system are as follows:

$$\begin{aligned}
f_{I,x} &= m_a \ddot{x} + m_p \ddot{x} + m_p l_z \dot{\omega}_y - m_p l_y \dot{\omega}_z, \\
f_{I,y} &= m_a \ddot{y} + m_p \ddot{y} - m_p l_z \dot{\omega}_x + m_p l_x \dot{\omega}_z, \\
f_{I,z} &= m_a \ddot{z} + m_p \ddot{z} + m_p l_y \dot{\omega}_x + m_p l_x \dot{\omega}_y, \\
M_{I,rx} &= ({}^B I_{p,xx} + {}^B I_{l,xx}) \dot{\omega}_x - m_p l_z \ddot{y} + m_p l_y \ddot{z} - m_p l_x l_y \dot{\omega}_y - m_p l_x l_z \dot{\omega}_z - {}^B I_{l,xy} \dot{\omega}_y - {}^B I_{l,xz} \dot{\omega}_z - {}^B I_{p,xy} \dot{\omega}_y - {}^B I_{p,xz} \dot{\omega}_z, \\
M_{I,ry} &= ({}^B I_{p,yy} + {}^B I_{l,yy}) \dot{\omega}_y + m_p l_z \ddot{x} - m_p l_x \ddot{z} - m_p l_y l_z \dot{\omega}_z - m_p l_x l_y \dot{\omega}_x - {}^B I_{l,xy} \dot{\omega}_x - {}^B I_{l,yz} \dot{\omega}_z - {}^B I_{p,xy} \dot{\omega}_x - {}^B I_{p,yz} \dot{\omega}_z, \\
M_{I,rz} &= ({}^B I_{p,zz} + {}^B I_{l,zz}) \dot{\omega}_z - m_p l_y \ddot{x} + m_p l_x \ddot{y} - m_p l_x l_z \dot{\omega}_x - m_p l_y l_z \dot{\omega}_y - {}^B I_{l,xz} \dot{\omega}_x - {}^B I_{l,yz} \dot{\omega}_y - {}^B I_{p,xz} \dot{\omega}_x - {}^B I_{p,yz} \dot{\omega}_y,
\end{aligned} \tag{16}$$

where ${}^B I_{p,xx}$, ${}^B I_{p,yy}$, and ${}^B I_{p,zz}$ are the moments of inertia of moving platform in the static coordinate system, and ${}^B I_{p,xx}$, ${}^B I_{p,yy}$, and ${}^B I_{p,zz}$ are the products of inertia of moving platform in the static coordinate system; ${}^B I_{p,xx}$, ${}^B I_{p,yy}$, and ${}^B I_{p,zz}$ are the moments of inertia of load in the static coordinate system, and ${}^B I_{l,xx}$, ${}^B I_{l,yy}$, and ${}^B I_{l,zz}$ are the products of inertia of load in the static coordinate system.

Equation (16) is the inertial force equation of the PM with load. The equation shows that the load dynamic parameters, such as mass, inertia tensor, and position of load centroid have an effect on the inertial force or moment of the system. Analyzing equation (16), we can see that the inertial

force or moment on a certain degree of freedom is not only determined by the acceleration on that degree of freedom, but also determined by the acceleration on the other degrees of freedom. So the PM with load has strong dynamic coupling characteristics among the six degrees of freedom. In addition, it is easy to know that the dynamic coupling is mainly caused by load.

Equation (16) can be rewritten as

$$\mathbf{F}_I = \mathbf{M}(\mathbf{q})\ddot{\mathbf{q}}. \tag{17}$$

The mass matrix of equation (17) can be expressed by

$$\mathbf{M}(\mathbf{q}) = \begin{bmatrix} m & 0 & 0 & 0 & m_p l_z & -m_p l_y \\ 0 & m & 0 & -m_p l_z & 0 & m_p l_x \\ 0 & 0 & m & m_p l_y & -m_p l_x & 0 \\ 0 & -m_p l_z & m_p l_y & {}^B I_{l,xx} + {}^B I_{p,xx} & -{}^B I_{p,xy} - {}^B I_{l,xy} - m_p l_x l_y & -{}^B I_{p,xz} - {}^B I_{l,xz} - m_p l_x l_z \\ m_p l_z & 0 & -m_p l_x & -{}^B I_{p,xy} - {}^B I_{l,xy} - m_p l_x l_y & {}^B I_{l,yy} + {}^B I_{p,yy} & -{}^B I_{p,yz} - {}^B I_{l,yz} - m_p l_y l_z \\ -m_p l_y & m_p l_x & 0 & -{}^B I_{p,xz} - {}^B I_{l,xz} - m_p l_x l_z & -{}^B I_{p,yz} - {}^B I_{l,yz} - m_p l_y l_z & {}^B I_{l,zz} + {}^B I_{p,zz} \end{bmatrix}. \tag{18}$$

The off-diagonal elements in $\mathbf{M}(\mathbf{q})$ are the cause of the dynamic coupling among the six DOFs. Extracting these nondiagonal elements, the coupled mass matrix $\mathbf{M}_c(\mathbf{q})$ can be obtained:

$$\mathbf{M}_c(\mathbf{q}) = \begin{bmatrix} 0_{3 \times 3} & \mathbf{L}_{3 \times 3} \\ \mathbf{J}_{3 \times 3} & \mathbf{K}_{3 \times 3} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & m_p l_z & -m_p l_y \\ 0 & 0 & 0 & -m_p l_z & 0 & m_p l_x \\ 0 & 0 & 0 & m_p l_y & -m_p l_x & 0 \\ 0 & -m_p l_z & m_p l_y & 0 & -{}^B I_{p,xy} - {}^B I_{l,xy} - m_p l_x l_y & -{}^B I_{p,xz} - {}^B I_{l,xz} - m_p l_x l_z \\ m_p l_z & 0 & -m_p l_x & -{}^B I_{p,xy} - {}^B I_{l,xy} - m_p l_x l_y & 0 & -{}^B I_{p,yz} - {}^B I_{l,yz} - m_p l_y l_z \\ -m_p l_y & m_p l_x & 0 & -{}^B I_{p,xz} - {}^B I_{l,xz} - m_p l_x l_z & -{}^B I_{p,yz} - {}^B I_{l,yz} - m_p l_y l_z & 0 \end{bmatrix}, \tag{19}$$

where $0_{3 \times 3}$ is third-order zero matrix and $\mathbf{L}_{3 \times 3}$, $\mathbf{J}_{3 \times 3}$, and $\mathbf{K}_{3 \times 3}$ are the third-order submatrix of $\mathbf{M}_c(\mathbf{q})$. First, analyze the

submatrix \mathbf{L} . The nonzero elements in the matrix \mathbf{L} are the coefficients of angular acceleration in the inertial force

equation. Thus, the element value in this matrix can be used to measure the dynamic coupling influence of rotational motion on the translational motion. The larger the element value, the stronger the coupling effect. For example, if the matrix is a zero matrix, the coefficients representing the angular acceleration in the inertial force term are all zero, indicating that the rotational motion does not affect the translational motion. In the same way, the matrix \mathbf{L} is composed of the coefficients of each translational acceleration in the moment of inertia, which represents the influence of translational motion on rotational motion. It can be seen from equation (19) that the elements in \mathbf{J} and \mathbf{L} are only related to the load, and the value of each nonzero element is determined by the load mass and the position of the centroid, so it is important to know their values a priori. Matrix \mathbf{K} describes the dynamic coupling between the three rotational degrees of freedom and the nondiagonal elements in the matrix \mathbf{K} are mainly determined by the load dynamic parameters.

In summary, the load has a greater impact on the dynamics of the system, especially the inertial force of the load. Therefore, the dynamic identification of the load is very important.

3. Identification Process

In order to eliminate the influence of the load on the dynamic characteristics of the parallel manipulator and ensure the application of some advanced model-based control strategies, it is necessary to identify the parameters of the dynamic model of the load. Since the mass of the load is generally easier to determine, this paper only studies the parameter identification of the position of the centroid and the inertia tensor of the load. The inertia tensor and the position of the center of mass used in equation (11) are referred to as the dynamic parameters in the static coordinate system.

The inertia tensor of load is a 3×3 matrix and can be expressed by

$${}^B\mathbf{I}_L = \mathbf{R} \cdot {}^A\mathbf{I}_L \cdot \mathbf{R}^T, \quad (20)$$

where ${}^A\mathbf{I}_L$ is the inertial tensor of load in moving coordinate system.

The position of the center of mass is a column vector and its equation is

$${}^B\mathbf{b} = \mathbf{R} \cdot {}^A\mathbf{b}, \quad (21)$$

where ${}^A\mathbf{b} = [\Delta x \ \Delta y \ \Delta z]^T$ is ${}^B\mathbf{b} = [l_x \ l_y \ l_z]^T$.

From equations (20) and (21), it is easy to know that the dynamic parameters are not constant and they change with the excitation trajectory of three rotational DOFs. However, the parameter identification of variables is very difficult and the corresponding real-time identification algorithm needs to be designed. To cope with this problem, the inertia tensor and the position of the center of mass relative to the moving coordinate system are selected as a base parameters set because these two parameters are constant and not affected by the trajectory of excitation, which can reduce the difficulty of identification.

3.1. Excitation Trajectory. Although there is no variable in the basic parameters set, the equation of inertial force/moment with regard to the identified parameters is not simple. In equation (16), the identification parameters are coupled with each other, which is impossible to separate. In order to reduce the coupling degree of the identified parameters in the dynamic equation, it is necessary to simplify the dynamic equation with regard to the identified parameters. When the given signal is a translational degree of freedom signal with a small amplitude, the coupling amplitude of each rotational degree of freedom is relatively small, and the system can be regarded as a small translational movement near the zero position, and the rotation matrix R can be treated as the identity matrix. In addition, when the given signal is a rotation signal with a small amplitude (less than 1 degree), the rotation matrix can also be regarded as the identity matrix. When the rotation matrix R is the identity matrix, formula (16) can be simplified to formula

$$\begin{aligned} f_{I,x}(t) &= m_a \ddot{x}(t) + m_p \ddot{x}(t) + m_p \Delta z \dot{\omega}_y(t), \\ f_{I,y}(t) &= m_a \ddot{y}(t) + m_p \ddot{y}(t) - m_p \Delta z(t) \dot{\omega}_x, \\ f_{I,z}(t) &= m_a \ddot{z}(t) + m_p \ddot{z}(t), \\ M_{I,rx}(t) &= ({}^A I_{p,xx} + {}^A I_{l,xx}(t) + m_p \Delta z(t)^2) \dot{\omega}_x(t) - m_p \Delta z(t) y(t), \\ M_{I,ry}(t) &= ({}^A I_{p,yy} + {}^A I_{l,yy}(t) + m_p \Delta z(t)^2) \dot{\omega}_y(t) + m_p \Delta z(t) \ddot{x}(t), \\ M_{I,rz}(t) &= ({}^A I_{p,zz} + {}^A I_{l,zz}(t)) \dot{\omega}_z(t). \end{aligned} \quad (22)$$

The identified parameters in equation (22) are ${}^A I_{l,xx}$, ${}^A I_{l,yy}$, ${}^A I_{l,zz}$, and Δz . In order to obtain sufficient information on these parameters, the exciting trajectory should be well designed.

Equation (3) requires that the amplitude of the given excitation signal on rotational direction should be small enough. Because when the rotation degree of freedom excitation signal is applied, the simplified model will

have a large modeling error relative to the original model. Due to the dynamic coupling characteristics between Dx and Ry , the motion on Dx and Ry both can excite dynamic parameters ${}^A I_{l,yy}$ and Δz . For these two parameters, the dx direction is selected to apply an excitation signal because the translation signal satisfies equation (6) better than the rotation signal. Similarly, the dx direction is selected to apply an excitation signal to excite dynamic parameters ${}^A I_{l,xx}$ and Δz . Since the last dynamic parameter ${}^A I_{l,zz}$ only exists in the equation of $\mathbf{M}_{I,rz}$, the Rz direction is selected to apply an excitation signal to excite dynamic parameter ${}^A I_{l,zz}$. Note that the amplitude of the excitation signal on Rz direction should satisfy the condition of equation (22).

The excitation trajectory of the system is as follows:

$$\begin{aligned} dx &= 0.2 \cdot \sin(\pi t), \\ dy &= 0.2 \cdot \sin(\pi t), \\ rz &= 0.2 \cdot \sin(\pi t). \end{aligned} \quad (23)$$

It can be seen from the above equation that the excitation signals of the system are two translational signals, one rotational signal, and the rotational signal with a small amplitude, so it satisfies the system dynamics equation (22). Note that the excitation signal in the rotation direction cannot be too small; otherwise, the dynamic characteristics of the system cannot be well excited.

3.2. EKF Identification Process. Select the state variable of the system: angular velocity in Rx , Ry , and Rz directions, centroid eccentricity z , and load moment of inertia ${}^A I_{l,xx}$, ${}^A I_{l,yy}$, ${}^A I_{l,zz}$.

The state vector of the system is

$$\mathbf{x} = [v_{rx}(t) \ v_{ry}(t) \ v_{rz}(t) \ z(t) \ {}^A I_{l,xx}(t) \ {}^A I_{l,yy}(t) \ {}^A I_{l,zz}(t)]^T. \quad (24)$$

Establish the system differential equation as follows:

$$\begin{aligned} \dot{v}_{rx}(t) &= \frac{m_p z(t) f_{I,y}(t) + m M_{I,rx}(t)}{m \left(({}^A I_{p,xx} + {}^A I_{l,xx}(t) + m_p z(t)^2) \right) - m_p^2 z(t)^2}, \\ \dot{v}_{ry}(t) &= \frac{m_p z(t) f_{I,x}(t) - m M_{I,ry}(t)}{m_p^2 z(t)^2 - m \left(({}^A I_{p,yy} + {}^A I_{l,yy}(t) + m_p z(t)^2) \right)}, \\ \dot{v}_{rz}(t) &= \frac{M_{I,rz}(t)}{{}^A I_{p,zz} + {}^A I_{l,zz}(t)}, \\ \dot{z}(t) &= 0, \\ {}^A \dot{I}_{l,xx}(t) &= 0, \\ {}^A \dot{I}_{l,yy}(t) &= 0, \\ {}^A \dot{I}_{l,zz}(k) &= 0. \end{aligned} \quad (25)$$

The discrete form of formula (25) is as follows:

$$\begin{aligned} \dot{v}_{rx}(k) &= \frac{m_p z(k-1) f_{I,y}(k-1) + m M_{I,rx}(k-1)}{m \left(({}^A I_{p,yy} + {}^A I_{l,yy}(t)(k-1) + m_p z(k-1)^2) \right) - m_p^2 z(k-1)^2}, \\ \dot{v}_{ry}(k) &= \frac{m_p z(k-1) f_{I,x}(k-1) - m M_{I,ry}(k-1)}{m_p^2 z(k-1)^2 - m \left(({}^A I_{p,xx} + {}^A I_{l,xx}(t)(k-1) + m_p z(k-1)^2) \right)}, \\ \dot{v}_{rz}(k) &= \frac{M_{I,rz}(k-1)}{{}^A I_{p,zz} + {}^A I_{l,zz}(t)(k-1)}, \\ \dot{z}(k) &= 0, \\ \dot{I}_{pc,xx}(k) &= 0, \\ \dot{I}_{pc,yy}(k) &= 0, \\ \dot{I}_{pc,zz}(k) &= 0. \end{aligned} \quad (26)$$

The derivative of the state vector is as follows:

$$\dot{\mathbf{x}}_k = [\dot{v}_{rx}(k) \ \dot{v}_{ry}(k) \ \dot{v}_{rz}(k) \ \dot{z}(k) \ \dot{I}_{l,xx}(k) \ \dot{I}_{l,yy}(k) \ \dot{I}_{l,zz}(k)]^T. \quad (27)$$

From formulas (26) and (27), the state equation of the system is

$$\mathbf{x}_k = \dot{\mathbf{x}}_{k-1} \Delta t + \mathbf{x}_{k-1} + \mathbf{W}_{k-1}. \quad (28)$$

The observed values of the system are the angular velocities in the directions of the three degrees of freedom of Rx , Ry , and Rz , and the observation equation is as follows:

$$\begin{aligned} z_{rx}(k) &= v_{rx}(k) + V_{rx}(k), \\ z_{ry}(k) &= v_{ry}(k) + V_{ry}(k), \\ z_{rz}(k) &= v_{rz}(k) + V_{rz}(k). \end{aligned} \quad (29)$$

The observation vector and the observation noise vector are as follows:

$$\begin{aligned} \mathbf{z}_k &= [z_{rx}(k) \ z_{ry}(k) \ z_{rz}(k)]^T, \\ \mathbf{V}_k &= [V_{rx}(k) \ V_{ry}(k) \ V_{rz}(k)]^T. \end{aligned} \quad (30)$$

Equations (28) and (29) are the state space equation and observation equation of the system, respectively, which are written as

$$\begin{cases} \mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}) + \mathbf{W}_{k-1}, \\ \mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{V}_{k-1}, \end{cases} \quad (31)$$

where $\mathbf{H} = [\mathbf{E}_{3 \times 3} \ 0_{4 \times 3}]$ is the observation matrix of the system.

Equation (31) is the system model that needs to be identified. The extended Kalman filter algorithm is used to estimate the parameters of the model. The extended Kalman filter algorithm is mainly divided into two parts, the time update part and the measurement update part. The two parts are introduced separately below.

3.2.1. Time Update. The main purpose of the time update equation is to calculate the next prior estimate vector on the

basis of the last optimal estimate vector and also to update the prior error covariance matrix. The time update (prediction) equation is

$$\begin{aligned}\hat{x}_k^- &= f(\hat{x}_{k-1}), \\ P_k^- &= J_{k-1}P_{k-1}J_{k-1}^T + Q_{k-1}.\end{aligned}\quad (32)$$

When the system performs state transition, there will be state transition noise, which is Gaussian white noise, and its covariance matrix is \mathbf{Q} in the time update equation.

\mathbf{J} is the Jacobian matrix of the system, which is obtained by the partial derivative of the state function f on the state vector x . The expression of the Jacobian matrix of the identification model in this paper is as follows

$$\mathbf{J} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \cdots & \frac{\partial f_1}{\partial x_7} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \cdots & \frac{\partial f_2}{\partial x_7} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_7}{\partial x_1} & \frac{\partial f_7}{\partial x_2} & \cdots & \frac{\partial f_7}{\partial x_7} \end{bmatrix} = \begin{bmatrix} \mathbf{E}_{3 \times 3} & \mathbf{N} \\ \mathbf{0}_{4 \times 3} & \mathbf{E}_{4 \times 4} \end{bmatrix}.\quad (33)$$

\mathbf{N} is the submatrix of \mathbf{J} , and its expression is as follows: where

$$\mathbf{N} = \begin{bmatrix} j_{11} & j_{12} & 0 & 0 \\ j_{21} & 0 & j_{22} & 0 \\ 0 & 0 & 0 & j_{34} \end{bmatrix},\quad (34)$$

$$\begin{aligned}j_{11} &= \frac{\Delta t f_{I,x}(k-1)m_p}{m_p^2 z(k-1)^2 - m(AI_{p,xx} + AI_{l,xx}(t)(k-1) + m_p^2 z^2(k-1))}, \\ &= \frac{\Delta t(2m_p^2 z(k-1) - 2mm_p z(k-1))(m_p z(k-1)f_{I,x}(k-1) - mM_{I,ry}(k-1))}{[m_p^2 z^2(k-1) - m(AI_{p,xx} + AI_{l,xx}(t)(k-1) + m_p z^2(k-1))]^2}, \\ j_{12} &= \left(\frac{m(m_p z(k-1)f_{I,x}(k-1) - mM_{I,ry}(k-1))}{[m_p^2 z^2(k-1) - m(AI_{p,xx} + AI_{l,xx}(t)(k-1) + m_p z^2(k-1))]^2} \right) \Delta t, \\ j_{21} &= \frac{f_{I,y}(k-1)m_p \Delta t}{m((AI_{p,yy} + AI_{l,yy}(t) + m_p z(k-1)^2)) - m_p^2 z(k-1)^2} \\ &= \frac{(2mm_p z(k-1) - 2m_p^2 z(k-1))(m_p z(k-1)f_{I,y}(k-1) + mM_{I,rx}(k-1))\Delta t}{[m((AI_{p,yy} + AI_{l,yy}(t)(k-1) + m_p z(k-1)^2)) - m_p^2 z(k-1)^2]^2}, \\ j_{22} &= \frac{m(m_p z(k-1)f_{I,y}(k-1) + mM_{I,rx}(k-1))\Delta t}{[m((AI_{p,yy} + AI_{l,yy}(k-1) + m_p z(k-1)^2)) - m_p^2 z(k-1)^2]^2}, \\ j_{34} &= \left(\frac{M_{I,rz}(k-1)}{[AI_{p,zz} + AI_{l,zz}(k-1)]^2} \right) \Delta t.\end{aligned}\quad (35)$$

Algorithm extended Kalman filter algorithm
(1) Initialize the estimated value of the state variable and the error covariance matrix
(2) Repeat
(3) Calculate prior state estimates, $\hat{x}_k^- = f(\hat{x}_{k-1})$
(4) Calculate the Jacobian matrix, J_{k-1}
(5) Calculate the prior error covariance, $P_k^- = J_{k-1}P_{k-1}J_{k-1}^T + Q_{k-1}$
(6) Calculate Kalman gain, $K_k = P_k^-H^T(HP_k^-H^T + R_k)^{-1}$
(7) Update the posterior state estimate, $\hat{x}_k = \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-)$
(8) Update posterior error covariance, $P_k = (I - K_kH)P_k^-$
(9) Output the best estimate of this iteration
(10) Until Simulation stopped
(11) End

ALGORITHM 1: Identification algorithm.

3.2.2. *Measurement Update.* The a priori estimated value and the actual value of the system are not necessarily equal, so the a priori estimated value needs to be corrected by the measured value. The main purpose of the measurement update equation is to find the optimal estimated value of the current iteration of the system. Update the Kalman gain, the optimal estimate of the system, and the posterior error covariance. The measurement update equation is

$$\begin{aligned} K_k &= P_k^-H^T(HP_k^-H^T + R_k)^{-1}, \\ \hat{x}_k &= \hat{x}_k^- + K_k(z_k - H\hat{x}_k^-), \\ P_k &= (I - K_kH)P_k^-. \end{aligned} \quad (36)$$

In the equation, K is the Kalman gain and P is the posterior error covariance matrix. The system will produce measurement noise during measurement. This noise is white noise. R in the measurement update equation is the covariance matrix of the measurement noise. Now, we summarize the EKF identification algorithm for the parallel robot manipulator as shown in Algorithm 1.

4. Numerical Simulations

In this section, the simulation analysis is carried out in MATLAB/Simulink. For the simulation model, the mechanical model is built using Multibody and the sampling time is set to 1 ms. Moreover, the main parameters of the Stewart robot with uncertain load are shown in Table 1.

Corresponding simulation strategies are designed for load dynamics parameter identification, as shown in Figure 3.

First, a specific excitation signal is used to excite the dynamic characteristics of the load. Then, the displacement, velocity, and output force of each leg were collected. According to the kinematic analysis of the parallel manipulator, the displacement, velocity, and force in the DOF space can be obtained by the kinematic positive solution or the velocity Jacobian matrix transformation of these measured values. Finally, the displacement, velocity, and force in the DOF space are input into the estimator and the estimator can calculate the dynamic parameters of the load based on the previously designed identification algorithm.

According to Figure 4, the corresponding simulation system was built in MATLAB/Simulink, as shown in

TABLE 1: Main parameters of the Stewart robot with load.

Parameters	Value
Radius of upper/lower platform (m)	0.4/0.6
Initial length of the linear hydraulic cylinder (m)	0.8741
Min/max stroke of the hydraulic cylinder (m)	-0.3/0.3
Mass of load (kg)	500
Mass of upper platform (kg)	100
Moment of inertia of the upper platform (kg·m ²)	diag(25, 25, 44)

Figure 4. The first is a signal generator block used to generate a target trajectory. The second is the kinematics block and its main function is to convert the task space signal to the joint space signal and calculate the speed Jacobian matrix. The third is the PID controller block. The fourth is the hydraulic system block. The fifth is the mechanical model of the Stewart parallel mechanism. The sixth is the EKF estimator.

In the simulation, the initial estimated value of the state variables is all 0 and the initial error covariance matrix is diag(1 1 1 1 10 500 500). Moreover, the value of the dynamic parameters to be identified is shown in Table 2.

The identification results of load dynamics parameters are shown in Figure 4.

It can be seen from Figure 5 that the proposed method in this paper can estimate the dynamic parameters of loads. It can be seen from Figure 5(a) that this identification method has a high estimation accuracy for centroid eccentricity along the z -axis direction, and the identification of Δz curve in the figure almost overlaps with the real value. Figures 5(b)–5(d) are identification curves of load dynamics parameters ${}^A I_{l,xx}$, ${}^A I_{l,yy}$, and ${}^A I_{l,zz}$, respectively. Although there is a certain error between the values of the three inertial parameters and the real values, the error is not large.

Table 3 shows the true value X_r , identification value X_{id} , and relative error e_r of the load dynamic parameters obtained by the proposed method. The relative error e_r is defined as

$$e_r = \frac{X_r - X_{id}}{X_r} \times 100\%. \quad (37)$$

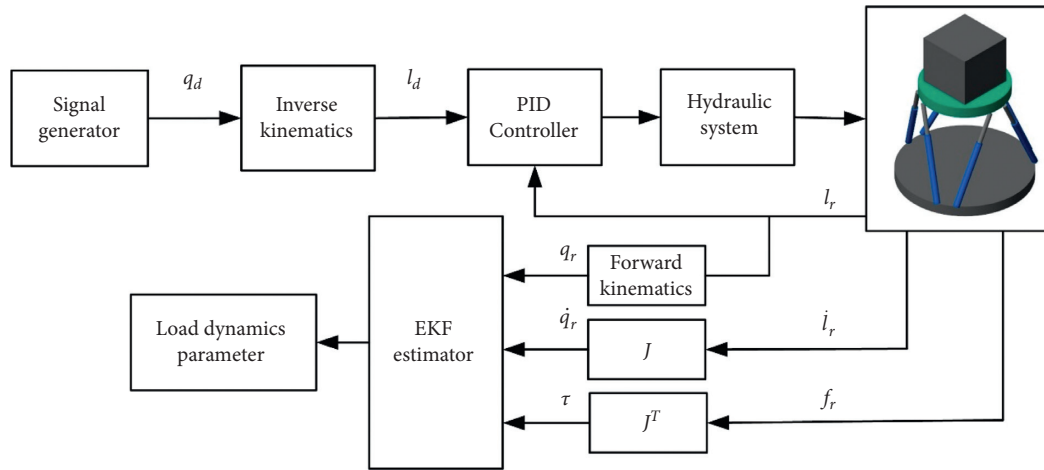


FIGURE 3: Load dynamic parameter identification system.

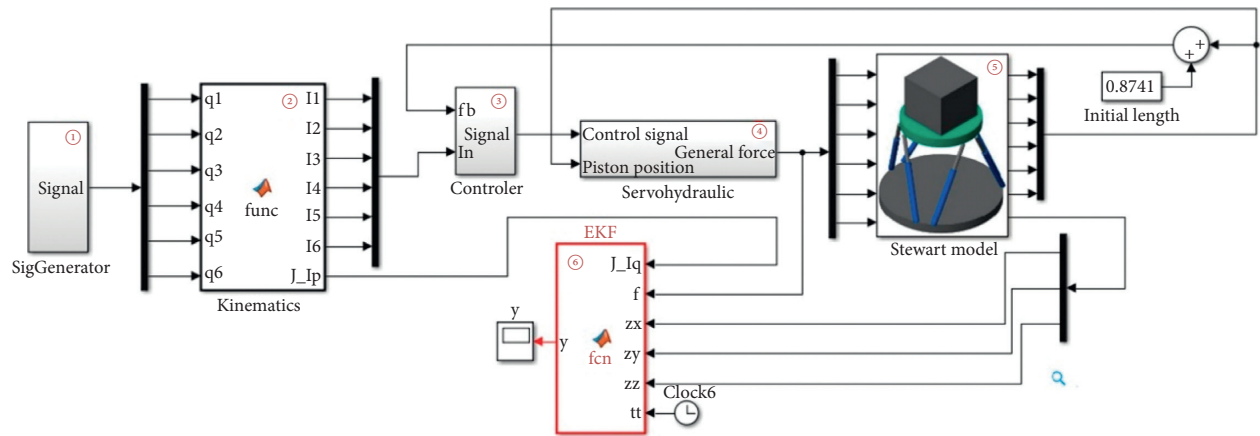
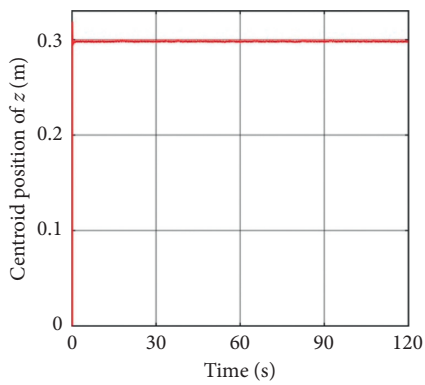


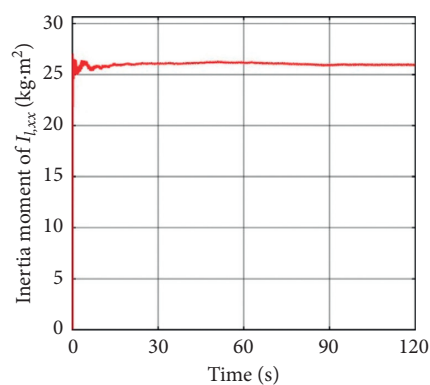
FIGURE 4: Simulink simulation model of load dynamics parameter identification system.

TABLE 2: Load dynamics parameters to be identified.

Symbol	Parameters	Value	Unit
Δz	Centroid position	0.3	m
$\text{diag}({}^A I_{l,xx} {}^A I_{l,yy} {}^A I_{l,zz})$	Moment of inertia of load ($\text{kg}\cdot\text{m}^2$)	$\text{diag}(25, 25, 25)$	$\text{kg}\cdot\text{m}^2$



(a)



(b)

FIGURE 5: Continued.

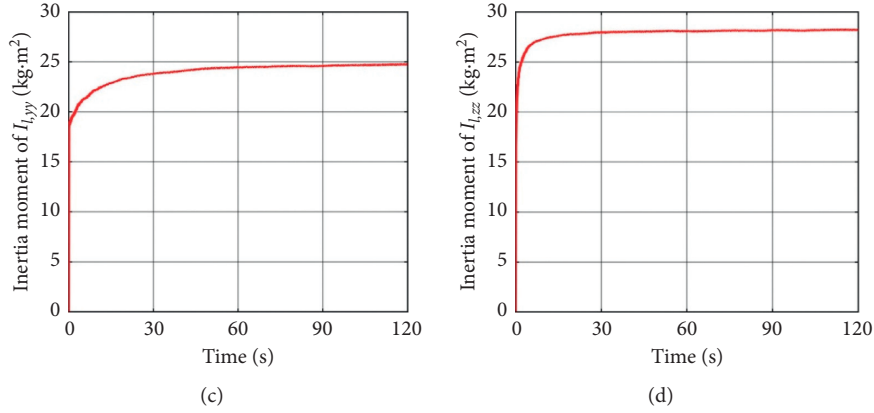


FIGURE 5: Identification results of load dynamics parameters. (a) The identification value of Δz . (b) The identification value of ${}^A I_{l,xx}$. (c) The identification value of ${}^A I_{l,yy}$. (d) The identification value of ${}^A I_{l,zz}$.

TABLE 3: Real and identification inertia parameters of the payload.

Parameter	Real value	Identification value	Relative error (%)
Δz	0.3	0.2978	0.73
${}^A I_{l,xx}$	25	25.94	3.76
${}^A I_{l,yy}$	25	24.75	1.00
${}^A I_{l,zz}$	25	28.20	12.80

5. Conclusion

The proposed method has been successfully applied to a six-degree-of-freedom Gough–Stewart parallel manipulator for load dynamic parameters estimation. The identification algorithm is simple and easy to implement. Compared with the traditional least square method, the proposed identification approach does not require linearization of the dynamic model and optimization of the excitation trajectory. Moreover, this method is not sensitive to measurement noise and without acceleration measurements in the process of identification.

Note that the method proposed in this paper is an offline identification method. How to study a load real-time identification algorithm not limited to trajectory needs further study.

Data Availability

All data included in this study are available upon request to the corresponding author.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Defense Pre-Research Foundation of China (1126170104A, 1126180204B, 1126190508A, and 1126190508A).

References

- [1] J.-H. Park, P. Stegall, and S. K. Agrawal, "Dynamic brace for correction of abnormal postures of the human spine," in *Proceedings 2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 5922–5927, Seattle, WA, USA, May 2015.
- [2] J. P. Merlet, *Parallel Robots (Solid Mechanics and Its Applications)*, pp. 2091–2127, Springer, Berlin, Germany, 2010.
- [3] C. Yang, H. Wu, Z. Li, W. He, N. Wang, and C.-Y. Su, "Mind control of a robotic arm with visual fusion technology," *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 3822–3830, 2018.
- [4] H. Lin, T. Zhang, Z. Chen, H. Song, and C. Yang, "Adaptive fuzzy gaussian mixture models for shape approximation in robot grasping," *International Journal of Fuzzy Systems*, vol. 21, no. 4, pp. 1026–1037, 2019.
- [5] P. Masarati, "Computed torque control of redundant manipulators using general-purpose software in real-time," *Multibody System Dynamics*, vol. 32, no. 4, pp. 403–428, 2013.
- [6] X. Niu, C. Yang, B. Tian, X. Li, and J. Han, "Modal decoupled dynamics feed-forward active force control of spatial multi-dof parallel robotic manipulator," *Mathematical Problems in Engineering*, vol. 2019, Article ID 1835308, 13 pages, 2019.
- [7] J.-F. He, H.-Z. Jiang, and Z.-Z. Tong, "Modal control of a hydraulically driven redundant actuated fully parallel mechanism," *Journal of Vibration and Control*, vol. 23, no. 10, pp. 1585–1592, 2017.
- [8] J. Zhao, Z. Wang, T. Yang, J. Xu, Z. Ma, and C. Wang, "Design of a novel modal space sliding mode controller for electro-hydraulic driven multi-dimensional force loading parallel mechanism," *ISA Transactions*, vol. 99, pp. 374–386, 2020.
- [9] C. Yang, Y. Jiang, W. He, J. Na, Z. Li, and B. Xu, "Adaptive parameter estimation and control design for robot manipulators with finite-time convergence," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 10, pp. 8112–8123, 2018.
- [10] C. Yang, G. Peng, L. Cheng et al., "Force sensorless admittance control for teleoperation of uncertain robot manipulator using neural networks," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2019.
- [11] A. R. Plummer and P. S. Guinzio, "Modal control of an electrohydrostatic flight simulator motion system," in *Proceedings of the ASME 2009 Dynamic Systems and Control Conference*, Hollywood, CA, USA, October 2009.

- [12] G. Peng, C. Yang, W. He et al., "Force sensorless admittance control with neural learning for robots with actuator saturation," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 4, pp. 3138–3148, 2019.
- [13] G. Shen, G. Li, W. Zang, X. Li, and Y. Tang, "Modal space feedforward control for electro-hydraulic parallel mechanism," *IEEE Access*, vol. 7, pp. 39751–39761, 2019.
- [14] Y. Pi and X. Wang, "Trajectory tracking control of a 6-DOF hydraulic parallel robot manipulator with uncertain load disturbances," *Control Engineering Practice*, vol. 19, no. 2, pp. 185–193, 2011.
- [15] C. Urrea and J. Pascal, "Design, simulation, comparison and evaluation of parameter identification methods for an industrial robot," *Computers & Electrical Engineering*, vol. 67, pp. 791–806, 2018.
- [16] W. Khalil, M. Gautier, and P. Lemoine, "Identification of the payload inertial parameters of industrial manipulators," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pp. 4943–4948, Roma, Italy, April 2007.
- [17] J. Jiang and Y. Zhang, "A revisit to block and recursive least squares for parameter estimation," *Computers & Electrical Engineering*, vol. 30, no. 5, pp. 403–416, 2004.
- [18] J. Hu and R. Xiong, "Contact force estimation for robot manipulator using semiparametric model and disturbance kalman filter," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 4, pp. 3365–3375, 2017.
- [19] J. Jung, J. Lee, and K. Huh, "Robust contact force estimation for robot manipulators in three-dimensional space," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 220, no. 9, pp. 1317–1327, 2006.
- [20] G. G. Rigatos, "Derivative-free nonlinear kalman filtering for mimo dynamical systems: application to multi-dof robotic manipulators," *International Journal of Advanced Robotic Systems*, vol. 8, no. 6, p. 72, 2011.
- [21] M. M. Olsen, J. Swevers, and W. Verdonck, "Maximum likelihood identification of a dynamic robot model: implementation issues," *The International Journal of Robotics Research*, vol. 21, no. 2, pp. 89–96, 2002.
- [22] J. Swevers, C. Gansseman, D. B. Tukul, J. De Schutter, and H. Van Brussel, "Optimal robot excitation and identification," *IEEE Transactions on Robotics and Automation*, vol. 13, no. 5, pp. 730–740, 1997.
- [23] J. Yuan, W. Wan, X. Fu et al., "A novel LLSDPso method for nonlinear dynamic parameter identification," *Assembly Automation*, vol. 37, 2017.
- [24] C.-T. Chen, J.-C. Renn, and Z.-Y. Yan, "Experimental identification of inertial and friction parameters for electro-hydraulic motion simulators," *Mechatronics*, vol. 21, no. 1, pp. 1–10, 2011.
- [25] T. Tian, H. Jiang, Z. Tong, J. He, and Q. Huang, "An inertial parameter identification method of eliminating system damping effect for a six-degree-of-freedom parallel manipulator," *Chinese Journal of Aeronautics*, vol. 28, no. 2, pp. 582–592, 2015.
- [26] S. Briot and M. Gautier, "Global identification of joint drive gains and dynamic parameters of parallel robots," *Multibody System Dynamics*, vol. 33, no. 1, pp. 3–26, 2015.
- [27] J. Wu, J. Wang, and L. Wang, "Identification of dynamic parameter of a 3DOF parallel manipulator with actuation redundancy," *Journal of Manufacturing Science and Engineering*, vol. 130, no. 4, 2008.
- [28] T. D. Thanh, J. Kotlarski, B. Heimann, and T. Ortmaier, "Dynamics identification of kinematically redundant parallel robots using the direct search method," *Mechanism and Machine Theory*, vol. 52, pp. 277–295, 2012.
- [29] J. Wu, J. Wang, and Z. You, "An overview of dynamic parameter identification of robots," *Robotics and Computer-Integrated Manufacturing*, vol. 26, no. 5, pp. 414–419, 2010.
- [30] V. Mata, F. Benimeli, N. Farhat, and A. Valera, "Dynamic parameter identification in industrial robots considering physical feasibility," *Advanced Robotics*, vol. 19, no. 1, pp. 101–119, 2005.
- [31] V. Mata, N. Farhat, M. Diaz-Rodriguez, A. Valera, and A. Page, "Parallel manipulators towards new applications, I-tech, ch," *Dynamic Parameter Identification for Parallel Manipulators*, pp. 21–44, 2008.

Research Article

Adaptive Robust Dynamic Surface Integral Sliding Mode Control for Quadrotor UAVs under Parametric Uncertainties and External Disturbances

Ye Zhang ^{1,2}, Ning Xu,^{1,2} Guoqiang Zhu ^{1,2}, Lingfang Sun,^{1,2} Shengxian Cao ^{1,2},
and Xiuyu Zhang^{1,2}

¹School of Automation Engineering, Northeast Electric Power University, Jilin, China

²Jilin Province International Research Center of Precision Drive and Intelligent Control, Jilin, China

Correspondence should be addressed to Guoqiang Zhu; zhugqn@gmail.com

Received 18 September 2020; Revised 4 November 2020; Accepted 16 November 2020; Published 27 November 2020

Academic Editor: Ning Wang

Copyright © 2020 Ye Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A robust adaptive fuzzy nonlinear controller based on dynamic surface and integral sliding mode control strategy (ADSISMC) is proposed to realize trajectory tracking for a class of quadrotor UAVs. In this study, the composite factors including parametric uncertainties and external disturbances are added to controller design, which make it more realistic. The quadrotor model is divided into two subsystems of attitude and position that make the control design become feasible. The main contributions of the proposed ADSISMC strategy are as follows: (1) The combination of dynamic surface and integral sliding mode makes the system always in sliding stage by finding the appropriate initial position compared with the common sliding mode, and the complexity of explosion in backstepping method is eliminated. (2) By introducing the fuzzy system, the unknown functions and uncertainties can be approximated which significantly improves the robustness and the tracking performance. (3) The switching control strategy is utilized to compensate for the errors between estimated and ideal inputs; the tracking performance of the whole system has been significantly improved. The simulation results show the effectiveness of the proposed control method.

1. Introduction

As a newborn member of the small unmanned aerial vehicle (UVA) family, quadrotor has attracted much research interest due to its extensive utility in several important applications, such as commercial photography, military surveillance, rescue mission, and agricultural investigation [1–4]. Compared with traditional unmanned fixed-wing flight vehicles and manned airplanes, the main advantages of the quadrotor lie in small size, low cost, stable hovering, vertical take-off and landing (VTOL), convenient portability, and versatile features [5]. However, trajectory tracking control of the quadrotor is a thorny problem because of its nonlinear, underactuated dynamics, and strong coupling [6, 7]. Moreover, the quadrotor system is susceptible to external disturbances such as wind and nonlinear frictions. What is more, taking robustness of the trajectory tracking controller into consideration poses a bigger challenge [8].

In early quadrotor research stage, many studies used conventional linear control methods such as proportional-integral-derivative (PID) [9, 10] and linear quadratic regulator (LQR) [11] to design the quadrotor controller in order to improve the simpleness and practicability. The linear control technology was developed to stabilize the quadrotor by neglecting the unimportant factors and linearizing the dynamic model. Therefore, it is poor and even not acceptable for the tracking accuracy and the robustness of the quadrotor. To overcome the drawbacks of the aforementioned linear control approaches, a large number of nonlinear control strategies, including backstepping control [12–15], sliding mode control (SMC) [16–20], and feedback linearization control, are utilized to improve the tracking performance of the flight control system.

Backstepping control method has received extensive attention, not only in the quadrotor control, but also in some mechanical systems. The main idea of backstepping technology

is to select the appropriate state variable function as the recursive virtual control input. Once the final control input is received, the stability of the whole system is guaranteed. In [21–24], a backstepping controller has been designed to stabilize the attitude system of quadrotor. In [24], attitude control using hybrid backstepping methodology based on Frenet–Serret theory is studied in detail. The results show that the controller has good robustness under wind disturbance. To solve the problem of trajectory tracking, in [25], an adaptive controller combining parameter adaptive and backstepping control is designed. However, the obvious limitation of conventional backstepping design is the problem of “complexity of explosion” caused by the repeated differentiation of some nonlinear functions and the lack of robustness against uncertainties. To overcome this limitation of traditional backstepping control, dynamic surface control (DSC) is proposed as an effective alternative method [26–30]. In [27], a dynamic surface control method based on RBF neural network approximation is proposed for a class of nonlinear time-delay systems with state variables all measurable, which greatly simplifies the design process of the controller.

Moreover, to enhance the attitude performance robustness, disturbance observer (DOB), parameter estimation [31], and the approximation-based adaptive control are generally combined with DSC to handle external disturbances and parameterized uncertainties. For instance, a class of adaptive control methods using fuzzy logic systems or neural networks to approximate unknown functions in nonlinear systems have been proposed in [32–38]. In [33], a dynamic surface control-based adaptive fuzzy control method is proposed to overcome the “explosion of complexity” problem of classical backstepping. In [35], a robust dynamic surface controller based on extended state observer is presented for a quadrotor UAV subject to external disturbances and parametric uncertainties. In [36], both indirect and direct global neural controllers with the dynamic surface design are developed for the strict-feedback systems. The simulation results are presented to demonstrate the feasibility of the proposed global neural DSC design. A robot control scheme based on dynamic surface considering output error constraints, unknown dynamics, and bounded disturbance has been proposed in [39]; by introducing an improved virtual variable, the robustness of the control system was improved. However, the performance properties and robustness are not taken into account in these papers. As a commonly used nonlinear control method, the SMC is utilized as an effective method to design robust controllers for a specific class of nonlinear tracking problems in the presence of uncertain conditions [40–45]. Traditional SMC features the low sensitivity to the disturbances and parameter variations of the system [46–51]. In [50], a method based on second-order sliding mode control is used to avoid the chattering phenomenon for quadrotor UAVs. In [51], a robust backstepping sliding mode nonlinear controller for quadrotor UAVs is proposed to improve the robustness of the controller against model uncertainty and external disturbances. Compared with traditional sliding mode control method, the integral sliding mode (ISM) can guarantee that the system always meets the desired dynamic performance

index during the whole arrival period which significantly improves the robustness of the control system.

Motivated by the aforementioned observations, a new control methodology combined with dynamic surface and ISMC is proposed for the quadrotor trajectory tracking problem under parametric uncertainties and external disturbances. The main contributions of this paper are summarized as follows: (1) By fusing the technique of DSC and the integral SMC, a new integral sliding mode robust dynamic surface trajectory tracking controller is designed, which eliminates the “explosion of complexity” in the backstepping and improves the robustness of the whole system. (2) The FLSs are introduced to approach the ideal control law. And the estimations of the weight vector norm are utilized in the FLSs to significantly reduce the number of online estimation parameters. Therefore, the amount of calculation is obviously reduced, and the structure of the proposed controller is simplified. (3) The adaptive switching control is introduced to compensate the error between the real control law and the ideal control law, and the tracking performance of the whole system has been significantly improved.

The rest of this paper is organized as follows. The modeling of a quadrotor and some preliminaries are introduced in Section 2. The control algorithms are introduced in Section 3. Section 4 gives the stability analysis of the control system. Extensive simulations under different operating scenarios are given in Section 5. This paper ends with the conclusions in Section 6.

2. Problem Formulation and Preliminaries

2.1. The Mathematical Model of UAV. The quadrotor UAV is an underactuated system because it has six degrees of freedom, but only four actual inputs [1, 2]. In this paper, the quadrotor UAV with four rotors is shown in Figure 1. The equations of the dynamic quadrotor UAV are basically a rotating rigid body with six degrees of freedom [4, 5] which are usually derived by Newton–Euler formulas [8–10].

Define $\xi = [\phi, \theta, \varphi]^T$ and $\omega_b = [p, q, r]^T$, with ϕ , θ , and φ being the angle of roll, pitch, and yaw with respect to the inertia frame. p , q , and r are the angular velocity of roll, pitch, and yaw with respect to the body-fixed frame. The rotation matrix from the rigid frame to the inertia frame can be expressed as

$$R_t = \begin{pmatrix} C_\varphi C_\theta & C_\varphi S_\theta S_\phi - S_\varphi C_\phi & C_\varphi S_\theta C_\phi + S_\varphi S_\phi \\ S_\varphi C_\theta & S_\varphi S_\theta S_\phi + C_\varphi C_\phi & S_\varphi S_\theta C_\phi - C_\varphi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{pmatrix}, \quad (1)$$

where $S_{(\cdot)}$ and $C_{(\cdot)}$ denote $\sin(\cdot)$ and $\cos(\cdot)$, respectively. According to the rotation matrix R_t , the relationship between ξ and ω_b can be described as

$$\omega_b = R_t \dot{\xi} = \begin{pmatrix} 1 & 0 & -S_\theta \\ 0 & C_\phi & C_\theta S_\phi \\ 0 & -S_\phi & C_\theta C_\phi \end{pmatrix} \begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\varphi} \end{pmatrix}. \quad (2)$$

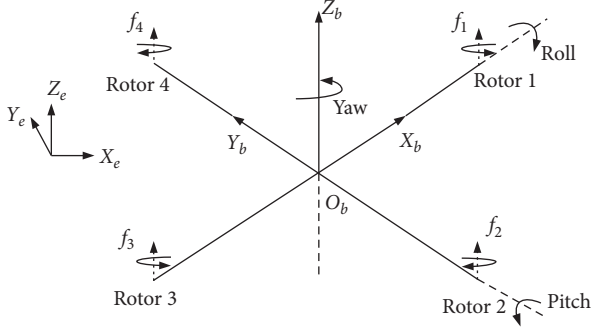


FIGURE 1: Schematic of the quadrotor UAV.

Define M_b as the torque provided by the rotors with respect to the body-fixed frame, and it is presented as follows:

$$M_b = \begin{pmatrix} M_{bx} \\ M_{by} \\ M_{bz} \end{pmatrix} = \begin{pmatrix} l_k(\Omega_4^2 - \Omega_2^2) \\ l_k(\Omega_3^2 - \Omega_1^2) \\ l(\Omega_4^2 + \Omega_2^2 - \Omega_1^2 - \Omega_3^2) \end{pmatrix}, \quad (3)$$

where Ω_i denotes the rotary speed of the front, right, rear, and left rotors, respectively; l_k is the distance between a rotor and the center of mass of the quadrotor; k is the drag force coefficient; and l is the reverse moment coefficient. Using the Newton–Euler equation, the rotational dynamic equation of the quadrotor is obtained as follows:

$$M_b = J_b \dot{\omega}_b + \omega_b \times J_b \omega_b + M_g + M_d, \quad (4)$$

where $J_b = \text{diag}(J_x, J_y, J_z)$ is a symmetric positive definite constant matrix with J_x , J_y , and J_z being the rotary inertia with respect to the $O_b X_b$, $O_b Y_b$, and $O_b Z_b$ axes, respectively; the notation \times denotes cross multiplication; M_g and M_d are the resultant torques due to the resultant of aerodynamic frictions torque and the gyroscopic effects. They are given as

$$M_g = \sum_{i=1}^4 \omega_b \times J_r [0, 0, (-1)^{i+1} \Omega_i]^T, M_d = \text{diag}(d_\phi, d_\theta, d_\varphi) \dot{\xi}, \quad (5)$$

where J_r denotes the moment of inertia of each rotor; d_ϕ , d_θ , and d_φ are the corresponding aerodynamic drag coefficients. According to (4), the following equation can be obtained:

$$\dot{\omega}_b = J_b^{-1} [M_b - M_g - M_d - \omega_b \times (J_b \omega_b)]. \quad (6)$$

Furthermore, with the help of approximation of Euler angles at equilibrium point, the following dynamic equations can be obtained:

$$\begin{cases} \ddot{\phi} = \frac{[\dot{\theta}\dot{\varphi}(J_y - J_z) - J_r \dot{\theta}\dot{\omega} - d_\phi \dot{\phi} + M_{bx}]}{J_x}, \\ \ddot{\theta} = \frac{[\dot{\phi}\dot{\varphi}(J_z - J_x) - J_r \dot{\phi}\dot{\omega} - d_\theta \dot{\theta} + M_{by}]}{J_y}, \\ \ddot{\varphi} = \frac{[\dot{\phi}\dot{\theta}(J_x - J_y) - d_\varphi \dot{\varphi} + M_{bz}]}{J_z}, \end{cases} \quad (7)$$

where $\dot{\omega} = \Omega_4 + \Omega_3 - \Omega_2 - \Omega_1$ can be got easily online. It should be noted that to make the roll and pitch angles physically meaningful, they are both limited to $(-\pi/2, \pi/2)$. In particular, the yaw angle is also limited to $(-\pi/2, \pi/2)$ in this study, while $P = [x, y, z]^T \in R^3$ is the position with respect to the inertial frame. The translational dynamic equations of the quadrotor are given as

$$m\ddot{P} = R_t \cdot F + \begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} - \begin{bmatrix} d_x \dot{x} \\ d_y \dot{y} \\ d_z \dot{z} \end{bmatrix}, \quad (8)$$

where d_x , d_y , and d_z are the air drag coefficients which are added in (8) to model the drag force caused by translational motions; F is the lift force generated by rotors with respect to the body-fixed frame.

$$F = \begin{bmatrix} 0 \\ 0 \\ k(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \end{bmatrix}. \quad (9)$$

By combining (7) and (8), a compact affine nonlinear equation of the quadrotor UAV is given as

$$\dot{X} = f(X) + g(X)U, \quad (10)$$

where $X = [x, \dot{x}, y, \dot{y}, z, \dot{z}, \phi, \dot{\phi}, \theta, \dot{\theta}, \varphi, \dot{\varphi}]^T \in R^{12}$ is the state variable; $f(X)$ and $g(X)$ are smooth functions on X . Equation (10) is expanded as follows:

$$\begin{cases} \dot{x}_1 = x_2, \\ \dot{x}_2 = (C_{x7} S_{x9} C_{x11} + S_{x7} S_{x11})U_1 - a_1 x_2 + d_1, \\ \dot{x}_3 = x_4, \\ \dot{x}_4 = (C_{x7} S_{x9} S_{x11} - S_{x7} S_{x11})U_1 - a_2 x_4 + d_2, \\ \dot{x}_5 = x_6, \\ \dot{x}_6 = (C_{x7} C_{x9})U_1 - g - a_3 x_6 + d_3, \\ \dot{x}_7 = x_8, \\ \dot{x}_8 = a_4 x_{10} x_{12} + a_5 \dot{\omega} x_{10} - a_6 x_8 + U_2 + d_4, \\ \dot{x}_9 = x_{10}, \\ \dot{x}_{10} = a_7 x_8 x_{12} + a_8 \dot{\omega} x_8 - a_9 x_{10} + U_3 + d_5, \\ \dot{x}_{11} = x_{12}, \\ \dot{x}_{12} = a_{10} x_8 x_{10} - a_{11} x_{12} + U_4 + d_6, \end{cases} \quad (11)$$

where U_i , $i = 1, 2, 3, 4$ represents the control inputs defined as follows:

$$\begin{aligned} U_1 &= \frac{l_k(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2)}{m}, \\ U_2 &= \frac{l_k(\Omega_4^2 - \Omega_2^2)}{J_x}, \\ U_3 &= \frac{l_k(\Omega_3^2 - \Omega_1^2)}{J_y}, \\ U_4 &= \frac{l_k(\Omega_4^2 + \Omega_2^2 - \Omega_3^2 - \Omega_1^2)}{J_z}, \end{aligned} \quad (12)$$

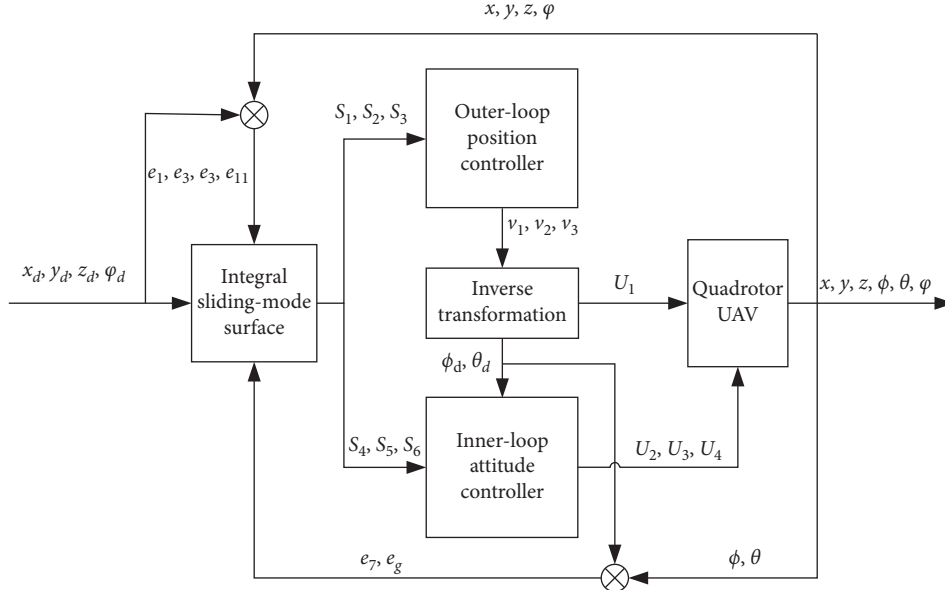


FIGURE 2: Schematic diagram of the proposed control scheme.

and a_i , $i = 1, 2, \dots, 11$, are the normalized parameters defined as follows:

$$\begin{aligned}
 a_1 &= \frac{d_x}{m}, \\
 a_2 &= \frac{d_y}{m}, \\
 a_3 &= \frac{d_z}{m}, \\
 a_4 &= \frac{J_y - J_z}{J_x}, \\
 a_5 &= \frac{J_r}{J_x}, \\
 a_6 &= \frac{d_\phi}{J_x}, \\
 a_7 &= \frac{J_z - J_x}{J_y}, \\
 a_8 &= \frac{J_r}{J_y}, \\
 a_9 &= \frac{d_\theta}{J_y}, \\
 a_{10} &= \frac{J_x - J_y}{J_z}, \\
 a_{11} &= \frac{d_\psi}{J_z}.
 \end{aligned} \tag{13}$$

2.2. Fuzzy Logic Systems (FLSs). In this study, the Fuzzy Logic Systems (FLSs) are introduced to approximate the continuous unknown functions on a given compact set. The FLSs consist of three main parts: fuzzy rule base, fuzzification, and defuzzification operators. The form of the fuzzy rules of the fuzzy controller is

Rule l : If x_1 is F_1^l and x_2 is F_2^l and \dots and x_n is F_n^l .

Then y is G^l , $l = 1, 2, \dots, N$, where $x(t) = [x_1, x_2, \dots, x_n]^T$ and y are the input and output of the whole fuzzy system, respectively. And N is the number of the rules. The fuzzy basis functions forms are defined as

$$y(x) = \frac{\sum_{l=1}^N \bar{y}^l \left(\prod_{i=1}^n \mu_{F_i^l}(x_i) \right)}{\sum_{l=1}^N \left(\prod_{i=1}^n \mu_{F_i^l}(x_i) \right)}, \tag{14}$$

where $\bar{y}^l = \max_{y \in R} \mu_{G^l}(y)$.

$$\xi_l(x) = \frac{\prod_{i=1}^n \mu_{F_i^l}(x_i)}{\sum_{l=1}^N \left(\prod_{i=1}^n \mu_{F_i^l}(x_i) \right)}. \tag{15}$$

Denoting $\alpha^T = [\bar{y}_1, \bar{y}_2, \dots, \bar{y}_N] = [\alpha_1, \alpha_2, \dots, \alpha_N]$ and $\xi(x) = [\xi_1(x), \xi_2(x), \dots, \xi_N(x)]^T$, then equation of the fuzzy system can be rewritten as

$$y(x) = \alpha^T \xi(x). \tag{16}$$

Lemma 1. For a continuous nonlinear function $f(x)$ in a compact set Ω_x , it can be effectively approximated by FLSs with any small approximated error $\varepsilon > 0$. $f(x)$ can be expressed as follows:

$$\sup_{x \in \Omega_x} |f(x) - \alpha^T \xi(x)| < \varepsilon, \tag{17}$$

TABLE 1: Position control algorithm of UAVs.

Step 1	
$e_1 = x_1 - x_{1d}$,	(T1.1)
$\bar{x}_2 = \dot{x}_{1d} - c_1 e_1$,	(T1.2)
$\tau_1 \dot{x}_{2d} + x_{2d} = \bar{x}_2$, $x_{2d}(0) = \bar{x}_2(0)$,	(T1.3)
$S_1 = x_2 - \int_0^t (\dot{x}_{2d} - k_1 \dot{e}_1 - k_2 e_1) dt$,	(T1.4)
$v_{1fs} = \alpha_1^T \xi_1$, $\hat{\alpha}_1 = -\eta_1 S_1 \xi_1$,	(T1.5)
$v_{1vs} = -\hat{E}_1 \text{sgn}(S_1)$, $\hat{E}_1 = \rho_1 S_1 $,	(T1.6)
Step 2	
$e_3 = x_3 - x_{3d}$,	(T1.7)
$\bar{x}_4 = \dot{x}_{3d} - c_2 e_3$,	(T1.8)
$\tau_2 \dot{x}_{4d} + x_{4d} = \bar{x}_4$, $x_{4d}(0) = \bar{x}_4(0)$,	(T1.9)
$S_2 = x_4 - \int_0^t (\dot{x}_{4d} - k_3 \dot{e}_3 - k_4 e_3) dt$,	(T1.10)
$v_{2fs} = \alpha_2^T \xi_2$, $\hat{\alpha}_2 = -\eta_2 S_2 \xi_2$,	(T1.11)
$v_{2vs} = -\hat{E}_2 \text{sgn}(S_2)$,	(T1.12)
$v_2 = v_{2fs} + v_{2vs}$.	(T1.12)
Step 3	
$e_5 = x_5 - x_{5d}$,	(T1.13)
$\bar{x}_6 = \dot{x}_{5d} - c_3 e_5$,	(T1.14)
$\tau_3 \dot{x}_{6d} + x_{6d} = \bar{x}_6$, $x_{6d}(0) = \bar{x}_6(0)$,	(T1.15)
$S_3 = x_6 - \int_0^t (\dot{x}_{6d} - k_5 \dot{e}_5 - k_6 e_5) dt$,	(T1.16)
$v_{3fs} = \alpha_3^T \xi_3$, $\hat{\alpha}_3 = -\eta_3 S_3 \xi_3$,	(T1.17)
$v_{3vs} = -\hat{E}_3 \text{sgn}(S_3)$,	(T1.17)
$v_3 = v_{3fs} + v_{3vs}$.	(T1.18)

if $x \in \Omega_x$, then the smooth nonlinear function $f(x)$ can be expressed as

$$f(x) = \alpha^{*T} \xi(x) + \varepsilon(x), \quad (18)$$

where α^* is the optimal fuzzy parameter vector, $\varepsilon(x)$ is the approximation error satisfies $\|\varepsilon(x)\| \leq \bar{\varepsilon} > 0$.

3. DSISM Controller Design Procedure

In this section, the design process of dynamic surface integral sliding mode controller is proposed and the control system block diagram is shown in Figure 2. The controller design process is divided into two parts: the position tracking controller design and attitude tracking controller design, as shown in Tables 1 and 2, respectively. To make the presentation clear, the specific controller design process is given in Appendix A.

In Table 1, e_i , ($i = 1, 2, 3$) are the tracking error, and \bar{x}_i , ($i = 2, 4, 6$) are the virtual laws. The low-pass first-order filters (T1.3), (T1.9), and (T1.15) are presented in each step to get a new variable x_{id} , ($i = 2, 4, 6$) with the time constants τ_i , ($i = 1, 2, 3$). The integral sliding mode surfaces S_i , ($i = 1, 2, 3$) are selected in each step which improves the robustness of the system against disturbances and parameters uncertainties. Due to the existence of unknown functions and parameters, the fuzzy system v_{ifs} , ($i = 1, 2, 3$) are utilized to approximate v_i , ($i = 1, 2, 3$) with α_i , ($i = 1, 2, 3$) and ξ_i , ($i = 1, 2, 3$) being the adjustable parameters and fuzzy basis vectors, respectively. Then, the switching control law v_{ivs} , ($i = 1, 2, 3$) are introduced to compensate the error of v_i , ($i = 1, 2, 3$) and the ideal input. E_i , ($i = 1, 2, 3$) are the switching gain, c_i , η_i , ρ_i , ($i = 1, 2, 3$), and k_i , ($i = 1, 2, \dots, 6$)

TABLE 2: Attitude control algorithm of UAVs.

Step 4	
$e_7 = x_7 - x_{7d}$,	(T2.1)
$\bar{x}_8 = \dot{x}_{7d} - c_4 e_7$,	(T2.2)
$\tau_4 \dot{x}_{8d} + x_{8d} = \bar{x}_8$, $x_{8d}(0) = \bar{x}_8(0)$,	(T2.3)
$S_4 = x_8 - \int_0^t (\dot{x}_{8d} - k_7 \dot{e}_7 - k_8 e_7) dt$,	(T2.4)
$U_{2fs} = \alpha_4^T \xi_4$, $\hat{\alpha}_4 = -\eta_4 S_4 \xi_4$,	(T2.5)
$U_{2vs} = -\hat{E}_4 \text{sgn}(S_4)$, $\hat{E}_4 = \rho_4 S_4 $,	(T2.6)
$U_2 = U_{2fs} + U_{2vs}$.	(T2.6)
Step 5	
$e_9 = x_9 - x_{9d}$,	(T2.7)
$\bar{x}_{10} = \dot{x}_{9d} - c_5 e_9$,	(T2.8)
$\tau_5 \dot{x}_{10d} + x_{10d} = \bar{x}_{10}$, $x_{10d}(0) = \bar{x}_{10}(0)$,	(T2.9)
$S_5 = x_{10} - \int_0^t (\dot{x}_{10d} - k_9 \dot{e}_9 - k_{10} e_9) dt$,	(T2.10)
$U_{3fs} = \alpha_5^T \xi_5$, $\hat{\alpha}_5 = -\eta_5 S_5 \xi_5$,	(T2.11)
$U_{3vs} = -\hat{E}_5 \text{sgn}(S_5)$, $\hat{E}_5 = \rho_5 S_5 $,	(T2.12)
$U_3 = U_{3fs} + U_{3vs}$.	(T2.12)
Step 6	
$e_{11} = x_{11} - x_{11d}$,	(T2.13)
$\bar{x}_{12} = \dot{x}_{11d} - c_6 e_{11}$,	(T2.14)
$\tau_6 \dot{x}_{12d} + x_{12d} = \bar{x}_{12}$, $x_{12d}(0) = \bar{x}_{12}(0)$,	(T2.15)
$S_6 = x_{12} - \int_0^t (\dot{x}_{12d} - k_{11} \dot{e}_{11} - k_{12} e_{11}) dt$,	(T2.16)
$U_{4fs} = \alpha_6^T \xi_6$, $\hat{\alpha}_6 = -\eta_6 S_6 \xi_6$,	(T2.17)
$U_{4vs} = -\hat{E}_6 \text{sgn}(S_6)$, $\hat{E}_6 = \rho_6 S_6 $,	(T2.18)
$U_4 = U_{4fs} + U_{4vs}$.	(T2.18)

TABLE 3: Quadrotor parameters.

Symbol	Case1	Case 2	Case 3	Case 4	Units
m	2	2	2	2	kg
l	0.2	0.2	0.2	0.2	m
κ	2.98	2.98	2.98	2.98	$10^{-6} \text{ N} \cdot \text{s}^2 \cdot \text{rad}^{-2}$
τ	1.14	1.14	1.14	1.14	$10^{-7} \text{ N} \cdot \text{s}^2 \cdot \text{rad}^{-2}$
d_ϕ	1.2	1.2	1.2	1.2	$10^{-2} \text{ N} \cdot \text{s} \cdot \text{rad}^{-1}$
$d_\theta d_\phi$	1.2	1.2	1.2	1.2	$10^{-2} \text{ N} \cdot \text{s} \cdot \text{rad}^{-1}$
J_x	1.25	1.25	1.25	1.25	$\text{N} \cdot \text{s}^2 \cdot \text{rad}^{-1}$
J_y	1.25	1.25	1.25	1.25	$\text{N} \cdot \text{s}^2 \cdot \text{rad}^{-1}$
J_z	2.50	2.88	3.25	3.75	$\text{N} \cdot \text{s}^2 \cdot \text{rad}^{-1}$

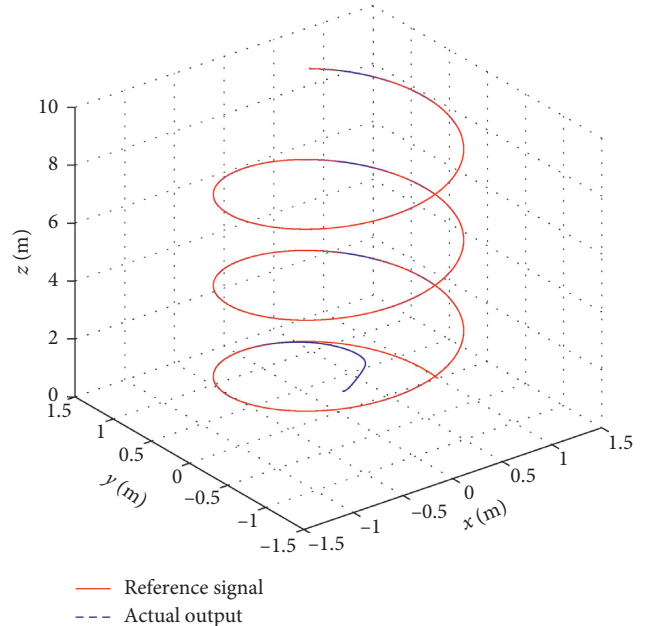


FIGURE 3: Space diagram of position in normal case.

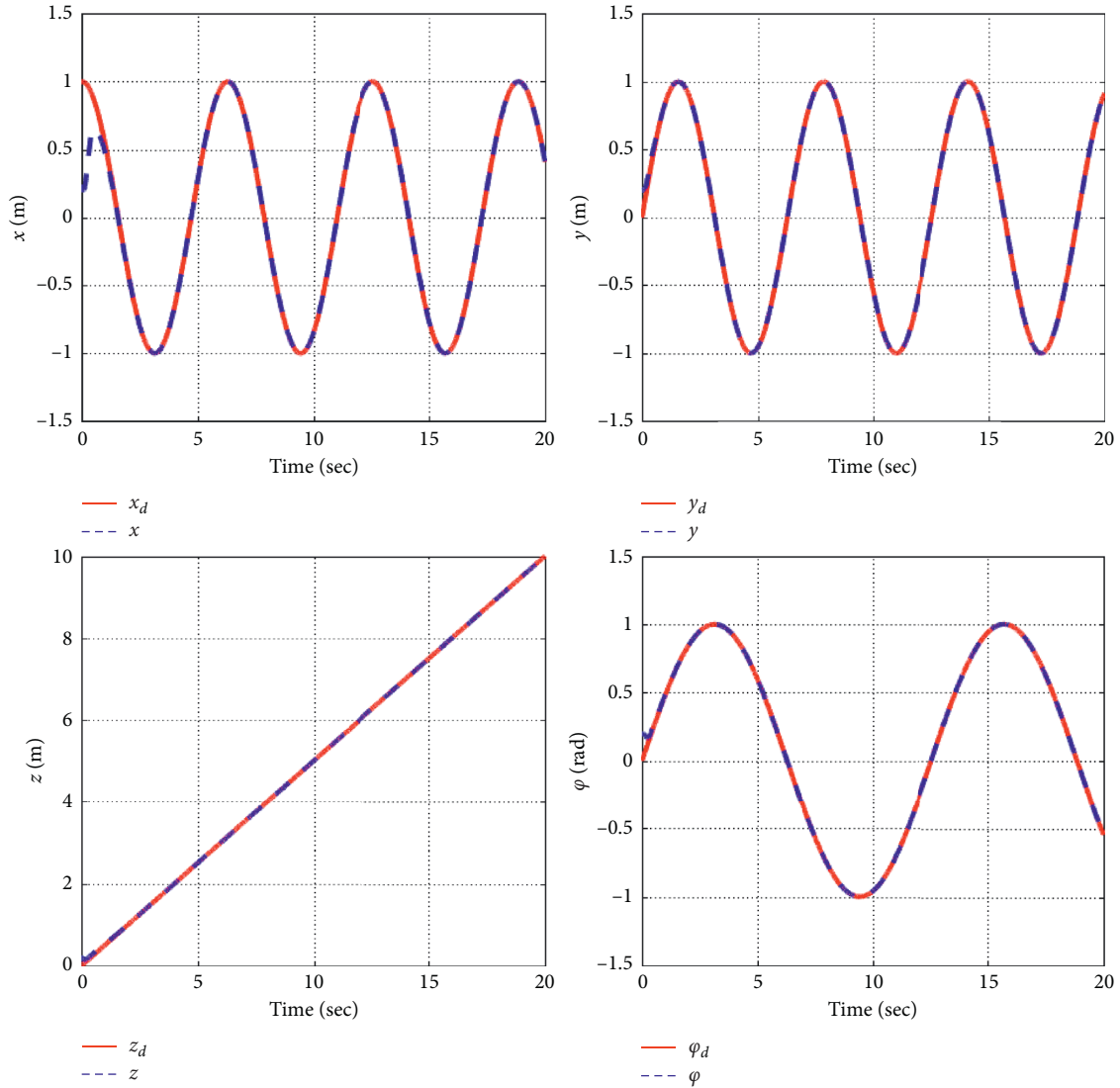


FIGURE 4: Reference signal and the actual outputs.

are positive constant. It is worth noting that $v = [v_1, v_2, v_3]^T$ is a group of virtual control laws and they can be given by $v_1 = (C_{x7}S_{x9}C_{x11} + S_{x7}S_{x11})U_1$, $v_2 = (C_{x7}S_{x9}S_{x11} - S_{x7}S_{x11})U_1$, $v_3 = (C_{x7}C_{x9})U_1$.

In Table 2, e_i , ($i = 4, 5, 6$) are the tracking error, and \bar{x}_i , ($i = 8, 10, 12$) are the virtual laws. The low-pass first-order filters (T2.3), (T2.9), and (T2.15) are presented in each step to get a new variable x_{id} , ($i = 8, 10, 12$) with the time constants τ_i , ($i = 4, 5, 6$). The integral sliding mode surfaces

S_i , ($i = 4, 5, 6$) are selected in each step to improve the robustness of the system. The fuzzy system U_{ifs} , ($i = 2, 3, 4$) are utilized to approximate U_i , ($i = 2, 3, 4$) with α_i , ($i = 4, 5, 6$) and ξ_i , ($i = 4, 5, 6$) being adjustable parameters and fuzzy basis vectors, respectively. Then, the switching control law U_{ivs} , ($i = 2, 3, 4$) are introduced to compensate for U_i , ($i = 2, 3, 4$) and the ideal input. E_i , ($i = 4, 5, 6$) are the switching gain. c_i , η_i , ρ_i , ($i = 4, 5, 6$), and k_i , ($i = 7, 8, \dots, 12$) are positive constant.

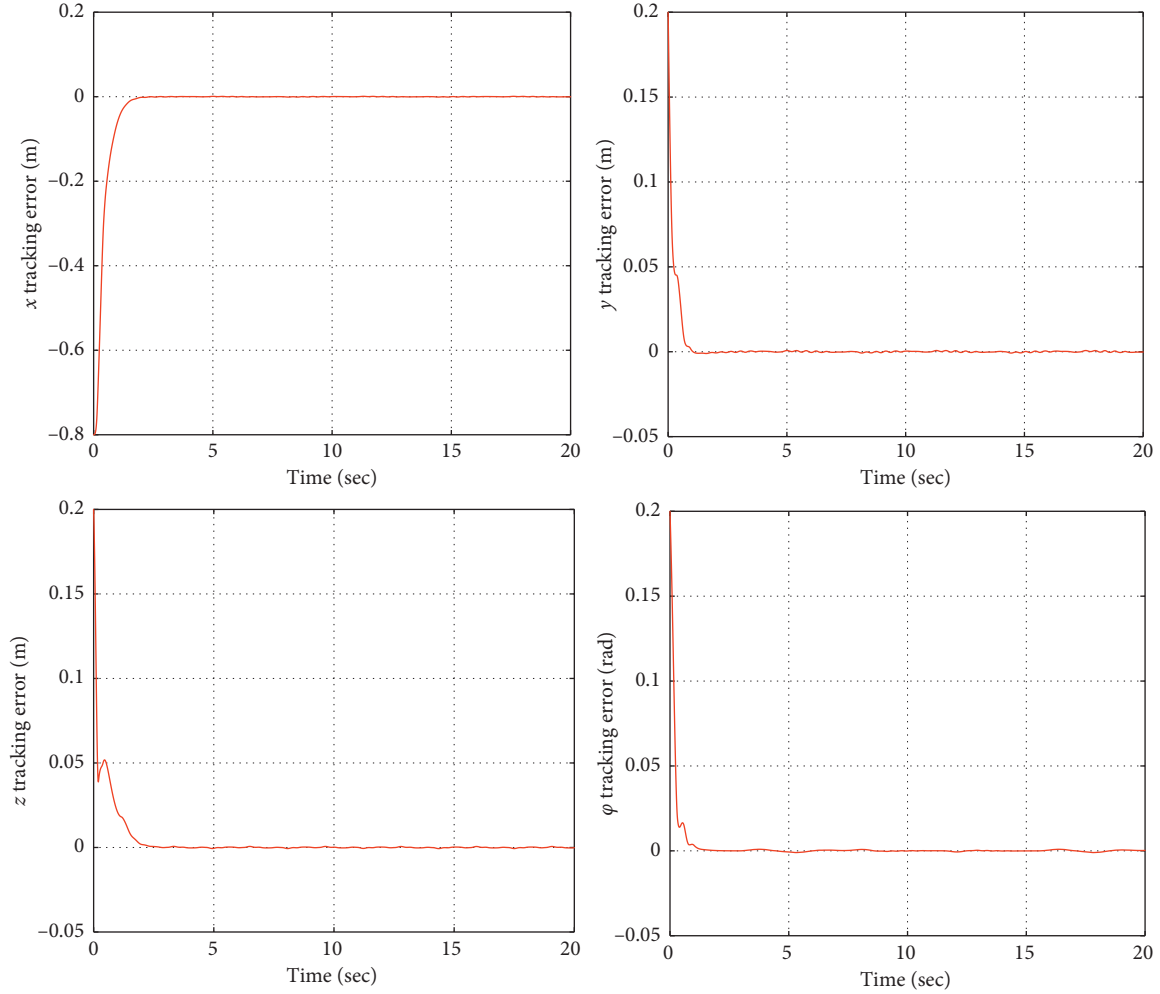


FIGURE 5: The actual tracking error.

Remark 1. v_1 , v_2 , and v_3 are combinations of available terms that can be given directly or measured in Tables 1 and 2. Therefore, the control input U_1 can be solved by $U_1 = v_3/C_{X_7}C_{X_9}$.

4. Stability Analysis of the Closed-Loop System

In this section, stability analysis of the proposed control system is established to confirm that all signals in the closed loop are ultimately bounded. The errors of the first-order filter are presented as follows:

$$y_i = x_{id} - \bar{x}_i, \quad (i = 2, 4, \dots, 12). \quad (19)$$

From (T1.3), (T1.9), (T1.15), (T2.3), (T2.9), and (T2.15), one can obtain

$$\dot{x}_{id} = -\frac{y_i}{\tau_{i/2}}, \quad (i = 2, 4, \dots, 12). \quad (20)$$

The derivative of y_i ($i = 2$) in time can be obtained as

$$\dot{y}_2 = \dot{x}_{2d} - \dot{\bar{x}}_2 = -\frac{y_2}{\tau_1} - \ddot{x}_{2d} + c_1 \dot{e}_1, \quad (21)$$

then, one can obtain

$$\dot{y}_2 = -\frac{y_2}{\tau_1} + B_2(e_1, e_2, y_2, \ddot{x}_{2d}), \quad (22)$$

where $B_2(e_1, e_2, y_2, \ddot{x}_{2d}) = -\ddot{x}_{2d} + c_1 \dot{e}_1$ is a continuous function. The following formula can also be obtained:

$$\dot{y}_i = -\frac{y_i}{\tau_{i/2}} + B_i(\cdot), \quad (i = 2, 4, 6, 8, 10, 12). \quad (23)$$

Then, the following inequalities hold:

$$y_i \dot{y}_i \leq -\frac{y_i^2}{\tau_{i/2}} + B_i |y_i|, \quad (i = 2, 4, 6, 8, 10, 12). \quad (24)$$

Consider the Lyapunov function candidate

$$V = V_1 + V_2, \quad (25)$$

with $V_1 = (1/2) \sum_{i=1}^6 (e_{2i-1}^2 + y_{2i}^2)$ and $V_2 = (1/2) \sum_{i=1}^6 (S_i^2 + (1/\eta_i) \tilde{\alpha}_i^T \tilde{\alpha}_i + (1/\rho_i) \tilde{E}_i^2)$. Then, the following theorem can be obtained.

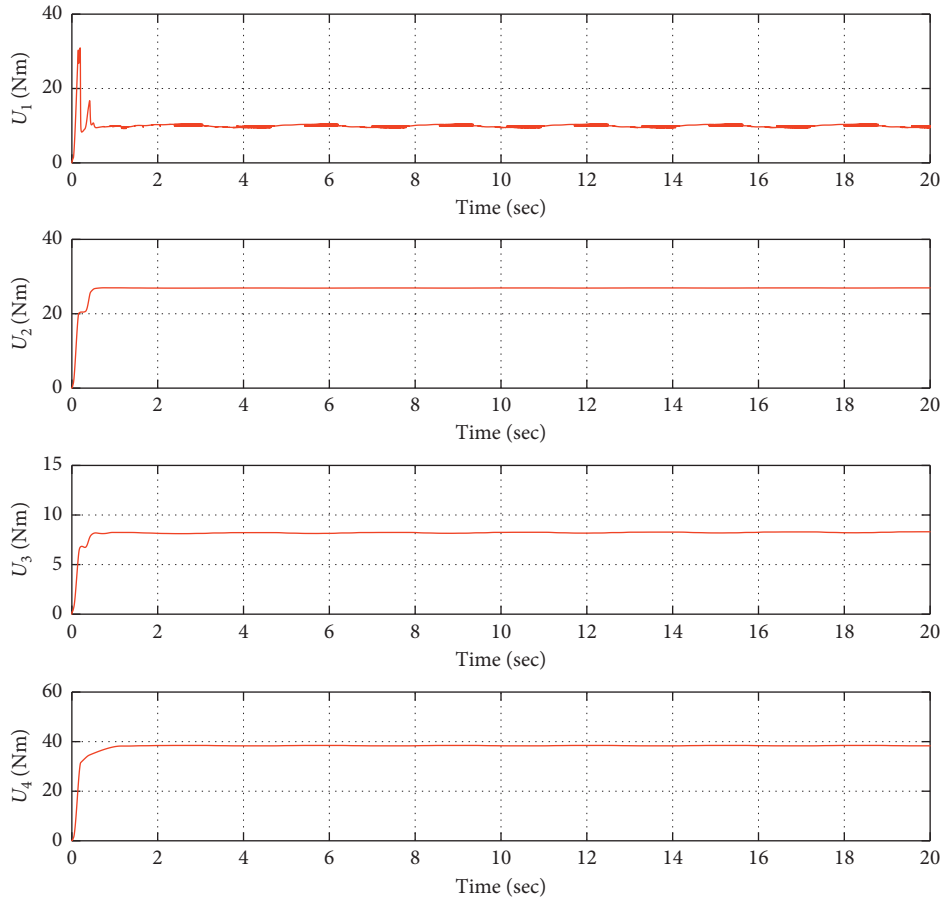


FIGURE 6: Control inputs.

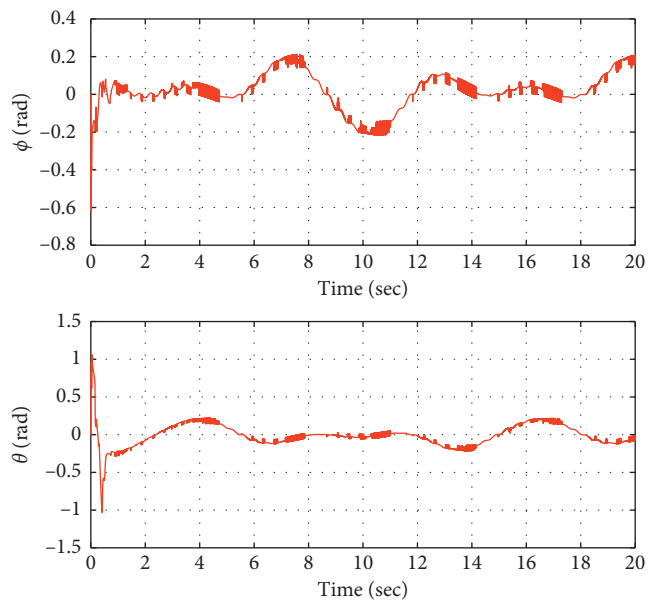


FIGURE 7: Change of roll and pitch.

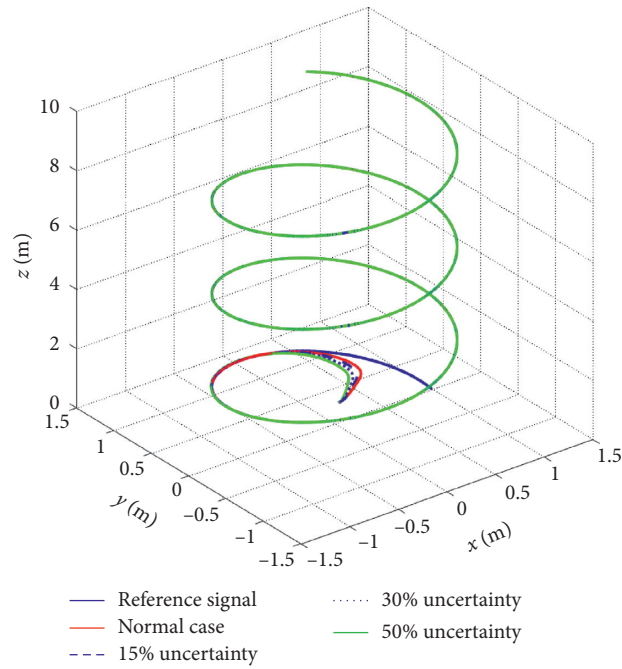


FIGURE 8: Space diagram of position of different uncertainty cases added in yaw axis.

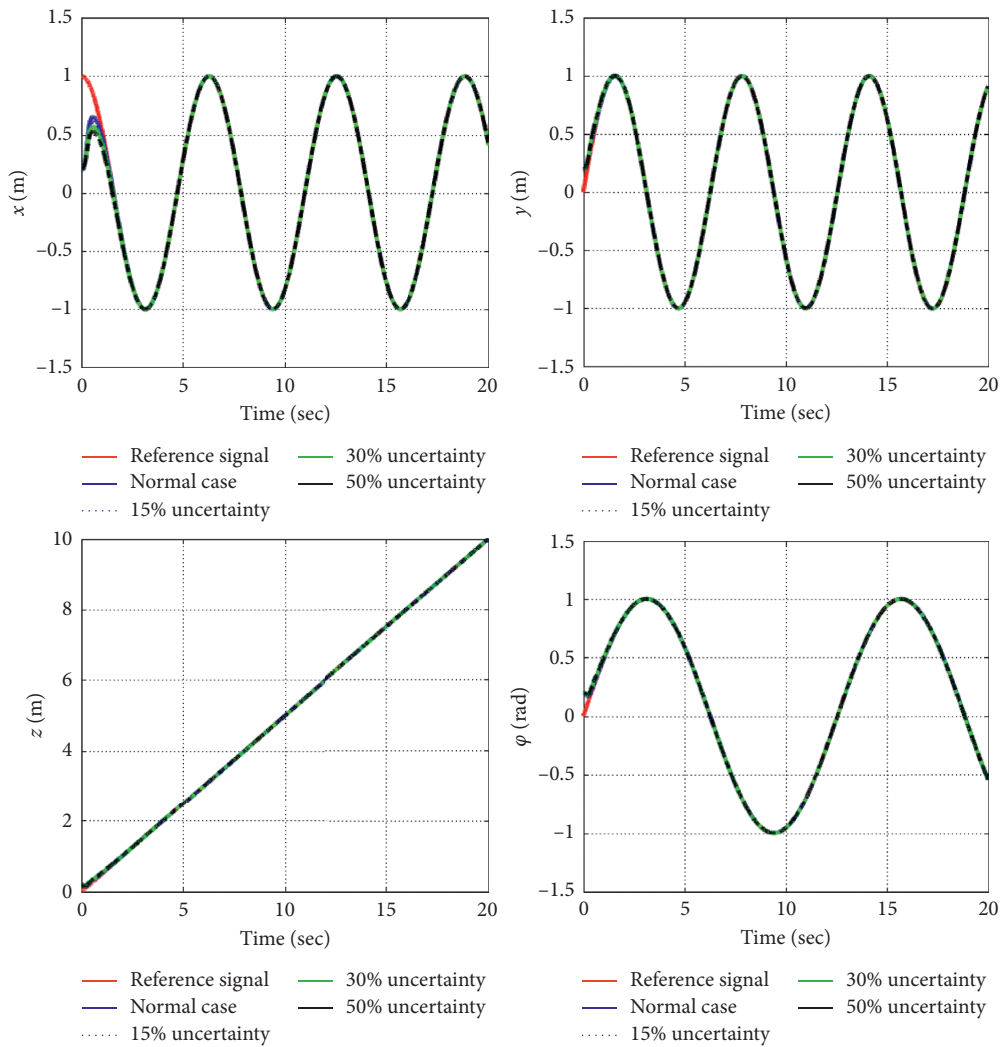


FIGURE 9: Reference signal and the actual outputs of different uncertainty cases added in yaw axis.

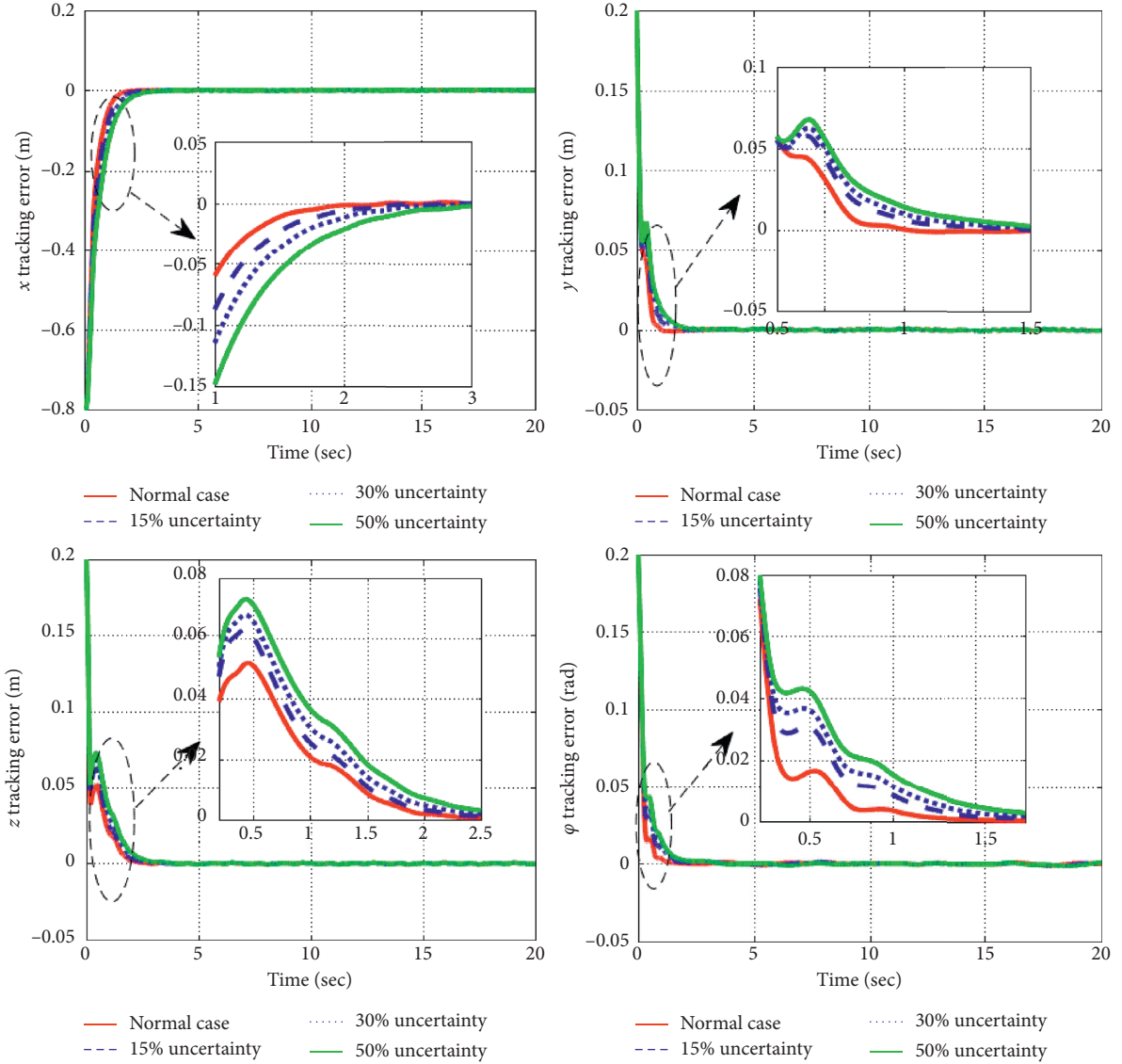


FIGURE 10: The actual tracking error of different uncertainty cases added in yaw axis.

Theorem 1. Consider the closed-loop system which consists of the position and attitude system, with the virtual control signals (T1.2), (T1.8), (T1.14), (T2.2), (T2.8), and (T2.14), the adaptive laws (T1.5), (T1.11), (T1.17), (T2.5), (T2.11), and (T2.17), and the switching control laws (T1.6), (T1.12), (T1.18), (T2.6), (T2.12), and (T2.18). If all the parameters c_i , η_i , ρ_i , ($i = 1, 2, \dots, 6$), k_i , ($i = 1, 2, \dots, 12$) and the time constant of first-order filter τ_i , ($i = 1, 2, \dots, 6$) are designed properly to satisfy $V(0) \leq p$, ($p > 0$), all the closed-loop signals are uniformly bounded and the tracking error can be kept arbitrarily small.

Proof. The specific proof process is presented in Appendix B. \square

5. Simulations

In this section, the following simulations are given to validate the effectiveness and the performance of the proposed

adaptive dynamic surface integral sliding mode control. The parameters for the quadrotor UAV adopted in this paper are presented in Table 3. In the following simulation, the desired trajectory of the position and yaw angle $\{x(t), y(t), z(t), \varphi(t)\}$ are chosen as $\{\sin(t), \cos(t), 0.5t, \sin(0.5t)\}$. The controller parameters chosen for simulation are $c_i = 0.01$, $\tau_i = 0.001$, $\eta_i = 200$, $\rho_i = 0.1$, ($i = 1, 2, \dots, 6$), $k_i = 9$, ($i = 1, 3, 5, 7, 9, 11$), and $k_i = 20$, ($i = 2, 4, 6, 8, 10, 12$). The fuzzy membership functions are chosen as follows: $u_1(s_i) = \exp(-[(s_i + (\pi/6) - (l-1) \times (\pi/12)) / (\pi/24)]^2)$, ($i = 1, 2, 3; l = 1, 2, 3, 4, 5, 6$). The disturbances are chosen as follows: $d_1 = \cos(t)$, $d_2 = \sin(t)$, $d_3 = \sin(t)\cos(t)$, $d_4 = 0.5 \sin(0.5t)$, $d_5 = 0.5 \cos(0.5t)$, $d_6 = 0.25 \sin(0.5t)\cos(0.5t)$.

Case 1. In this case, the parameters of the quadrotor are assumed normal. The simulations for this case are presented.

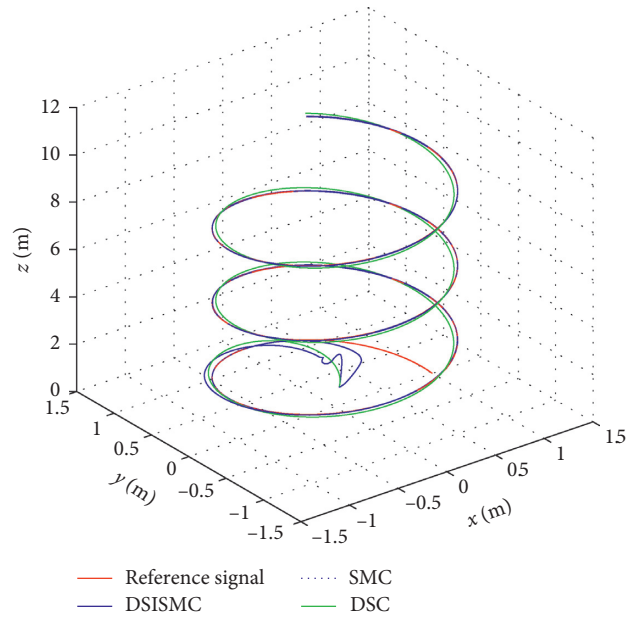


FIGURE 11: Space diagram of position by using DSISMC, SMC, and DSC.

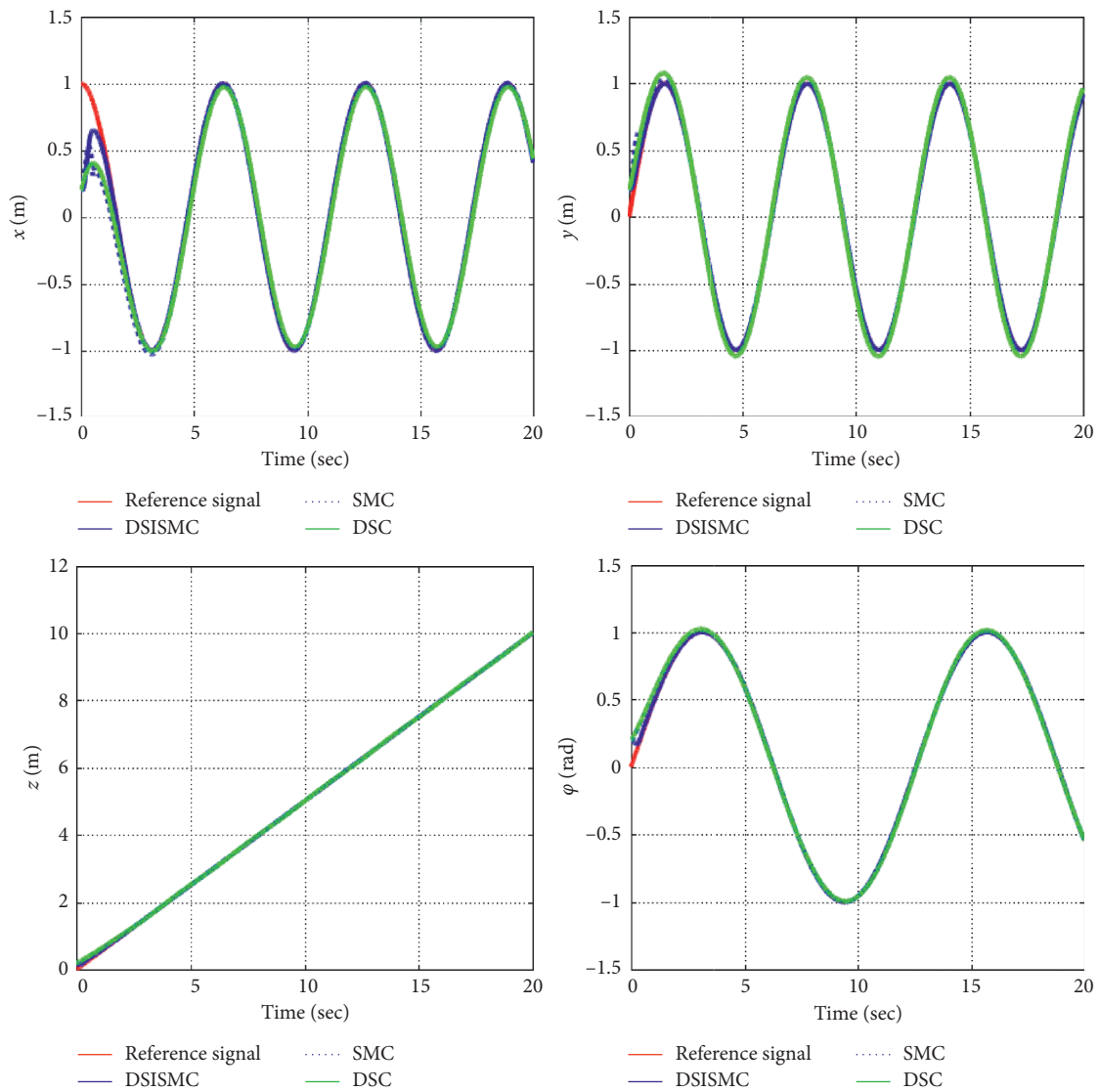


FIGURE 12: Reference signal and the actual outputs by using DSISMC, SMC and DSC.

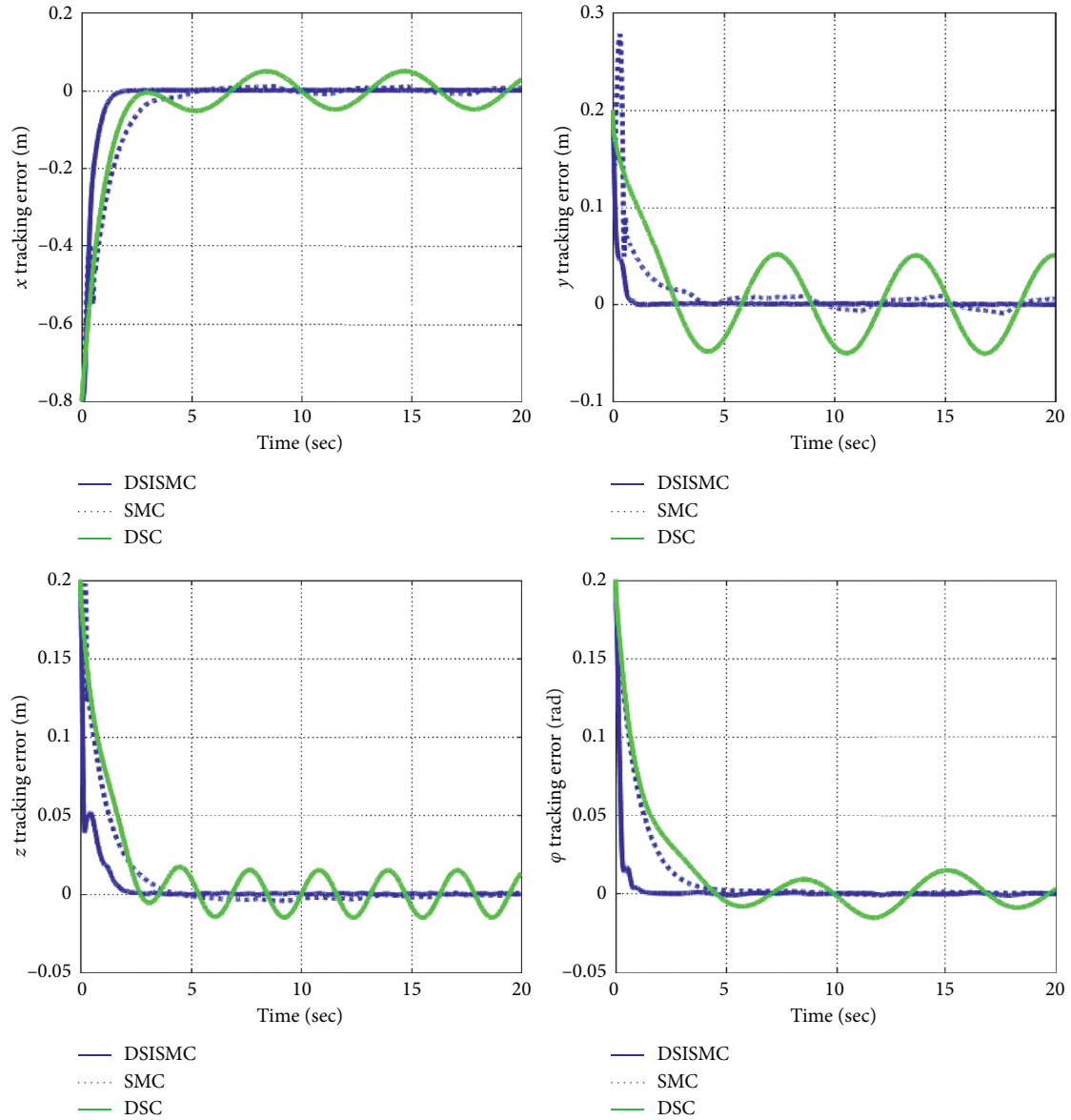


FIGURE 13: The actual tracking error by using DSISMC, SMC, DSC.

TABLE 4: The MVTE and RMSVTE of different schemes.

Kind of errors	Proposed scheme	ADSC scheme	ASMC scheme
MVTE of x (m)	$7.66e^{-4}$	$4.90e^{-2}$	$9.76e^{-3}$
MVTE of y (m)	$6.58e^{-4}$	$5.03e^{-2}$	$7.15e^{-3}$
MVTE of z (m)	$5.70e^{-4}$	$1.50e^{-2}$	$3.15e^{-3}$
MVTE of φ (rad)	$4.83e^{-4}$	$1.53e^{-2}$	$7.84e^{-4}$
RMSVTE of x (m)	$3.33e^{-4}$	$3.81e^{-2}$	$7.02e^{-3}$
RMSVTE of y (m)	$2.92e^{-4}$	$4.03e^{-2}$	$5.16e^{-3}$
RMSVTE of z (m)	$2.16e^{-4}$	$9.82e^{-3}$	$2.73e^{-3}$
RMSVTE of φ (rad)	$1.34e^{-4}$	$1.16e^{-2}$	$5.04e^{-4}$

Cases 2–4. Uncertainty (15%, 30%, and 50% added) in the rotary inertia as well as the presence of external disturbance. In these cases, we give a consideration to both external disturbance and uncertainty in rotary inertia. The uncertainty in yaw axis is 15%, 30%, and 50%, respectively.

The simulation results of Case 1 are presented from Figures 3 to 7. Figure 3 shows the 3D tracking trajectory by using the proposed controller. The position and yaw angle trajectories are shown in Figure 4, and the tracking errors are given in Figure 5. Figure 6 shows the control signals. Figure 7 shows the change of roll and pitch angles. From the tracking performance, it can be concluded that the proposed control scheme can guarantee that all the variables are bounded and that the control system has strong robustness. Figures 8 to 10 illustrate the tracking performance under the different uncertainty cases. The proposed scheme has the better robustness against external disturbances and uncertainty parameters. Figures 11 to 13 show the tracking error comparisons between the ADSIMC, the ADSC, and the ASMC methods. Meanwhile, the maximum values (MVTE) and the root mean square values (RMSVTE) of tracking error in steady of the proposed scheme and the other two schemes are given in Table 4. The simulation results show that the proposed scheme has better tracking performance and robustness compared with the ADSC and the ASMC methods.

6. Conclusion

This paper proposed a dynamic surface integral sliding mode control scheme for a quadrotor UAV under the conditions of parameter uncertainty and external disturbances. Virtual control inputs are introduced in the robust controller design to guarantee the trajectory tracking performance, and the problem of “explosion of complexity” in the backstepping design has been greatly simplified. The fuzzy systems are utilized to approximate the ideal control inputs and the switch control is introduced to compensate for errors between estimated and ideal inputs which improves the control performance and robustness of the whole system. In addition, the stability analysis of the overall system through Lyapunov stability theory is presented, and all signals of the closed loop are ultimately bounded. Finally, the simulation results show that robustness and improved tracking performance can be achieved with the proposed control scheme.

Appendix

A. The Controller Design Procedures

Step 1. Define the position error:

$$e_1 = x_1 - x_{1d}, \quad (\text{A.1})$$

where x_{1d} is the desired x position command, and the derivative of e_1 with respect to time is

$$\dot{e}_1 = \dot{x}_1 - \dot{x}_{1d}. \quad (\text{A.2})$$

Define the virtual control \bar{x}_2 ,

$$\bar{x}_2 = -c_1 e_1 + \dot{x}_{1d}, \quad (\text{A.3})$$

where c_1 is a positive constant. To solve the problem of “complexity of explosion” caused by the repeated differentiation, a new state variable x_{2d} is introduced and let \bar{x}_2 pass through the following first-order filter with constant τ_1 (T1.3) to obtain x_{2d}

$$\tau_1 \dot{x}_{2d} + x_{2d} = \bar{x}_2, x_{2d}(0) = \bar{x}_2(0), \quad (\text{A.4})$$

where x_{2d} is the output of the first-order filter; the filter error is $y_1 = x_{2d} - \bar{x}_2$. A proper integral sliding mode manifold is chosen (T1.4):

$$S_1 = x_2 - \int_0^t (\dot{x}_{2d} - k_1 \dot{e}_1 - k_2 e_1) dt, \quad (\text{A.5})$$

where k_1 and k_2 are both the positive constant. If the sliding mode control is in an ideal state, the derivative of S_1 with respect to time is

$$S_1 = \dot{S}_1 = \dot{x}_2 - \dot{x}_{2d} + k_1 \dot{e}_1 + k_2 e_1 = \dot{e}_2 + k_1 \dot{e}_1 + k_2 e_1 = 0. \quad (\text{A.6})$$

Then, a variable $v_1 = (C_{x7} S_{x9} C_{x11} + S_{x7} S_{x11}) U_1$ is introduced to be a new control input; then

$$v_1 - a_1 x_2 + d_1 - \dot{x}_{2d} + k_1 \dot{e}_1 + k_2 e_1 = \dot{e}_2 + k_1 \dot{e}_1 + k_2 e_1 = 0. \quad (\text{A.7})$$

Assuming that the perturbations and parameters in the equation are known, the control law for x motion in an ideal state is designed as follows:

$$v_1^* = a_1 x_2 - d_1 + \dot{x}_{2d} - k_1 \dot{e}_1 - k_2 e_1. \quad (\text{A.8})$$

In most cases, the system parameters are uncertain and there are also unknown external disturbances, which makes it difficult to obtain the ideal control signal v_1^* . Therefore, the fuzzy system is used to approximate the ideal control signal v_1^* and obtain

$$v_1^* = v_{1fz}(s_1, \alpha_1) + \varepsilon_1 = \alpha_1^T \xi_1 + \varepsilon_1, \quad (\text{A.9})$$

where ε_1 is the approximation error, and $|\varepsilon_1| < E_1$. Introducing switching control law v_{1vs} (T1.5) to compensate for v_1^* and v_{1fz} ,

$$v_{1vs} = -\hat{E}_1 \text{sgn}(S_1), \quad (\text{A.10})$$

where \hat{E}_1 is the estimation of E_1 .

Then, the actual control law (T1.6) is obtained as

$$v_1 = v_{1fz} + v_{1vs}. \quad (\text{A.11})$$

Consider the Lyapunov function

$$\Gamma_1 = \frac{1}{2} S_1^2 + \frac{1}{2\eta_1} \tilde{\alpha}_1^T \tilde{\alpha}_1 + \frac{1}{2\rho_1} \tilde{E}_1^2, \quad (\text{A.12})$$

where η_1 and ρ_1 are positive constant. The derivative of Γ_1 with respect to time can be presented below:

$$\dot{\Gamma}_1 = S_1 \dot{S}_1 + \frac{1}{\eta_1} \tilde{\alpha}_1^T \dot{\tilde{\alpha}}_1 + \frac{1}{\rho_1} \tilde{E}_1 \dot{\tilde{E}}_1, \quad (\text{A.13})$$

while

$$\begin{aligned} v_1^* &= a_1 x_2 - d_1 + \dot{x}_{2d} - k_1 e_1 - k_2 e_1 \\ &= a_1 x_2 - d_1 + \dot{x}_2 - \dot{S}_1 \\ &= v_1 - \dot{S}_1. \end{aligned} \quad (\text{A.14})$$

So

$$\dot{S}_1 = v_1 - v_1^* = v_{1fz} + v_{1vs} - v_1^*. \quad (\text{A.15})$$

Substituting (A.15) into (A.13), then

$$\begin{aligned} \dot{\Gamma}_1 &= S_1 \dot{S}_1 + \frac{1}{\eta_1} \tilde{\alpha}_1^T \dot{\tilde{\alpha}}_1 + \frac{1}{\rho_1} \tilde{E}_1 \dot{\tilde{E}}_1 \\ &= S_1 (v_{1fz} + v_{1vs} - v_1^*) + \frac{1}{\eta_1} \tilde{\alpha}_1^T \dot{\tilde{\alpha}}_1 + \frac{1}{\rho_1} \tilde{E}_1 \dot{\tilde{E}}_1 \\ &= \tilde{\alpha}_1^T \left(S_1 \xi_1 + \frac{1}{\eta_1} \dot{\tilde{\alpha}}_1 \right) + S_1 (v_{1vs} - \varepsilon_1) + \frac{1}{\rho_1} \tilde{E}_1 \dot{\tilde{E}}_1. \end{aligned} \quad (\text{A.16})$$

The adaptive law and the switching control (T1.5) are chosen below:

$$\dot{\tilde{\alpha}}_1 = -\eta_1 S_1 \xi_1. \quad (\text{A.17})$$

And formula (A.16) becomes

$$\begin{aligned} \dot{\Gamma}_1 &= S_1 \dot{S}_1 + \frac{1}{\eta_1} \tilde{\alpha}_1^T \dot{\tilde{\alpha}}_1 + \frac{1}{\rho_1} \tilde{E}_1 \dot{\tilde{E}}_1 \\ &= \tilde{\alpha}_1^T \left(S_1 \xi_1 + \frac{1}{\eta_1} \dot{\tilde{\alpha}}_1 \right) + S_1 (v_{1vs} - \varepsilon_1) + \frac{1}{\rho_1} \tilde{E}_1 \dot{\tilde{E}}_1 \\ &= -\hat{E}_1 |S_1| - \varepsilon_1 S_1 + \frac{1}{\rho_1} (\hat{E}_1 - E_1) \dot{\tilde{E}}_1. \end{aligned} \quad (\text{A.18})$$

To make the Lyapunov function $\dot{\Gamma}_1 \leq 0$, the adaptive law of switching control is updated below:

$$\dot{\tilde{E}}_1 = \rho_1 |S_1|. \quad (\text{A.19})$$

Then,

$$\begin{aligned} \dot{\Gamma}_1 &= -\hat{E}_1 |S_1| - \varepsilon_1 S_1 + \frac{1}{\rho_1} (\hat{E}_1 - E_1) \dot{\tilde{E}}_1 \\ &\leq -E_1 |S_1| - \varepsilon_1 S_1 \\ &\leq -(E_1 - |\varepsilon_1|) |S_1| \\ &\leq 0. \end{aligned} \quad (\text{A.20})$$

The similar design processes are presented to design the trajectory tracking of y -axis position x_3 and z -axis position x_5 using the dynamic surface integral sliding mode control. Introduce the variables $v_2 = (C_{x7} S_{x9} S_{x11} - S_{x7} S_{x11}) U_1$ and $v_3 = (C_{x7} C_{x9}) U_1$, the specific procedures are presented in Step 2 and Step 3.

Step 2. Define the position error:

$$e_3 = x_3 - x_{3d}, \quad (\text{A.21})$$

where x_{3d} is the desired y -position command, and the derivative of e_3 with respect to time is

$$\dot{e}_3 = \dot{x}_3 - \dot{x}_{3d}. \quad (\text{A.22})$$

Define the virtual control \bar{x}_4 ,

$$\bar{x}_4 = -c_2 e_3 + \dot{x}_{3d}, \quad (\text{A.23})$$

where c_2 is a positive constant. A new state variable x_{4d} is introduced and let \bar{x}_4 pass through the following first-order filter with constant τ_2 (T1.9) to obtain x_{4d}

$$\tau_2 \dot{x}_{4d} + x_{4d} = \bar{x}_4, \quad x_{4d}(0) = \bar{x}_4(0), \quad (\text{A.24})$$

where x_{4d} is the output of the first-order filter; the filter error is $y_4 = x_{4d} - \bar{x}_4$. A proper integral sliding mode manifold is chosen (T1.10):

$$S_2 = x_4 - \int_0^t (\dot{x}_{4d} - k_3 \dot{e}_3 - k_4 e_3) dt, \quad (\text{A.25})$$

where k_3 and k_4 are the positive constant. If the sliding mode control is in an ideal state, and the derivative of S_2 with respect to time is

$$S_2 = \dot{S}_2 = \dot{x}_4 - \dot{x}_{4d} + k_3 \dot{e}_3 + k_4 e_3 = \dot{e}_4 + k_3 \dot{e}_3 + k_4 e_3 = 0. \quad (\text{A.26})$$

Then, a variable $v_2 = (C_{x7} S_{x9} S_{x11} - S_{x7} S_{x11}) U_1$ is introduced to be a new control input. Assuming that the perturbations and parameters in the equation are known, the control law for y -motion in an ideal state is designed as follows:

$$v_2^* = a_2 x_4 - d_2 + \dot{x}_{4d} - k_3 \dot{e}_3 - k_4 e_3. \quad (\text{A.27})$$

In most cases, the system parameters are uncertain and there are also unknown external disturbances, which makes it difficult to obtain the ideal control signal v_2^* . Therefore, the fuzzy system is used to approximate the ideal control signal v_2^* and obtain

$$v_2^* = v_{2fz}(s_2, \alpha_2) + \varepsilon_2 = \alpha_2^T \xi_2 + \varepsilon_2, \quad (\text{A.28})$$

where ε_2 is the approximation error, and $|\varepsilon_2| < E_2$. Introducing switching control law v_{2vs} to compensate for v_2^* and v_{2fz}

$$v_{2vs} = -\widehat{E}_2 \operatorname{sgn}(S_2), \quad (\text{A.29})$$

where \widehat{E}_2 is the estimation of E_2 . Then, the actual control law (T1.12) is obtained

$$v_2 = v_{2fz} + v_{2vs}. \quad (\text{A.30})$$

Consider the Lyapunov function

$$\Gamma_2 = \frac{1}{2}S_2^2 + \frac{1}{2\eta_2}\widetilde{\alpha}_2^T\widetilde{\alpha}_2 + \frac{1}{2\rho_2}\widetilde{E}_2^2, \quad (\text{A.31})$$

where η_2 and ρ_2 are positive constant. The derivative of Γ_2 with respect to time can be presented as below:

$$\dot{\Gamma}_2 = S_2\dot{S}_2 + \frac{1}{\eta_2}\widetilde{\alpha}_2^T\dot{\widetilde{\alpha}}_2 + \frac{1}{\rho_2}\widetilde{E}_2\dot{\widetilde{E}}_2, \quad (\text{A.32})$$

while

$$\begin{aligned} v_2^* &= a_2x_4 - d_2 + \dot{x}_{4d} - k_3\dot{e}_3 - k_4e_3 \\ &= a_2x_4 - d_2 + \dot{x}_4 - \dot{S}_2 \\ &= v_2 - \dot{S}_2. \end{aligned} \quad (\text{A.33})$$

So,

$$\dot{S}_2 = v_2 - v_2^* = v_{2fz} + v_{2vs} - v_2^*. \quad (\text{A.34})$$

Substituting (A.34) into (A.32), then

$$\begin{aligned} \dot{\Gamma}_2 &= S_2\dot{S}_2 + \frac{1}{\eta_2}\widetilde{\alpha}_2^T\dot{\widetilde{\alpha}}_2 + \frac{1}{\rho_2}\widetilde{E}_2\dot{\widetilde{E}}_2 \\ &= S_2(v_{2fz} + v_{2vs} - v_2^*) + \frac{1}{\eta_2}\widetilde{\alpha}_2^T\dot{\widetilde{\alpha}}_2 + \frac{1}{\rho_2}\widetilde{E}_2\dot{\widetilde{E}}_2 \\ &= \widetilde{\alpha}_2^T\left(S_2\xi_2 + \frac{1}{\eta_2}\dot{\widetilde{\alpha}}_2\right) + S_2(v_{2vs} - \varepsilon_2) + \frac{1}{\rho_2}\widetilde{E}_2\dot{\widetilde{E}}_2. \end{aligned} \quad (\text{A.35})$$

The adaptive law (T1.11) is chosen below:

$$\dot{\widetilde{\alpha}}_2 = -\eta_2 S_2 \xi_2. \quad (\text{A.36})$$

And formula (A.35) becomes

$$\begin{aligned} \dot{\Gamma}_2 &= S_2\dot{S}_2 + \frac{1}{\eta_2}\widetilde{\alpha}_2^T\dot{\widetilde{\alpha}}_2 + \frac{1}{\rho_2}\widetilde{E}_2\dot{\widetilde{E}}_2 \\ &= \widetilde{\alpha}_2^T\left(S_2\xi_2 + \frac{1}{\eta_2}\dot{\widetilde{\alpha}}_2\right) + S_2(v_{2vs} - \varepsilon_2) + \frac{1}{\rho_2}\widetilde{E}_2\dot{\widetilde{E}}_2 \\ &= -\widehat{E}_2|S_2| - \varepsilon_2 S_2 + \frac{1}{\rho_2}(\widehat{E}_2 - E_2)\dot{\widetilde{E}}_2. \end{aligned} \quad (\text{A.37})$$

To make the Lyapunov function $\dot{\Gamma}_2 \leq 0$, the adaptive law of switching control is updated below:

$$\dot{\widetilde{E}}_2 = \rho_2 |S_2|. \quad (\text{A.38})$$

Then,

$$\begin{aligned} \dot{\Gamma}_2 &= -\widehat{E}_2|S_2| - \varepsilon_2 S_2 + \frac{1}{\rho_2}(\widehat{E}_2 - E_2)\dot{\widetilde{E}}_2 \\ &\leq -E_2|S_2| - \varepsilon_2 S_2 \\ &\leq -(E_2 - |\varepsilon_2|)|S_2| \\ &\leq 0. \end{aligned} \quad (\text{A.39})$$

Step 3. Define the position error:

$$e_5 = x_5 - x_{5d}, \quad (\text{A.40})$$

where x_{5d} is the desired z -position command, and the derivative of e_5 with respect to time is

$$\dot{e}_5 = x_6 - \dot{x}_{5d}. \quad (\text{A.41})$$

Define the virtual control \bar{x}_6 ,

$$\bar{x}_6 = -c_3 e_5 + \dot{x}_{5d}, \quad (\text{A.42})$$

where c_3 is a positive constant. A new state variable x_{6d} is introduced and let \bar{x}_6 pass through the following first-order filter with constant τ_3 (T1.15) to obtain x_{6d}

$$\begin{aligned} \tau_3 \dot{x}_{6d} + x_{6d} &= \bar{x}_6, \\ x_{6d}(0) &= \bar{x}_6(0), \end{aligned} \quad (\text{A.43})$$

where x_{6d} is the output of the first-order filter; the filter error is $y_6 = x_{6d} - \bar{x}_6$. A proper integral sliding mode manifold is chosen (T1.16):

$$S_3 = x_6 - \int_0^t (\dot{x}_{6d} - k_5 \dot{e}_5 - k_6 e_5) dt, \quad (\text{A.44})$$

where k_5 and k_6 are the positive constant. If the sliding mode control is in an ideal state, and the derivative of S_3 with respect to time is

$$S_3 = \dot{S}_3 = \dot{x}_6 - \dot{x}_{6d} + k_5 \dot{e}_5 + k_6 e_5 = \dot{e}_6 + k_5 \dot{e}_5 + k_6 e_5 = 0. \quad (\text{A.45})$$

Then, a variable $v_3 = (C_{x7} C_{x9})U_1$ is introduced to be a new control input. Assuming that the perturbations and parameters in the equation are known, the control law for z motion in an ideal state is designed as follows:

$$v_3^* = a_3 x_6 - d_3 + g + \dot{x}_{6d} - k_5 \dot{e}_5 - k_6 e_5. \quad (\text{A.46})$$

In most cases, the system parameters are uncertain and there are also unknown external disturbances, which makes it difficult to obtain the ideal control signal v_3^* . Therefore, the fuzzy system is used to approximate the ideal control signal v_3^* and obtain

$$v_3^* = v_{3fz}(s_3, \alpha_3) + \varepsilon_3 = \alpha_3^T \xi_3 + \varepsilon_3, \quad (\text{A.47})$$

where ε_3 is the approximation error, and $|\varepsilon_3| < E_3$. Introducing switching control law v_{3vs} (T1.17) to compensate for v_3^* and v_{3fz}

$$v_{3vs} = -\widehat{E}_3 \operatorname{sgn}(S_3), \quad (\text{A.48})$$

where \widehat{E}_3 is the estimation of E_3 . Then, the actual control law (T1.18) is obtained as

$$v_3 = v_{3fz} + v_{3vs}. \quad (\text{A.49})$$

Consider the Lyapunov function

$$\Gamma_3 = \frac{1}{2}S_3^2 + \frac{1}{2\eta_3}\widetilde{\alpha}_3^T\widetilde{\alpha}_3 + \frac{1}{2\rho_3}\widetilde{E}_3^2, \quad (\text{A.50})$$

where η_3 and ρ_3 are the positive constant. The derivative of Γ_3 with respect to time can be presented as below:

$$\dot{\Gamma}_3 = S_3\dot{S}_3 + \frac{1}{\eta_3}\widetilde{\alpha}_3^T\dot{\widetilde{\alpha}}_3 + \frac{1}{\rho_3}\widetilde{E}_3\dot{\widetilde{E}}_3, \quad (\text{A.51})$$

while

$$\begin{aligned} v_3^* &= a_3x_6 - d_3 + g + \dot{x}_{6d} - k_5\dot{e}_5 - k_6e_5 \\ &= a_3x_6 - d_3 + g + \dot{x}_6 - \dot{S}_3 \\ &= v_3 - \dot{S}_3. \end{aligned} \quad (\text{A.52})$$

So,

$$\dot{S}_3 = v_3 - v_3^* = v_{3fz} + v_{3vs} - v_3^*. \quad (\text{A.53})$$

Substituting (A.53) into (A.51), then

$$\begin{aligned} \dot{\Gamma}_3 &= S_3\dot{S}_3 + \frac{1}{\eta_3}\widetilde{\alpha}_3^T\dot{\widetilde{\alpha}}_3 + \frac{1}{\rho_3}\widetilde{E}_3\dot{\widetilde{E}}_3 \\ &= S_3(v_{3fz} + v_{3vs} - v_3^*) + \frac{1}{\eta_3}\widetilde{\alpha}_3^T\dot{\widetilde{\alpha}}_3 + \frac{1}{\rho_3}\widetilde{E}_3\dot{\widetilde{E}}_3 \\ &= \widetilde{\alpha}_3^T\left(S_3\xi_3 + \frac{1}{\eta_3}\dot{\widetilde{\alpha}}_3\right) + S_3(v_{3vs} - \varepsilon_3) + \frac{1}{\rho_3}\widetilde{E}_3\dot{\widetilde{E}}_3. \end{aligned} \quad (\text{A.54})$$

The adaptive law (T1.17) is chosen below:

$$\dot{\widetilde{\alpha}}_3 = -\eta_3 S_3 \xi_3. \quad (\text{A.55})$$

And formula (A.54) becomes

$$\begin{aligned} \dot{\Gamma}_3 &= S_3\dot{S}_3 + \frac{1}{\eta_3}\widetilde{\alpha}_3^T\dot{\widetilde{\alpha}}_3 + \frac{1}{\rho_3}\widetilde{E}_3\dot{\widetilde{E}}_3 \\ &= \widetilde{\alpha}_3^T\left(S_3\xi_3 + \frac{1}{\eta_3}\dot{\widetilde{\alpha}}_3\right) + S_3(v_{3vs} - \varepsilon_3) + \frac{1}{\rho_3}\widetilde{E}_3\dot{\widetilde{E}}_3 \\ &= -\widehat{E}_3|S_3| - \varepsilon_3 S_3 + \frac{1}{\rho_3}(\widehat{E}_3 - E_3)\dot{\widetilde{E}}_3. \end{aligned} \quad (\text{A.56})$$

To make the Lyapunov function $\dot{\Gamma}_3 \leq 0$, the adaptive law of switching control is updated below:

$$\dot{\widehat{E}}_3 = \rho_3 |S_3|. \quad (\text{A.57})$$

Then,

$$\begin{aligned} \dot{\Gamma}_3 &= -\widehat{E}_3|S_3| - \varepsilon_3 S_3 + \frac{1}{\rho_3}(\widehat{E}_3 - E_3)\dot{\widetilde{E}}_3 \\ &\leq -E_3|S_3| - \varepsilon_3 S_3 \\ &\leq -(E_3 - |\varepsilon_3|)|S_3| \\ &\leq 0. \end{aligned} \quad (\text{A.58})$$

By associating v_1, v_2 , and v_3 , a group of virtual controls is obtained as follows:

$$\begin{cases} v_1 = (C_{x7}S_{x9}C_{x11} + S_{x7}S_{x11})U_1, \\ v_2 = (C_{x7}S_{x9}S_{x11} - S_{x7}S_{x11})U_1, \\ v_3 = (C_{x7}C_{x9})U_1. \end{cases} \quad (\text{A.59})$$

Remark 2. v_1, v_2 , and v_3 are combinations of available terms that can be given directly or measured above. Therefore, the control input U_1 can be solved by regarding them as known in the controlled system (A.59). Apparently, (A.59) has four unknown variables, namely, x_7, x_9, x_{11} , and U_1 . However, x_{11d} is usually given as an extra reference signal in advance and the integral SMC controller is designed above to ensure the rapid convergence of x_{11} to x_{11d} . Thus, x_{11} is regarded as known and can be replaced by x_{11d} in this situation, and the unknown variables are reduced. So, we can obtain the unknown variables as follows:

$$\begin{cases} x_{7d} = \arctan\left(C_{x9}\frac{bv_1 - av_2}{v_3}\right), \\ x_{9d} = \arctan\left(\frac{av_1 + bv_2}{v_3}\right), \\ U_1 = \frac{v_3}{C_{x7}C_{x9}}, \end{cases} \quad (\text{A.60})$$

where x_{7d} and x_{9d} are desired the roll and pitch angle trajectory, and U_1 is part of the ultimate control laws, $a = \cos(x_{11d})$ and $b = \sin(x_{11d})$.

In the attitude tracking system, x_{7d}, x_{9d} , and x_{11d} are taken as the desired attitude trajectory, and the design procedure of attitude tracking contains three steps.

Step 4. Define the roll error:

$$e_7 = x_7 - x_{7d}, \quad (\text{A.61})$$

where x_{7d} is the desired roll command, and the derivative of e_7 with respect to time is:

$$\dot{e}_7 = x_8 - \dot{x}_{7d}. \quad (\text{A.62})$$

Define the virtual control \bar{x}_8 ,

$$\bar{x}_8 = -c_4 e_7 + \dot{x}_{7d}, \quad (\text{A.63})$$

where c_4 is a positive constant. A new state variable x_{8d} is introduced and let \bar{x}_8 pass through the following first-order filter with constant τ_4 (T2.3) to obtain x_{8d}

$$\begin{aligned}\tau_4 \dot{x}_{8d} + x_{8d} &= \bar{x}_8, \\ x_{8d}(0) &= \bar{x}_8(0),\end{aligned}\quad (\text{A.64})$$

where x_{8d} is the output of the first-order filter; the filter error is $y_8 = x_{8d} - \bar{x}_8$. A proper integral sliding mode manifold is chosen (T2.4):

$$S_4 = x_8 - \int_0^t (\dot{x}_{8d} - k_7 \dot{e}_7 - k_8 e_7) dt, \quad (\text{A.65})$$

where k_7 and k_8 are the positive constant. If the sliding mode control is in an ideal state, the derivative of S_4 with respect to time is

$$S_4 = \dot{S}_4 = \dot{x}_8 - \dot{x}_{8d} + k_7 \dot{e}_7 + k_8 e_7 = \dot{e}_8 + k_7 \dot{e}_7 + k_8 e_7 = 0. \quad (\text{A.66})$$

Assuming that the perturbations and parameters in the equation are known, the control law for roll motion in an ideal state is designed as follows:

$$U_2^* = -a_4 x_{10} x_{12} - a_5 \omega x_{10} + a_6 x_8 - d_4 + \dot{x}_{8d} - k_7 \dot{e}_7 - k_8 e_7. \quad (\text{A.67})$$

In most cases, the system parameters are uncertain and there are also unknown external disturbances, which makes it difficult to obtain the ideal control signal U_2^* . Therefore, the fuzzy system is used to approximate the ideal control signal U_2^* and obtain

$$U_2^* = U_{2fz}(s_2, \alpha_4) + \varepsilon_4 = \alpha_4^T \xi_4 + \varepsilon_4, \quad (\text{A.68})$$

where ε_4 is the approximation error, and $|\varepsilon_4| < E_4$. Introducing switching control law U_{2vs} (T2.5) to compensate for U_2^* and U_{2fz}

$$U_{2vs} = -\hat{E}_4 \text{sgn}(S_4), \quad (\text{A.69})$$

where \hat{E}_4 is the estimation of E_4 . Then, the actual control law (T2.6) is obtained as

$$U_2 = U_{2fz} + U_{2vs}. \quad (\text{A.70})$$

Consider the Lyapunov function

$$\Gamma_4 = \frac{1}{2} S_4^2 + \frac{1}{2\eta_4} \tilde{\alpha}_4^T \tilde{\alpha}_4 + \frac{1}{2\rho_4} \tilde{E}_4^2, \quad (\text{A.71})$$

where η_4 and ρ_4 are the positive constant. The derivative of Γ_4 with respect to time can be presented as below:

$$\dot{\Gamma}_4 = S_4 \dot{S}_4 + \frac{1}{\eta_4} \tilde{\alpha}_4^T \dot{\tilde{\alpha}}_4 + \frac{1}{\rho_4} \tilde{E}_4 \dot{\tilde{E}}_4, \quad (\text{A.72})$$

while

$$\begin{aligned}U_2^* &= -a_4 x_{10} x_{12} - a_5 \omega x_{10} + a_6 x_8 - d_4 + \dot{x}_{8d} - k_7 \dot{e}_7 - k_8 e_7 \\ &= -a_4 x_{10} x_{12} - a_5 \omega x_{10} + a_6 x_8 - d_4 + \dot{x}_8 - \dot{S}_4 \\ &= U_3 - \dot{S}_4.\end{aligned}\quad (\text{A.73})$$

So,

$$\dot{S}_4 = U_2 - U_2^* = U_{2fz} + U_{2vs} - U_2^*. \quad (\text{A.74})$$

Substituting (A.74) into (A.72), then

$$\begin{aligned}\dot{\Gamma}_4 &= S_4 \dot{S}_4 + \frac{1}{\eta_4} \tilde{\alpha}_4^T \dot{\tilde{\alpha}}_4 + \frac{1}{\rho_4} \tilde{E}_4 \dot{\tilde{E}}_4 \\ &= S_4 (U_{2fz} + U_{2vs} - U_2^*) + \frac{1}{\eta_4} \tilde{\alpha}_4^T \dot{\tilde{\alpha}}_4 + \frac{1}{\rho_4} \tilde{E}_4 \dot{\tilde{E}}_4 \\ &= \tilde{\alpha}_4^T \left(S_4 \xi_4 + \frac{1}{\eta_4} \dot{\tilde{\alpha}}_4 \right) + S_4 (U_{2vs} - \varepsilon_4) + \frac{1}{\rho_4} \tilde{E}_4 \dot{\tilde{E}}_4.\end{aligned}\quad (\text{A.75})$$

The adaptive law (T2.5) is chosen below:

$$\dot{\tilde{\alpha}}_4 = -\eta_4 S_4 \xi_4. \quad (\text{A.76})$$

And formula (A.75) becomes

$$\begin{aligned}\dot{\Gamma}_4 &= S_4 \dot{S}_4 + \frac{1}{\eta_4} \tilde{\alpha}_4^T \dot{\tilde{\alpha}}_4 + \frac{1}{\rho_4} \tilde{E}_4 \dot{\tilde{E}}_4 \\ &= \tilde{\alpha}_4^T \left(S_4 \xi_4 + \frac{1}{\eta_4} \dot{\tilde{\alpha}}_4 \right) + S_4 (U_{2vs} - \varepsilon_4) + \frac{1}{\rho_4} \tilde{E}_4 \dot{\tilde{E}}_4 \\ &= -\hat{E}_4 |S_4| - \varepsilon_4 S_4 + \frac{1}{\rho_4} (\hat{E}_4 - E_4) \dot{\tilde{E}}_4.\end{aligned}\quad (\text{A.77})$$

To make the Lyapunov function $\dot{\Gamma}_4 \leq 0$, the adaptive law of switching control is updated below:

$$\dot{\tilde{E}}_4 = \rho_4 |S_4|. \quad (\text{A.78})$$

Then,

$$\begin{aligned}
\dot{\Gamma}_4 &= -\widehat{E}_4|S_4| - \varepsilon_4 S_4 + \frac{1}{\rho_4}(\widehat{E}_4 - E_4)\dot{\widehat{E}}_4 \\
&\leq -E_4|S_4| - \varepsilon_4 S_4 \\
&\leq -(E_4 - |\varepsilon_4|)|S_4| \\
&\leq 0.
\end{aligned} \tag{A.79}$$

The similar design processes are presented to design the trajectory tracking of pitch x_9 and yaw x_{11} using the dynamic surface integral sliding mode control. The integral sliding mode manifolds (T2.10), (T2.16) and the first-order filter (T2.9), (T2.15) are chosen properly. The adaptive law and the switching control (T2.11), (T2.18) of the pitch and yaw are presented in Table 2. And (T2.12), (T2.18) are the actual control inputs of the pitch and yaw equation. $c_5, c_6, \tau_5, \tau_6, k_9, k_{10}, k_{11}, k_{12}, \eta_5, \eta_6, \rho_5$, and ρ_6 are positive constants which need to be assigned to meet the performance requirements of the pitch and yaw angle system. The derivatives of Lyapunov function of pitch and yaw angle are presented as follows:

$$\begin{aligned}
\dot{\Gamma}_5 &= -\widehat{E}_5|S_5| - \varepsilon_5 S_5 + \frac{1}{\rho_5}(\widehat{E}_5 - E_5)\dot{\widehat{E}}_5 \\
&\leq -E_5|S_5| - \varepsilon_5 S_5 \\
&\leq -(E_5 - |\varepsilon_5|)|S_5| \\
&\leq 0, \\
\dot{\Gamma}_6 &= -\widehat{E}_6|S_6| - \varepsilon_6 S_6 + \frac{1}{\rho_6}(\widehat{E}_6 - E_6)\dot{\widehat{E}}_6 \\
&\leq -E_6|S_6| - \varepsilon_6 S_6 \\
&\leq -(E_6 - |\varepsilon_6|)|S_6| \\
&\leq 0.
\end{aligned} \tag{A.80}$$

B. The Proof of Theorem

Proof. Consider the Lyapunov function candidate

$$\begin{aligned}
V &= V_1 + V_2, \\
V_1 &= \frac{1}{2} \sum_{i=1}^6 (e_{2i-1}^2 + y_{2i}^2), \\
V_2 &= \frac{1}{2} \sum_{i=1}^6 \left(S_i^2 + \frac{1}{\eta_i} \widetilde{\alpha}_i^T \widetilde{\alpha}_i + \frac{1}{\rho_i} \widetilde{E}_i^2 \right).
\end{aligned} \tag{B.1}$$

The derivative of V_1 with respect to time can be obtained as follows:

$$\dot{V}_1 = \sum_{i=1}^6 (e_{2i-1} \dot{e}_{2i-1} + y_{2i} \dot{y}_{2i}). \tag{B.2}$$

Substitute the adaptive law and the switching control law into (B.2):

$$\begin{aligned}
\dot{V}_1 &= \sum_{i=1}^6 \left(-c_i e_{2i-1}^2 - \frac{y_{2i}^2}{\tau_i} + |y_{2i} B_{2i}| \right) \\
&\leq \sum_{i=1}^6 \left(-c_i e_{2i-1}^2 - \frac{y_{2i}^2}{\tau_i} + |y_{2i} B_{2i}| \right).
\end{aligned} \tag{B.3}$$

Noting that, for any positive number $\lambda, |y_{2i} B_{2i}| \leq (y_{2i}^2 B_{2i}^2 / 2\lambda) + (\lambda/2)$. Assume that $|B_{2i}| < M_{2i}$, where M_{2i} is a positive constant.

$$\begin{aligned}
\dot{V}_1 &\leq \sum_{i=1}^6 \left(-c_i e_{2i-1}^2 - \frac{y_{2i}^2}{\tau_i} + \frac{y_{2i}^2 B_{2i}^2}{2\lambda} + \frac{\lambda}{2} \right) \\
&\leq \sum_{i=1}^6 \left(-c_i e_{2i-1}^2 - \left(\frac{2}{\tau_i} - \frac{M_{2i}^2}{\lambda} \right) \frac{y_{2i}^2}{2} + \frac{\lambda}{2} \right).
\end{aligned} \tag{B.4}$$

Let

$$\begin{cases} c_i = \alpha_0, \\ \frac{2}{\tau_i} - \frac{M_{2i}^2}{\lambda} = \alpha_0, \end{cases} \tag{B.5}$$

where α_0 is a positive constant.

$$\dot{V}_1 \leq -\alpha_0 V_1 + 3\lambda. \tag{B.6}$$

Let $\alpha_0 > (3\lambda/2p)$; then $\dot{V}_1 \leq 0$ on $V_1 = p$. And $V_1 \leq p$ is an invariant set. Thus, if $V_1(0) \leq p$, then $V_1(t) \leq p$ for all $t \geq 0$. Thus, (B.6) holds for all $V_1(t) \leq p$ and all $t \geq 0$.

$$0 \leq V_1(t) \leq \frac{3\lambda}{4\alpha_0} + \left(V_1(0) - \frac{3\lambda}{4\alpha_0} \right) e^{-2\alpha_0 t}, \quad \forall t \geq 0. \tag{B.7}$$

From (B.7), we can get that $V_1(t)$ is eventually bounded by $(3\lambda/4\alpha_0)$.

$$V_2 = \frac{1}{2} \sum_{i=1}^6 \left(S_i^2 + \frac{1}{\eta_i} \widetilde{\alpha}_i^T \widetilde{\alpha}_i + \frac{1}{\rho_i} \widetilde{E}_i^2 \right). \tag{B.8}$$

The derivative of V_2 with respect to time can be obtained as follows:

$$\begin{aligned}
\dot{V}_2 &= \sum_{i=1}^6 \left(S_i \dot{S}_i + \frac{1}{\eta_i} \widetilde{\alpha}_i^T \dot{\widetilde{\alpha}}_i + \frac{1}{\rho_i} \widetilde{E}_i \dot{\widetilde{E}}_i \right) \\
&= \sum_{i=1}^6 \left[S_i (U_{ifz} + U_{ivs} - U_i^*) + \frac{1}{\eta_i} \widetilde{\alpha}_i^T \dot{\widetilde{\alpha}}_i + \frac{1}{\rho_i} \widetilde{E}_i \dot{\widetilde{E}}_i \right] \\
&= \sum_{i=1}^6 \left[\widetilde{\alpha}_i^T \left(S_i \xi_i + \frac{1}{\eta_i} \dot{\widetilde{\alpha}}_i \right) + S_i (U_{ivs} - \varepsilon_i) + \frac{1}{\rho_i} \widetilde{E}_i \dot{\widetilde{E}}_i \right].
\end{aligned} \tag{B.9}$$

Substitute the adaptive laws and the switching control laws (T1.5), (T1.11), (T1.17), (T2.5), (T2.11), and (T2.17) into (B.9)

$$\begin{aligned}
\dot{V}_2 &= \sum_{i=1}^6 \left[-\hat{E}_i |S_i| - \varepsilon_i S_i + (\hat{E}_i - E_i) |S_i| \right] \\
&= \sum_{i=1}^3 \left[-E_i |S_i| - \varepsilon_i S_i \right] \\
&\leq \sum_{i=1}^3 \left[-(E_i - |\varepsilon_i|) |S_i| \right] \\
&\leq 0,
\end{aligned} \tag{B.10}$$

where $E_i \geq |\varepsilon_i|$, so we can know that $V_2(t)$ is eventually bounded. Hence, all signals in the closed loop are ultimately bounded. Particularly, the tracking errors and the estimation errors can be arbitrarily small. \square

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported in part by the NSF of China under Grant no. 61673101 and the Thirteenth Five-Year Science Research Plan of Jilin Province under Grant JJKH20200119KJ.

References

- [1] B. Zhao, B. Xian, Y. Zhang, and X. Zhang, "Nonlinear robust adaptive tracking control of a quadrotor uav via immersion and invariance methodology," *IEEE Transactions on Industrial Electronics*, vol. 62, no. 5, pp. 2891–2902, 2014.
- [2] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: modeling, estimation, and control of quadrotor," *IEEE Robotics & Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.
- [3] T. Huang, D. Huang, Z. Wang, and A. Shah, "Robust tracking control of a quadrotor UAV based on adaptive sliding mode controller," *Complexity*, vol. 2019, Article ID 7931632, 15 pages, 2019.
- [4] L. Besnard, Y. B. Shtessel, and B. Landrum, "Quadrotor vehicle control via sliding mode controller driven by sliding mode disturbance observer," *Journal of the Franklin Institute*, vol. 349, no. 2, pp. 658–684, 2012.
- [5] Z. T. Dydek, A. M. Annaswamy, and E. Lavretsky, "Adaptive control of quadrotor uavs: a design trade study with flight evaluations," *IEEE Transactions on Control Systems Technology*, vol. 21, no. 4, pp. 1400–1406, 2012.
- [6] X. Zhang, Y. Wang, G. Zhu et al., "Compound adaptive fuzzy quantized control for quadrotor and its experimental verification," *IEEE Transactions on Cybernetics*, pp. 1–13, 2020.
- [7] X. Dong, B. Yu, Z. Shi, and Y. Zhong, "Time-varying formation control for unmanned aerial vehicles: theories and applications," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 1, pp. 340–348, 2014.
- [8] J.-J. Xiong and E.-H. Zheng, "Position and attitude tracking control for a quadrotor uav," *ISA Transactions*, vol. 53, no. 3, pp. 725–731, 2014.
- [9] P. E. Pounds, D. R. Bersak, and A. M. Dollar, "Stability of small-scale uav helicopters and quadrotors with added payload mass under pid control," *Autonomous Robots*, vol. 33, no. 1-2, pp. 129–142, 2012.
- [10] Z. Zuo, "Trajectory tracking control design with command-filtered compensation for a quadrotor," *IET Control Theory & Applications*, vol. 4, no. 11, pp. 2343–2355, 2010.
- [11] S. He, K. Relf, and R. Unbehauen, "A neural approach for control of nonlinear systems with feedback linearization," *IEEE Transactions on Neural Networks*, vol. 9, no. 6, pp. 1409–1421, 1998.
- [12] I. A. Raptis, K. P. Valavanis, and W. A. Moreno, "A novel nonlinear backstepping controller design for helicopters using the rotation matrix," *IEEE Transactions on Control Systems Technology*, vol. 19, no. 2, pp. 465–473, 2010.
- [13] G. Zhu and J. Liu, "Neural network-based adaptive backstepping control for hypersonic flight vehicles with prescribed tracking performance," *Mathematical Problems in Engineering*, vol. 2015, Article ID 591789, 10 pages, 2015.
- [14] Y. Yu, C. Zhang, and M. Zhou, "Narmax model-based hysteresis modeling of magnetic shape memory alloy actuators," *IEEE Transactions on Nanotechnology*, vol. 19, pp. 1–4, 2020.
- [15] X. Zhang, Y. Wang, X. Chen et al., "Decentralized adaptive neural approximated inverse control for a class of large-scale nonlinear hysteretic systems with time delays," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 49, no. 12, pp. 2424–2437, 2019.
- [16] Z. Jia, J. Yu, Y. Mei, Y. Chen, Y. Shen, and X. Ai, "Integral backstepping sliding mode control for quadrotor helicopter under external uncertain disturbances," *Aerospace Science and Technology*, vol. 68, pp. 299–307, 2017.
- [17] Y. Kui, G. Feng, Y. Liying, H. Yuqing, and H. Jianda, "Sliding Mode Control for a Quadrotor Slung Load System," in *Proceedings of the 2017 36th Chinese Control Conference (CCC)*, pp. 3697–3703, IEEE, Dalian, China, July 2017.
- [18] Y. Zou, "Nonlinear robust adaptive hierarchical sliding mode control approach for quadrotors," *International Journal of Robust and Nonlinear Control*, vol. 27, no. 6, pp. 925–941, 2017.
- [19] X. Xiang, C. Liu, H. Su, and Q. Zhang, "On decentralized adaptive full-order sliding mode control of multiple uavs," *ISA Transactions*, vol. 71, pp. 196–205, 2017.
- [20] T. Li, Y. Zhang, and B. W. Gordon, "Nonlinear fault-tolerant control of a quadrotor uav based on sliding mode control technique," *IFAC Proceedings Volumes*, vol. 45, no. 20, pp. 1317–1322, 2012.
- [21] B. Sumantri, N. Uchiyama, and S. Sano, "Least square based sliding mode control for a quad-rotor helicopter and energy saving by chattering reduction," *Mechanical Systems and Signal Processing*, vol. 66–67, pp. 769–784, 2016.
- [22] B. Xu, X. Liu, H. Wang, and Y. Zhou, "Event-triggered adaptive backstepping control for strict-feedback nonlinear systems with zero dynamics," *Complexity*, vol. 2019, Article ID 7890968, 13 pages, 2019.
- [23] S. Zhang and C. Chen, "Backstepping based nonlinear integral sliding mode control for quadrotors under external disturbances," in *Proceedings of the 2019 Chinese Control Conference (CCC)*, pp. 8355–8359, IEEE, Guangzhou, China, July 2019.
- [24] M. A. M. Basri, "Robust backstepping controller design with a fuzzy compensator for autonomous hovering quadrotor uav,"

- Iranian Journal of Science and Technology, Transactions of Electrical Engineering*, vol. 42, no. 3, pp. 379–391, 2018.
- [25] H. Shraim, Y. Harkouss, and H. Bazzi, “A novel hybrid backstepping and fuzzy logic control strategy for a quadcopter,” *The Aeronautical Journal*, vol. 121, no. 1244, pp. 1444–1463, 2017.
- [26] X. Zhang, X. Chen, G. Zhu, and C.-Y. Su, “Output feedback adaptive motion control and its experimental verification for time-delay nonlinear systems with asymmetric hysteresis,” *IEEE Transactions on Industrial Electronics*, vol. 67, no. 8, pp. 6824–6834, 2020.
- [27] X. Tang, “Adaptive neural network dynamic surface control for a class of unknown time-delay nonlinear hysteresis system,” *Journal of Northeast Electric Power University*, vol. 38, no. 3, pp. 52–60, 2018.
- [28] Z. Li, J. Shan, and U. Gabbert, “Dynamics modeling and inversion-based synchronized model predictive control for a Fabry-perot spectrometer,” *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 4, pp. 1818–1828, 2019.
- [29] G. Zhu, L. Nie, M. Zhou, X. Zhang, L. Sun, and C. Zhong, “Adaptive fuzzy dynamic surface control for multi-machine power system based on composite learning method and disturbance observer,” *IEEE Access*, vol. 8, pp. 163163–163175, 2020.
- [30] X. Zhang, B. Li, X. Chen, Z. Li, Y. Peng, and C.-Y. Su, “Adaptive implicit inverse control for a class of discrete-time hysteretic nonlinear systems and its application,” *IEEE/ASME Transactions on Mechatronics*, vol. 25, no. 4, pp. 2112–2122, 2020.
- [31] C. Yang, Y. Jiang, W. He, J. Na, Z. Li, and B. Xu, “Adaptive parameter estimation and control design for robot manipulators with finite-time convergence,” *IEEE Transactions on Industrial Electronics*, vol. 65, no. 10, pp. 8112–8123, 2018.
- [32] Z. Li and J. Shan, “Inverse compensation based synchronization control of the piezo-actuated Fabry-perot spectrometer,” *IEEE Transactions on Industrial Electronics*, vol. 64, no. 11, pp. 8588–8597, 2017.
- [33] J. Yu, Y. Ma, H. Yu, and C. Lin, “Adaptive fuzzy dynamic surface control for induction motors with iron losses in electric vehicle drive systems via backstepping,” *Information Sciences*, vol. 376, pp. 172–189, 2017.
- [34] Y. Jiang, C. Yang, S.-l. Dai, and B. Ren, “Deterministic learning enhanced neural network control of unmanned helicopter,” *International Journal of Advanced Robotic Systems*, vol. 13, no. 6, 2016.
- [35] X. Shao, J. Liu, H. Cao, C. Shen, and H. Wang, “Robust dynamic surface trajectory tracking control for a quadrotor uav via extended state observer,” *International Journal of Robust and Nonlinear Control*, vol. 28, no. 7, pp. 2700–2719, 2018.
- [36] B. Xu, C. Yang, and Y. Pan, “Global neural dynamic surface tracking control of strict-feedback systems with application to hypersonic flight vehicle,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 26, no. 10, pp. 2563–2575, 2015.
- [37] X. Zhang, Y. Wang, C. Wang, C.-Y. Su, Z. Li, and X. Chen, “Adaptive estimated inverse output-feedback quantized control for piezoelectric positioning stage,” *IEEE Transactions on Cybernetics*, vol. 49, no. 6, pp. 2106–2118, 2019.
- [38] C. Yang, Y. Jiang, J. Na, Z. Li, L. Cheng, and C.-Y. Su, “Finite-time convergence adaptive fuzzy control for dual-arm robot with unknown kinematics and dynamics,” *IEEE Transactions on Fuzzy Systems*, vol. 27, no. 3, pp. 574–588, 2018.
- [39] D. Huang, C. Yang, Y. Pan, and L. Cheng, “Composite learning enhanced neural control for robot manipulator with output error constraints,” *IEEE Transactions on Industrial Informatics*, vol. 17, no. 1, pp. 209–218, 2019.
- [40] G. Zhu, L. Nie, Z. Lv, L. Sun, X. Zhang, and C. Wang, “Adaptive fuzzy dynamic surface sliding mode control of large-scale power systems with prescribe output tracking performance,” *ISA Transactions*, vol. 99, pp. 305–321, 2020.
- [41] S. Mobayen, F. Tchier, and L. Ragoub, “Design of an adaptive tracker for n-link rigid robotic manipulators based on super-twisting global nonlinear sliding mode control,” *International Journal of Systems Science*, vol. 48, no. 9, pp. 1990–2002, 2017.
- [42] S. He and J. Song, “Finite-time sliding mode control design for a class of uncertain conic nonlinear systems,” *IEEE/CAA Journal of Automatica Sinica*, vol. 4, no. 4, pp. 809–816, 2017.
- [43] G. Zhu, S. Wang, L. Sun, W. Ge, and X. Zhang, “Output feedback adaptive dynamic surface sliding mode control for quadrotor uavs with tracking error constraints,” *Complexity*, vol. 2020, Article ID 8537198, 23 pages, 2020.
- [44] M. Manceur, N. Essounbouli, and A. Hamzaoui, “Second-order sliding fuzzy interval type-2 control for an uncertain system with real application,” *IEEE Transactions on Fuzzy Systems*, vol. 20, no. 2, pp. 262–275, 2011.
- [45] S. Mobayen, “Design of cnf-based nonlinear integral sliding surface for matched uncertain linear systems with multiple state-delays,” *Nonlinear Dynamics*, vol. 77, no. 3, pp. 1047–1054, 2014.
- [46] M. Golestani, F. Tchier, and S. Mobayen, “Adaptive finite-time tracking control of uncertain non-linear n-order systems with unmatched uncertainties,” *IET Control Theory & Applications*, vol. 10, no. 14, pp. 1675–1683, 2016.
- [47] I. González, S. Salazar, and R. Lozano, “Chattering-free sliding mode altitude control for a quadrotor aircraft: real-time application,” *Journal of Intelligent & Robotic Systems*, vol. 73, no. 1–4, pp. 137–155, 2014.
- [48] S. Mobayen, “A novel global sliding mode control based on exponential reaching law for a class of under actuated systems with external disturbances,” *Journal of Computational and Nonlinear Dynamics*, vol. 11, no. 2, 9 pages, Article ID 021011, 2016.
- [49] M. Chen, Q.-X. Wu, and R.-X. Cui, “Terminal sliding mode tracking control for a class of siso uncertain nonlinear systems,” *ISA Transactions*, vol. 52, no. 2, pp. 198–206, 2013.
- [50] E.-H. Zheng, J.-J. Xiong, and J.-L. Luo, “Second order sliding mode control for a quadrotor uav,” *ISA Transactions*, vol. 53, no. 4, pp. 1350–1356, 2014.
- [51] F. Chen, R. Jiang, K. Zhang, B. Jiang, and G. Tao, “Robust backstepping sliding-mode control and observer-based fault estimation for a quadrotor UAV,” *IEEE Transactions on Industrial Electronics*, vol. 63, no. 8, pp. 5044–5056, 2016.

Research Article

Acceleration Level Control of Redundant Manipulators with Physical Constraints Compliance and Disturbance Rejection under Complex Environment

Jinglun Liang,¹ Yisheng Rong,¹ Guoliang Ye,¹ Xiaoxiao Li ,² Jianwen Guo,¹ and Zhenzhen He³

¹School of Mechanical Engineering, Dongguan University of Technology, Dongguan, Guangdong 523808, China

²Guangdong Key Laboratory of Modern Control Technology, Guangdong Institute of Intelligent Manufacturing, Guangzhou, Guangdong 510070, China

³Guangdong OPT Co., Ltd., Dongguan, Guangdong, China

Correspondence should be addressed to Xiaoxiao Li; xiaoxiaoli1993@sina.com

Received 12 August 2020; Revised 4 September 2020; Accepted 20 October 2020; Published 9 November 2020

Academic Editor: Ning Wang

Copyright © 2020 Jinglun Liang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Investigation of joint torque constraint compliance is of significance for robot manipulators especially working in complex environments. A lot of which is attributed to that, on the one hand, it is beneficial to the improvement of both safety and reliability of the mission execution. On the other hand, the energy consumption required by the robot to complete the desired mission can be reduced. Most existing schemes do not take the joint torque limit and other inherent physical structure limits in a manipulator into account at the same time. In addition, many unavoidable uncertainties such as the external environmental disturbance and/or electromagnetism interferences in the circuit system may influence the accuracy and effectiveness of the task execution for a robot. In this study, we cast light on the acceleration level control of redundant robot manipulators considering both four physical constraint limits and interference rejection. A robust unified quadratic-programming-based hybrid control scheme is proposed, where the joint torque constraints are converted as two inequality constraints based on the robots' dynamics equation. A recurrent-neural-network-based controller is designed for solving the control variable. Numerical experiments performing in PUMA 560 manipulator and planer manipulator illustrate that a rational torque distribution is obtained among the joints and the considered physical structural vectors are all restricted to the respective constraint range. In addition, even disturbed by the noise, the manipulator still successfully tracks the desired trajectory under the proposed control scheme.

1. Introduction

With the gradually mature robotic technology, the robot is being applied to all kinds of complicated or dangerous tasks such as deep-sea exploring, search, and rescue tasks in quake-hit areas [1]. It imposes a challenge on the safety and reliability of the mission execution for a robot [2, 3]. A manipulator is considered to be redundant if its degrees of freedom (DOFs) is more than the minimal ones required by the robot to complete the desired end-effector task [4–8]. Due to redundancy, except that the manipulator can complete the primary end-effector task, optimization of

some performance indices and multiple additional subtasks such as physical constraints compliance [9, 10], avoiding collision with the detected obstacles [11, 12], repetitive motion planning [13, 14], etc., can be achieved simultaneously for redundant manipulators [15]. Multiobjectives-integrated hybrid tasks have been achieved in both a single manipulator [4–6] and collective ones [7–9]. As one of the important physical variables, joint torque limits compliance of a redundant manipulator is necessary to be considered. If the joint torque is suddenly enlarged, the robotic structure or the surrounding things will be possible to be damaged. Moreover, if too large torque is always imposed on the

manipulator, the service life of the motor embedded inside the robot will be reduced. More importantly, if the torque is overload, the desired end-effector task cannot be done at all owing to the deviation between the desired driving force and the actual driving force output by the motor-driven robot. Joint torque optimization is equivalent to optimize the input power of the manipulators' joint actuator [15, 16]. Therefore, to improve both the safety and reliability of mission execution for a robot manipulator and reduce energy consumption while completing the desired task, it is necessary to consider the joint torque limits of redundant manipulator among the robot control scheme.

Many efforts taking joint torque limits into account have been made. The easier schemes are solved through the null-space or pseudoinverse method [17, 18]. In addition to the found instability problem for both, the pseudoinverse method is with the higher computational cost due to the pseudoinverse of the Jacobian matrix needs to be computed. Moreover, high demand for the robot working in a complicated environment makes the robot expected to simultaneously achieve the primary end-effector task and multiple secondary subtasks. It has no ability to meet this requirement obviously owing to the pseudoinverse method that does not handle such physical constraints inherent in the robot. To solve it, the quadratic-programming- (QP-) based optimization scheme is developed and widely used, whose typical feature is that these subtasks such as obstacle avoidance and physical constraints are uniformly described as independent attachment equality or inequality constraint.

The literature [19] investigated the inverse kinematics problem of redundant manipulators subject to torque limit, where minimum torque infinity norm (MIN) was chosen as the objective function that was to be minimized, the primary task was described as an equality constraint. Therefore, a time-varying QP formulation was obtained, which was solved by the recurrent neural network (RNN) or called Lagrange neural network. Merely, the joint angle, velocity, and acceleration limits were ignored. As an extension of [19], Zhang et al. considered joint angle physical constraint in [20] by describing it as an inequality constraint. The resultant QP formulation was solved with a dual neural network, where the cost function was chosen as a minimum torque norm (MTN). In [15] and [21], physical constraints such as joint velocity and joint acceleration were considered in the QP formulation and a linear-variational-inequalities-based primal-dual neural network was then employed to solve the control variable. Five schemes were investigated in [15] by minimizing different objective functions. It was concluded that the minimum acceleration norm (MAN) was superior to the MTN scheme. However, these three papers did not take the joint torque limit into account. In [16] and [21], the bicriteria joint torque minimization was considered, where the objective functions were chosen as a weighted combination of the MTN and MIN schemes. In [14], bicriteria minimization integrating both the MTN and the repetitive motion planning schemes was studied. However, any joint limits were not considered in this paper. In [22], the motion-force control problem of redundant manipulators was investigated based on RNN. The joint

acceleration limit and the dynamics were not be considered yet. Among the above-mentioned control scheme, they did not take physical limits including joint torque, joint angle, velocity, and acceleration into account at the same time. Considering the physical structure of a robot manipulator, whose every joint is usually driven by a motor, the reachable workspace and the output torque of a robot are eventually constrained. No matter what any physical constraints are satisfied, the robot would have no ability to execute the desired task, and the output control variables are unavailable.

A consensus in [14–16, 19–21] is that neural networks are utilized to control the redundant manipulators. Recently, the neural network method is popular and has been widely used, such as in [22–30] and references therein due to its being parallel, nonlinear, and simple to be realized by hardware [31]. However, most of them are based on an assumption of disturbance-free inverse kinematic control of the manipulators among the reported products. In practice, many uncertainties such as the external environmental disturbance and/or electromagnetism interferences in circuit system inevitably exist during the control signal transmission [32–35]. Disturbed by them, the accuracy and effectiveness of the task execution may be influenced greatly by a robot system.

To sum up, although some brilliant advances have been made in joint torque optimization for redundant manipulators, there exist some limits. For one thing, some works do not take the torque limit into account or joint limits such as velocity, angle, and acceleration are not considered. For another thing, the existing torque optimization works take the disturbance into account except for reference [14]. However, [14] did not take any physical limits from robots into account. Motivated by it, this paper investigates the inverse kinematics control problem of redundant manipulator considering both the interference rejection and the above-mentioned four physical constraints compliance and proposes a robust unified QP-based hybrid optimization scheme. Among the resultant QP scheme, the MAN is chosen as the objective function, and the inverse kinematics problem is described as an equality constraint. Joint angle, velocity, and acceleration limits are uniformly described as inequality constraints and solved in acceleration level. Joint torque limits are converted to two inequality constraints based on the robot's dynamics equation. To illustrate the difference between this paper and the previous QP-based works, a comparison between them is conducted and listed in Table 1. The main contributions of this paper are summarized as follows:

- (1) Based on RNN, this paper investigates the acceleration level inverse kinematics control problem of redundant manipulators with physical constraints compliance and disturbance rejection. A multi-objectives-integrated robust unified QP hybrid scheme is proposed.
- (2) Different from the previous works, this paper considers joint angle, velocity, acceleration, and joint torque limits simultaneously. A simple way is given

TABLE 1: Comparison between this paper and the existing QP-based torque schemes.

Methods	Joint angle limit	Joint velocity limit	Joint acceleration limit	Joint torque limit	Disturbances	Dynamics
Our	Yes	Yes	Yes	Yes	Yes	Yes
[19]	No	No	No	No	No	Yes
[20]	Yes	No	No	No	No	Yes
[15]	Yes	Yes	Yes	No	No	Yes
[16]	No	No	No	Yes	No	Yes
[21]	Yes	Yes	Yes	No	No	Yes
[14]	No	No	No	No	Yes	Yes
[22]	Yes	Yes	No	Yes	No	No

to handle the considered four double-sides inequality constraints because they are built in different levels.

- (3) Under the hybrid scheme, high joint acceleration and joint torque can be avoided during the mission execution. Constant-valued and time-varying noises are investigated, validating the robustness of the proposed scheme against external interferences.
- (4) The theoretical analysis and numerical experiment combinatorially show that the effectiveness and feasibility of the designed dynamic RNN controller for the redundant manipulators' acceleration level control.

2. QP Problem Formulation

In this paper, we are aimed at achieving that in addition to successfully complete the desired end-effector tracking task, high joint angle, velocity, acceleration, and torque are all avoided, and they all should be restricted within the respective constrained range. At the same time, the control scheme is expected to be robust against external interferences. Therefore, three objectives need to be achieved simultaneously. They are as follows:

Objective 1 (inverse kinematic control): The inverse kinematic control of the manipulator is a fundamental problem in robotic control. Given the desired trajectory of the manipulators' end-effector, to find the corresponding joint variables is called the inverse kinematics problem of the robot. As for Objective 1, we have the following:

$$r \longrightarrow r_d, \quad (1)$$

where $r_d \in \mathbb{R}^m$ denotes the desired end-effector coordinate that the robot is expected to follow in Cartesian space. The actual end-effector coordinate achieved by the manipulator is denoted by the vector r and $r \in \mathbb{R}^m$. In addition, the relationship between the Cartesian coordinate r of the manipulators' end-effector and its joint space coordinate vector θ is described as follows:

$$r = f(\theta), \quad (2)$$

where $\theta \in \mathbb{R}^n$. $f(\cdot)$ denotes a nonlinear mapping from joint space to Cartesian space.

Objective 2 (physical constraint compliance): Every joint is usually driven by a motor for a manipulator. Therefore its reachable workspace and the output joint torque, velocity, and acceleration are all limited. No matter which constraints are violated, the accuracy of task execution for a manipulator will be affected. For practicability and safety, therefore, overloaded or high joint velocity, acceleration, and joint torque should be avoided when the manipulator executes the desired task. These above-mentioned robot physical limits can be described by the following two-sides inequality constraints:

$$\theta^- \leq \theta \leq \theta^+, \quad (3a)$$

$$\dot{\theta}^- \leq \dot{\theta} \leq \dot{\theta}^+, \quad (3b)$$

$$\ddot{\theta}^- \leq \ddot{\theta} \leq \ddot{\theta}^+, \quad (3c)$$

$$\tau^- \leq \tau \leq \tau^+, \quad (3d)$$

where the variables θ , $\dot{\theta}$, $\ddot{\theta}$, and $\tau \in \mathbb{R}^n$ are joint angle vector, joint velocity vector, joint acceleration vector, and joint torque vector of the manipulator, respectively. θ^- , $\dot{\theta}^-$, $\ddot{\theta}^-$, τ^- and θ^+ , $\dot{\theta}^+$, $\ddot{\theta}^+$, τ^+ correspond to lower bound and upper bound of θ , $\dot{\theta}$, $\ddot{\theta}$, τ , respectively.

Due to redundancy ($m < n$), the unique solution satisfying equation (2) does not exist. In addition, because joint variables θ , $\dot{\theta}$, $\ddot{\theta}$, and τ are built in different levels, directly solving them is difficult. In this paper, the control problem is solved at the acceleration level. Specially, computing the second-derivatives of equation (2), the acceleration level kinematics is described as follows:

$$\ddot{r} = \ddot{J}\dot{\theta} + \dot{J}\dot{\theta}, \quad (4)$$

where \dot{J} is derivative of J . $J \in \mathbb{R}^{m \times n}$ is a Jacobian matrix, determined by the DH parameters of the manipulator. In general, the manipulators' physical structure and DH parameters are known in advance; therefore J and \dot{J} are convenient to be obtained. \ddot{r} is the second-derivative of r , denoting acceleration vector of the robot in Cartesian space.

As for the two-sides inequality constraints equations (3a)–(3c), based on [36, 37], they can be described as follows:

$$\begin{aligned}\xi^+ &= \min\left\{\kappa_1(\theta^+ - \vartheta - \theta), \kappa_2(\dot{\theta}^+ - \dot{\theta}), \ddot{\theta}^+\right\}, \\ \xi^- &= \max\left\{\kappa_1(\theta^- + \vartheta - \theta), \kappa_2(\dot{\theta}^- - \dot{\theta}), \ddot{\theta}^-\right\},\end{aligned}\quad (5)$$

where constants $\vartheta, \kappa_1, \kappa_2 > 0 \in \mathbb{R}$ are adjusted by the designer based on the required experimental results. For τ , based on the manipulator's dynamics, it is obtained [15, 20] that

$$\tau = M(\theta)\ddot{\theta} + c(\theta, \dot{\theta}) + g(\theta), \quad (6)$$

where $M(\theta) \in \mathbb{R}^{n \times n}$ is the inertia matrix, $c(\theta, \dot{\theta}) \in \mathbb{R}^n$ is the Coriolis and centrifugal force vector, and $g(\theta) \in \mathbb{R}^n$ is the gravitational force vector. Therefore, equation (3d) can be rewritten as two inequality constraints related to θ :

$$\begin{cases} M(\theta)\ddot{\theta} \leq \tau^+ - c(\theta, \dot{\theta}) - g(\theta), \\ -M(\theta)\ddot{\theta} \leq -\tau^- + c(\theta, \dot{\theta}) + g(\theta). \end{cases} \quad (7)$$

Objective 3 (disturbance rejection): Many uncertainties originating from external and internal interferences are possible to influence both the accuracy and effectiveness of the manipulator's mission execution. Uncertainties caused by the change of external environment or interaction between the robot and environment are viewed as external disturbances. Uncertainties caused by the internal parameters deviation among the manipulator structure are then viewed as the internal disturbances, such as differential error.

In this paper, we focus on the external disturbance rejection. The rejection investigation of the internal disturbance can refer to [14]. Uncertainties that disturb the task execution precision of the robot system can be described mathematically. As described in [35], the external disturbances can be mathematically described as linear-form disturbance, sine-form, exponential-form, and random disturbances and so on, respectively. Specifically, offset errors in robot hardware implementation could be described as linear disturbances. In the process of signal processing and transmission for robots, the signal disturbances caused by electromagnetic interference are described as sine-form disturbances. The case of instantaneous decline of a power source which causes the loss of control signal is viewed as the exponential-decay-form disturbance. Interferences caused by the change of external environment is then described as a random disturbance. The Objective 1 disturbed by the external disturbance in acceleration level can be rewritten as follows:

$$\ddot{r} + \omega \longrightarrow \ddot{r}_d, \quad J\ddot{\theta} + \dot{J}\dot{\theta} + \omega \longrightarrow \ddot{r}_d, \quad (8)$$

where $\omega \in \mathbb{R}^m$ denotes the external disturbance vector. \ddot{r}_d is the second-order derivation of r_d , denoting the desired acceleration vector.

To reject these noises and achieve $\ddot{r} + \omega \longrightarrow \ddot{r}_d$, inspired by [38–40], an integration-enhanced negative feedback is introduced in equation (8), that is to say,

$$J\ddot{\theta} = \ddot{r}_d - \dot{J}\dot{\theta} - \alpha(r - r_d) - \beta(J\dot{\theta} - \dot{r}_d) - \gamma \int_0^t (r - r_d)dt + \omega, \quad (9)$$

where constants $\alpha, \beta, \gamma > 0 \in \mathbb{R}$ are used to scale the tracking accuracy to the desired tracking trajectory. \dot{r}_d denotes the desired velocity vector. The specific derivation of equation (9) would be shown in the appendix.

In conclusion, the above-mentioned three objectives are able to be uniformly described as a dynamic QP formulation:

$$\min_{\ddot{\theta}} \ddot{\theta}^T \left(\frac{\ddot{\theta}}{2} \right), \quad (10a)$$

$$\text{s.t. } J\ddot{\theta} = b_0, \quad (10b)$$

$$M(\theta)\ddot{\theta} \leq b_1, \quad (10c)$$

$$-M(\theta)\ddot{\theta} \leq b_2, \quad (10d)$$

$$\xi^+ = \min\left\{\kappa_1(\theta^+ - \vartheta - \theta), \kappa_2(\dot{\theta}^+ - \dot{\theta}), \ddot{\theta}^+\right\}, \quad (10e)$$

$$\xi^- = \max\left\{\kappa_1(\theta^- + \vartheta - \theta), \kappa_2(\dot{\theta}^- - \dot{\theta}), \ddot{\theta}^-\right\}, \quad (10f)$$

where the cost function is chosen as the minimization of the joint acceleration norm. b_0, b_1 , and b_2 are defined as $b_0 = \ddot{r}_d - \dot{J}\dot{\theta} - \alpha e - \beta \dot{e} - \gamma \int_0^t e dt + \omega$, $b_1 = \tau^+ - c(\theta, \dot{\theta}) - g(\theta)$, $b_2 = -\tau^- + c(\theta, \dot{\theta}) + g(\theta)$, respectively. $e = r - r_d$, denoting the tracking error between the desired trajectory and the actual trajectory achieved by the manipulator, and \dot{e} is derivative of e , $\dot{e} = J\dot{\theta} - \dot{r}_d$.

Remark 1. For comparison, we give the velocity level QP formulation without considering the joint acceleration and torque limits in [12], as follows:

$$\min_{\dot{\theta}} \dot{\theta}^T \left(\frac{\dot{\theta}}{2} \right), \quad (11a)$$

$$\text{s.t. } J\dot{\theta} = \dot{r}_d - k_1(r - r_d), \quad (11b)$$

$$\max\left(k_2(\theta^- - \theta), \dot{\theta}^-\right) \leq \dot{\theta} \leq \min\left(\dot{\theta}^+, k_2(\theta^+ - \theta)\right), \quad (11c)$$

where the minimization of the joint velocity norm is chosen as the cost function because the kinematics of redundant manipulator is solved in velocity level. Parameters k_1 is similar to α, β, γ in equation (10), $k_2 > 0 \in \mathbb{R}$.

3. RNN Solver

In this section, we will design a dynamic neural solver to solve equation (10) based on RNN. Then, the theoretical proof is given that under the designed solver, the optimal solution of equation (10) can be found.

To solve equation (10), a Lagrange function is defined as follows:

$$L = \ddot{\theta}^T \frac{\ddot{\theta}}{2} + \lambda_1^T (b_0 - J\ddot{\theta}) + \lambda_2^T (M(\theta)\ddot{\theta} - b_1) + \lambda_3^T (-M(\theta)\ddot{\theta} - b_2), \quad (12)$$

where λ_1 , λ_2 , and λ_3 are the Lagrange multiplier. Based on the KKT conditions, the optimal solution of equation (12) can be equivalently rewritten as follows:

$$\ddot{\theta} = P_\Omega \left(\ddot{\theta} - \frac{\partial L}{\partial \ddot{\theta}} \right), \quad (13a)$$

$$J\ddot{\theta} = b_0, \quad (13b)$$

$$\begin{cases} \lambda_2 = 0, & \text{if } M(\theta)\ddot{\theta} \leq b_1, \\ \lambda_2 > 0, & \text{otherwise,} \end{cases} \quad (13c)$$

$$\begin{cases} \lambda_3 = 0, & \text{if } -M(\theta)\ddot{\theta} \leq b_2, \\ \lambda_3 > 0, & \text{otherwise,} \end{cases} \quad (13d)$$

where P_Ω is a projection operation to a set Ω , and $P_\Omega(x) = \arg \min_{y \in \Omega} \|y - x\|$ [41]. Equations (13c) and (13d) can be further written as follows:

$$\begin{aligned} \lambda_2 &= \max \left(\left(\lambda_2 + M(\theta)\ddot{\theta} - b_1 \right), 0 \right), \\ \lambda_3 &= \max \left(\left(\lambda_3 - M(\theta)\ddot{\theta} - b_2 \right), 0 \right). \end{aligned} \quad (14)$$

The designed RNN controller is as follows:

$$\epsilon \ddot{\theta} = -\ddot{\theta} + P_\Omega (J^T \lambda_1 - M(\theta)^T \lambda_2 + M(\theta)^T \lambda_3), \quad (15a)$$

$$\epsilon \dot{\lambda}_1 = b_0 - J\ddot{\theta}, \quad (15b)$$

$$\epsilon \dot{\lambda}_2 = \max \left(\left(M(\theta)\ddot{\theta} - b_1 + \lambda_2 \right), 0 \right) - \lambda_2, \quad (15c)$$

$$\epsilon \dot{\lambda}_3 = \max \left(\left(-M(\theta)\ddot{\theta} - b_2 + \lambda_3 \right), 0 \right) - \lambda_3, \quad (15d)$$

where ($\epsilon > 0$) is a constant which is used to scale the convergence rate of the neural network. In general, the smaller ϵ , the faster the RNN controller converges.

Rewrite the designed RNN controller equation (15) as follows:

$$\epsilon \begin{bmatrix} \ddot{\theta} \\ \dot{\lambda}_1 \\ \dot{\lambda}_2 \\ \dot{\lambda}_3 \end{bmatrix} = \begin{bmatrix} -\ddot{\theta} + P_\Omega (\ddot{\theta} + \delta - \ddot{\theta}) \\ -\lambda_1 + (\lambda_1 + b_0 - J\ddot{\theta}) \\ -\lambda_2 + (\lambda_2 + M(\theta)\ddot{\theta} - b_1) \\ -\lambda_3 + (\lambda_3 - M(\theta)\ddot{\theta} - b_2) \end{bmatrix}, \quad (16)$$

where $\delta = J^T \lambda_1 - M(\theta)^T \lambda_2 + M(\theta)^T \lambda_3$.

Let $\eta = [\ddot{\theta}^T, \lambda_1^T, \lambda_2^T, \lambda_3^T]^T$, equation (16) is reformulated as follows:

$$\epsilon \dot{\eta} = -\eta + P_\Omega (\eta - \mathcal{F}(\eta)), \quad (17)$$

in which

$$\mathcal{F}(\eta) = \begin{bmatrix} \ddot{\theta} - J^T \lambda_1 + M(\theta)^T \lambda_2 - M(\theta)^T \lambda_3 \\ J\ddot{\theta} - b_0 \\ -M(\theta)^T \ddot{\theta} - b_1 \\ M(\theta)^T \ddot{\theta} + b_2 \end{bmatrix}, \quad (18)$$

Therefore, the gradient of \mathcal{F} is as follows:

$$\nabla \mathcal{F} = \frac{\partial \mathcal{F}}{\partial \eta} = \begin{bmatrix} I & -J^T & M(\theta)^T & -M(\theta)^T \\ J & 0 & 0 & 0 \\ -M(\theta)^T & 0 & 0 & 0 \\ M(\theta)^T & 0 & 0 & 0 \end{bmatrix}, \quad (19)$$

$$\nabla \mathcal{F}(\eta) + \nabla \mathcal{F}^T(\eta) = \begin{bmatrix} 2I & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \text{ is positive semidefinite;}$$

therefore $\mathcal{F}(\eta)$ is a monotone function. Following [9], it can be said that the constructed RNN controller equation (15) will globally converge to an equilibrium that is equivalent to the optimal solution of equation (10).

4. Numerical Experiments

In this part, numerical experiments are performed based on MATLAB ROBOTIC TOOLBOX. The effectiveness and robustness of the proposed QP control scheme equation (10) are shown by a six-DOFs PUMA 560 manipulator and a four-DOFs planer manipulator to show application feasibility of the proposed scheme equation (10) in all kinds of redundant robot manipulators.

4.1. PUMA 560 Manipulator Experiments. In this experiment, two different trajectory tracking experiments, i.e., the butterfly and triangle trajectories, are conducted. Table 2 shows DH parameters of the used PUMA 560 redundant manipulator. Other parameters involved in the simulation experiment are concluded in Table 3, where $\theta(0)$, $\dot{\theta}(0)$, $\ddot{\theta}(0)$ denote the initial value of θ , $\dot{\theta}$, $\ddot{\theta}$ at the initial instant, respectively. The simulation duration T_d is 20 s.

4.1.1. Butterfly Trajectory Tracking. Assume that the manipulator is required to track a butterfly trajectory with the following definition:

$$r_d = \begin{bmatrix} 0.05 \sin(0.5t) (\exp(a) - 2b) + 0.4521 \\ 0.05 \cos(0.5t) (\exp(a) - 2b) - 0.3 \\ 0.6318 \end{bmatrix}, \quad (20)$$

TABLE 2: The D-H parameter of the redundant manipulator PUMA 560 used in this paper.

Link	a (m)	α (rad)	d (m)
1	0	$(\pi/2)$	0
2	0.4318	0	0
3	0.0203	$-(\pi/2)$	0.15005
4	0	$(\pi/2)$	0.4318
5	0	$-(\pi/2)$	0
6	0	0	0.2

TABLE 3: Parameters involved in the simulative experiment.

Parameters	Value	Parameters	Value
$\theta(0)$	$0 \in \mathbb{R}^6$	θ^+	+4
$\dot{\theta}(0)$	$0 \in \mathbb{R}^6$	θ^-	-4
$\ddot{\theta}(0)$	$0 \in \mathbb{R}^6$	$\dot{\theta}^+$	+4
(ε)	0.002	$\dot{\theta}^-$	-4
α	21	$\ddot{\theta}^+$	+4
β	7	$\ddot{\theta}^-$	-4
γ	21	θ	-4
κ_1	20	ϑ	0.1
κ_2	20	τ^+	40
		τ^-	-40

where $a = \cos(0.5t)$, $b = \cos(2t)$. Firstly, a comparison between the velocity level scheme equation (11) and the acceleration level scheme equation (10) is conducted with corresponding simulative results illustrated in Figure 1. Figures 1(a)–1(f) represent results achieved by equation (11). Figures 1(g)–1(l) represent the ones achieved by equation (10). From left to right are 3D tracking results, planer tracking results, tracking error profiles at x -axis, y -axis, and z -axis, joint speed profiles, joint acceleration profiles, and joint torque profiles consecutively. Note that in the velocity level experiment, parameters k_1 and k_2 are valued as 10 and 20, respectively. We observe from Figures 1(a) and 1(b) that the actual trajectory (black color path) achieved by the manipulator successfully coincides with the desired butterfly path (red color path). It can also be observed from Figure 1(c) that the deviations between the desired trajectory and the actual trajectory at x -axis, y -axis, and z -axis quickly reduces to zero. Following Figure 1(d), the manipulator’s six joint speeds are smooth and within the respective constraint ranges. For now, equation (11) without considering the joint acceleration limit seems to be effective for the kinematic control of the PUMA 560 manipulator. However, in this case that does not consider the joint acceleration limit, the manipulator’s acceleration and torque profiles are extremely high (up to $-200(\text{rad}/\text{s}^2)$ and -600Nm , respectively) at the beginning, as shown in Figures 1(e) and 1(f). It is unreasonable in practice, obviously. Compared to the velocity level scheme equations (11) and (10) taking the joint acceleration and the joint torque limits into account, the kinematics problem is solved in acceleration level. Following the simulation results illustrated in Figures 1(g)–1(l), what the naked eye can see is that the manipulator does not only track the desired butterfly trajectory, its joint speed, joint acceleration, and torque are

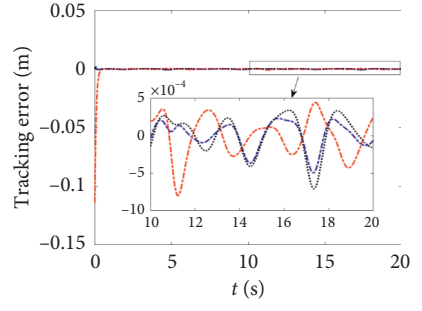
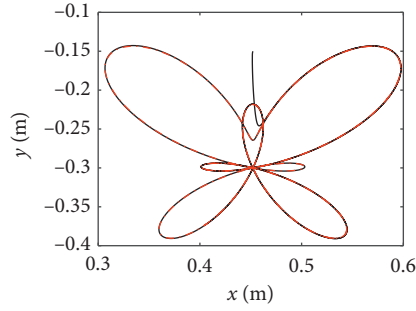
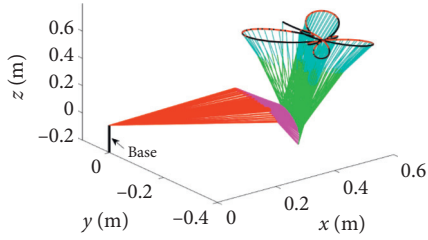
all restricted to the constrained range. As shown in Figure 1(k), when the acceleration of the joint 1 exceed its lower bound, the joint acceleration would be restricted and maintained in its lower bound, avoiding the high acceleration and joint torque. Therefore, it is concluded that the acceleration level optimization scheme considering the joint acceleration and torque limits is superior to the velocity level for the butterfly path tracking.

4.1.2. Triangle Trajectory Tracking. To further validate the effectiveness and superiority of the proposed hybrid acceleration level scheme equation (10), in this experiment, the manipulator is required to track a triangle trajectory with the following definition:

$$r_d = \begin{bmatrix} 0.075 \sin(\rho) - 0.0325 \cos(2\rho) + 0.48 \\ 0.075 \cos(\rho) - 0.0325 \sin(2\rho) - 0.10 \\ 0.6 \end{bmatrix}, \quad (21)$$

where $\rho = (2\pi t/10)$. Comparative results between the scheme equation (11) and the scheme equation (10) are illustrated in Figure 2. Figures 2(a)–2(f) represent results achieved by equation (11). Others represent ones achieved by equation (10). The simulation environment and parameters in the triangle trajectory tracking experiment are the same as the butterfly trajectory tracking experiment. Similar to the previous butterfly experiment, we observe from Figures 2(a)–2(f) that the manipulator successfully tracks the desired triangle path with a tracking error being 10^{-4} level, and the manipulator’s six joint speeds are smooth and within the respective constraint ranges. However, the manipulator’s acceleration and torque profiles that are shown in Figures 2(e) and 2(f) are extremely high (up to $(200 \text{ rad}/\text{s}^2)$ and 600 Nm , respectively) at the beginning. When both the inherent joint acceleration limit and torque limit are considered, the manipulator does not only track the desired triangle trajectory. Its joint speed, joint acceleration, and torque are all restricted to the respective constrained ranges. As shown in Figure 2(k), when the acceleration of joint 1 exceeds its upper bound, the joint acceleration would be restricted and maintained in the upper bound, avoiding the high acceleration and joint torque. Therefore, it is concluded that the acceleration level optimization scheme equation (10) considering the joint acceleration and torque limits is superior to the velocity level scheme equation (11). This experiment validates the effectiveness of the proposed hybrid scheme equation (10) and the RNN controller equation (15) for butterfly trajectory tracking task.

4.1.3. Robustness Comparison. In real scenes, the external disturbance is unavoidable and may influence the tracking accuracy of the manipulator in the trajectory tracking mission. To this, we propose a robust hybrid QP minimization scheme equation (10) by introducing integration-enhanced feedback to reject these interferences. In general, these disturbances can be mathematically described as constant-valued noise and time-varying noise.



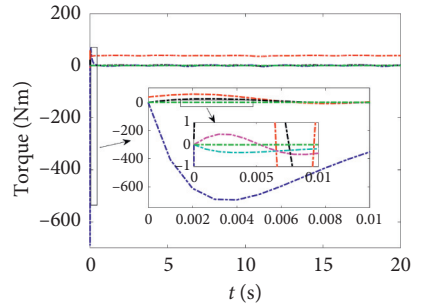
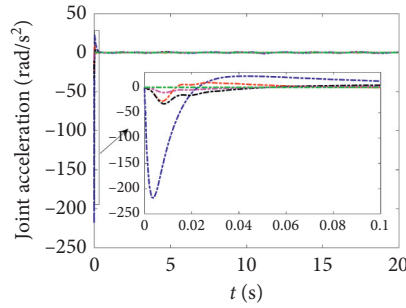
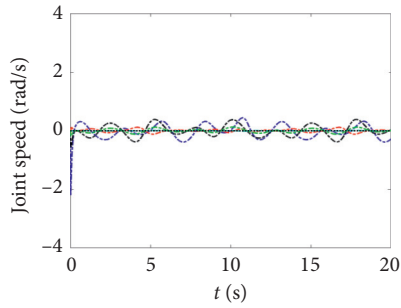
— Actual trajectory
 - - - Desired trajectory

- - - e_x
 - - - e_y
 ····· e_z

(a)

(b)

(c)



- - - $\dot{\theta}_1$ - - - $\dot{\theta}_4$
 - - - $\dot{\theta}_2$ - - - $\dot{\theta}_5$
 - - - $\dot{\theta}_3$ ····· $\dot{\theta}_6$

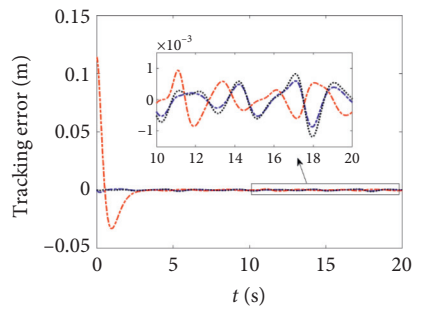
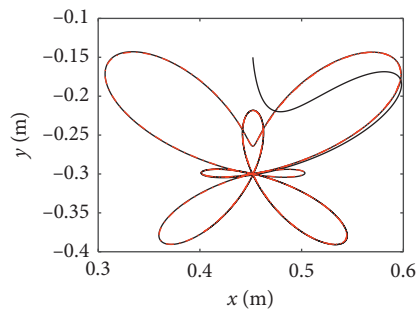
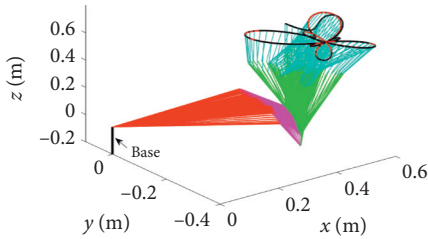
- - - $\ddot{\theta}_1$ - - - $\ddot{\theta}_4$
 - - - $\ddot{\theta}_2$ - - - $\ddot{\theta}_5$
 - - - $\ddot{\theta}_3$ - - - $\ddot{\theta}_6$

- - - τ_1 - - - τ_4
 - - - τ_2 - - - τ_5
 - - - τ_3 - - - τ_6

(d)

(e)

(f)



— Actual trajectory
 - - - Desired trajectory

- - - e_x
 - - - e_y
 ····· e_z

(g)

(h)

(i)

FIGURE 1: Continued.

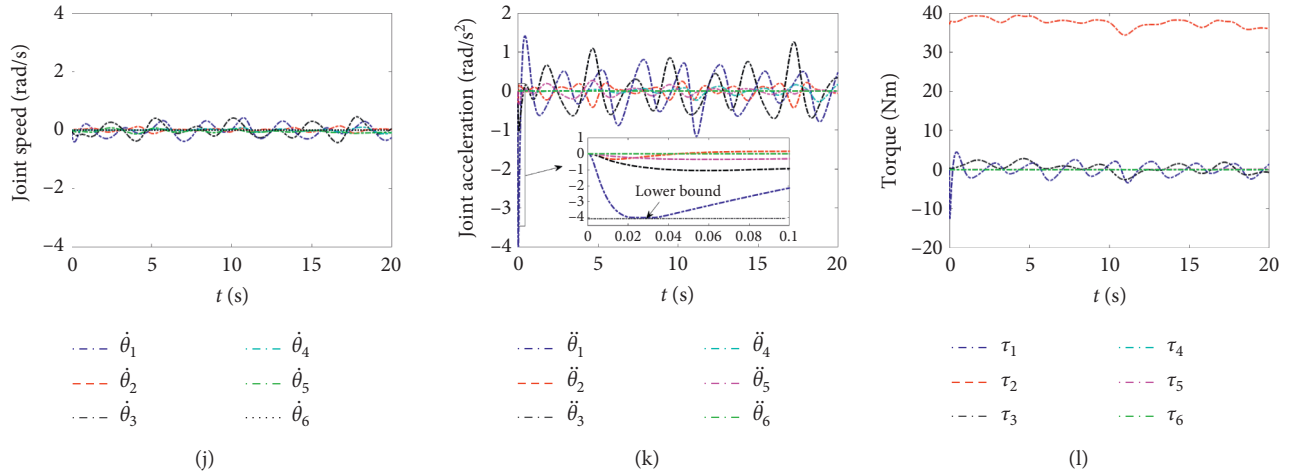


FIGURE 1: Comparison between the velocity level scheme equation (11) (a–f) and the acceleration scheme equation (10) (g–l) when the manipulator is expected to follow a butterfly path. From left to right are 3D tracking results, planer tracking results, tracking error profiles, joint speed profiles, joint acceleration profiles, and joint torque profiles.

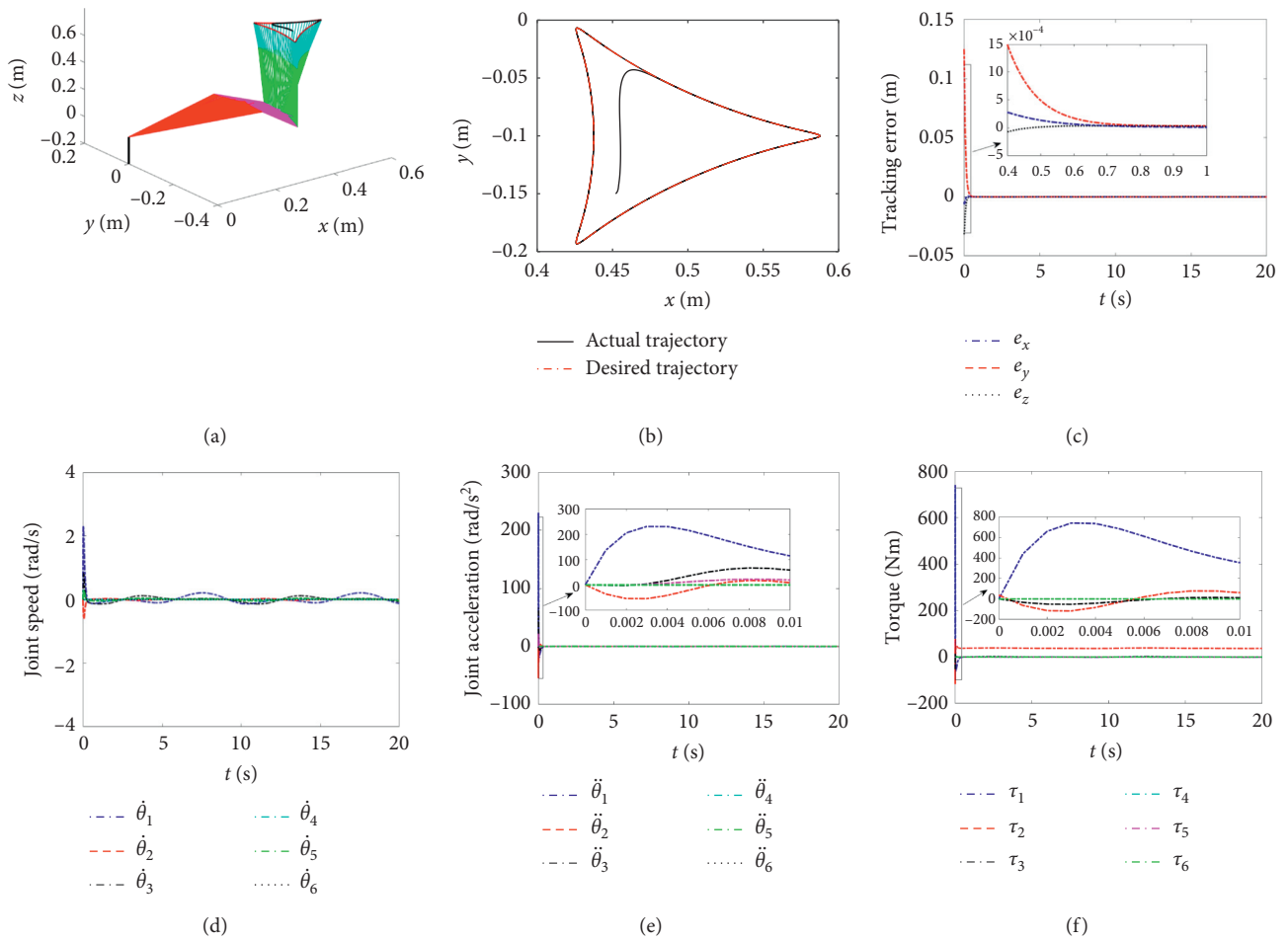


FIGURE 2: Continued.

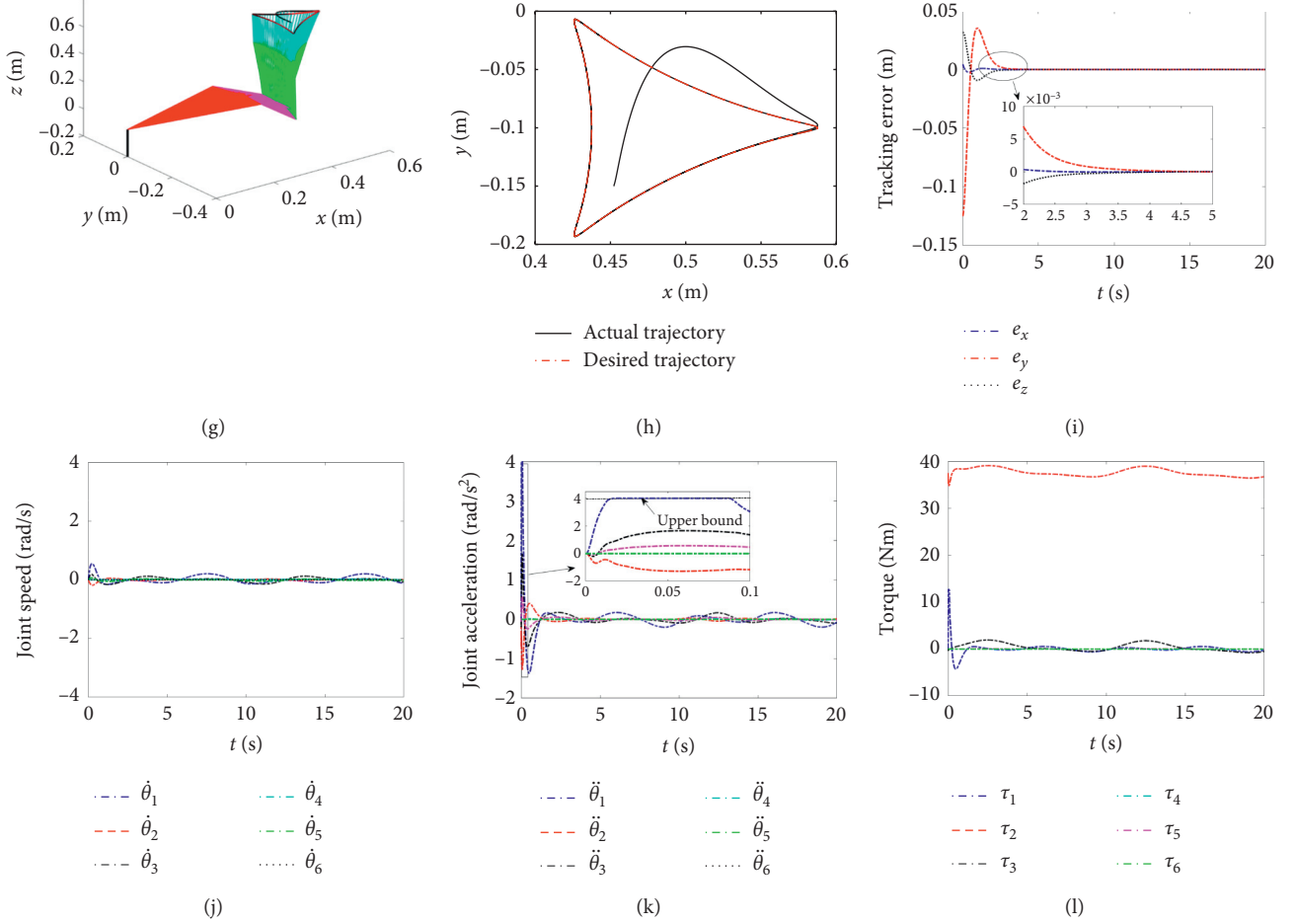


FIGURE 2: Comparison between the velocity level scheme equation (11) (a–f) and the acceleration scheme equation (10) (g–l) when the manipulator is expected to follow a triangle path. From left to right are 3D tracking results, planer tracking results, tracking error profiles, joint speed profiles, joint acceleration profiles, and joint torque profiles.

In this part, we would show the robustness of the scheme equation (10) against external disturbances, where four common types of time-varying external disturbances are considered. They are linear disturbances denoting offset errors in robot hardware implementation, sine-form disturbances caused by signal electromagnetic interference in the process of signal processing and transmission, the exponential-decay-form disturbance, and random disturbance caused by the change of the external environment, respectively. For comparison, the conventional acceleration level scheme is given as follows:

$$\min_{\ddot{\theta}} \ddot{\theta}^T \left(\frac{\ddot{\theta}}{2} \right), \quad (22a)$$

$$\text{s.t. } J\ddot{\theta} = \dot{r}_d - \dot{J}\dot{\theta} - \delta_1 e - \delta_2 \dot{e} + \omega, \quad (22b)$$

$$M(\theta)\ddot{\theta} \leq b_1, \quad (22c)$$

$$-M(\theta)\ddot{\theta} \leq b_2, \quad (22d)$$

$$\xi^+ = \min \left\{ \kappa_1 (\theta^+ - \vartheta - \theta), \kappa_2 (\dot{\theta}^+ - \dot{\theta}), \ddot{\theta}^+ \right\}, \quad (22e)$$

$$\xi^- = \max \left\{ \kappa_1 (\theta^- + \vartheta - \theta), \kappa_2 (\dot{\theta}^- - \dot{\theta}), \ddot{\theta}^- \right\}, \quad (22f)$$

where $\delta_1, \delta_2 > 0 \in \mathbb{R}$ are feedback gain parameters, which are used to scale the trajectory tracking accuracy. $\delta_1 = 21$, $\delta_2 = 7$ in the noise-rejection experiments; other parameters are the same as the ones used in equation (10). Figure 3 shows tracking results of two desired trajectories achieved by schemes (10) and (22) disturbed by the constant-valued noise and time-varying random noise, respectively. In simulative experiments, the constant-valued disturbance and time-varying random disturbance are formulated as 1 and rand, respectively, where $\text{rand} \in \{0, 1\}$. Figure 3 shows the trajectory tracking results achieved by equation (22) and our scheme equation (10). Following it, it can be observed that the conventional scheme equation (22) successfully tracks the triangle and butterfly shapes of the desired following trajectories. However, due to the disturbance by the constant-valued and time-varying random noise, the actual

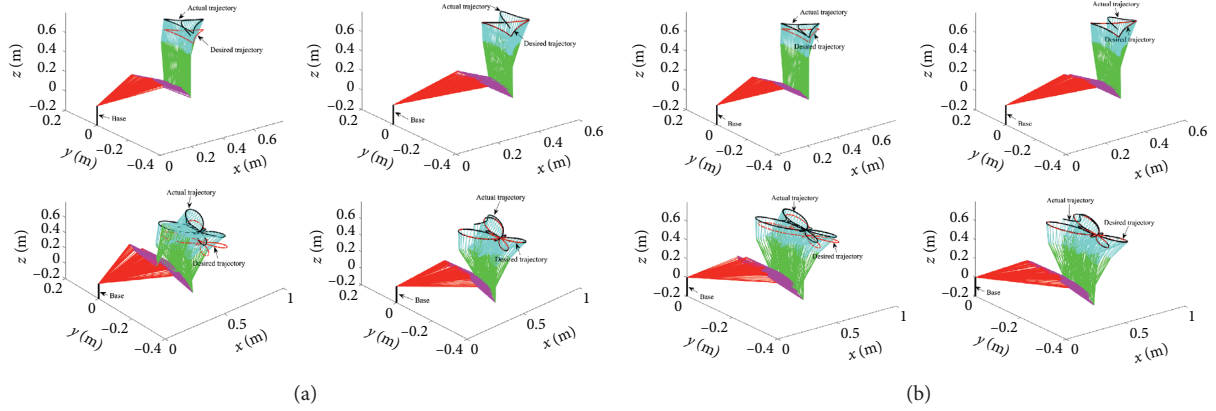


FIGURE 3: Comparison between schemes equations (22) and (10) under (a) constant-valued noise $\omega = 1$ and (b) time-varying random noise $\omega = \text{rand} \in \{0, 1\}$, respectively.

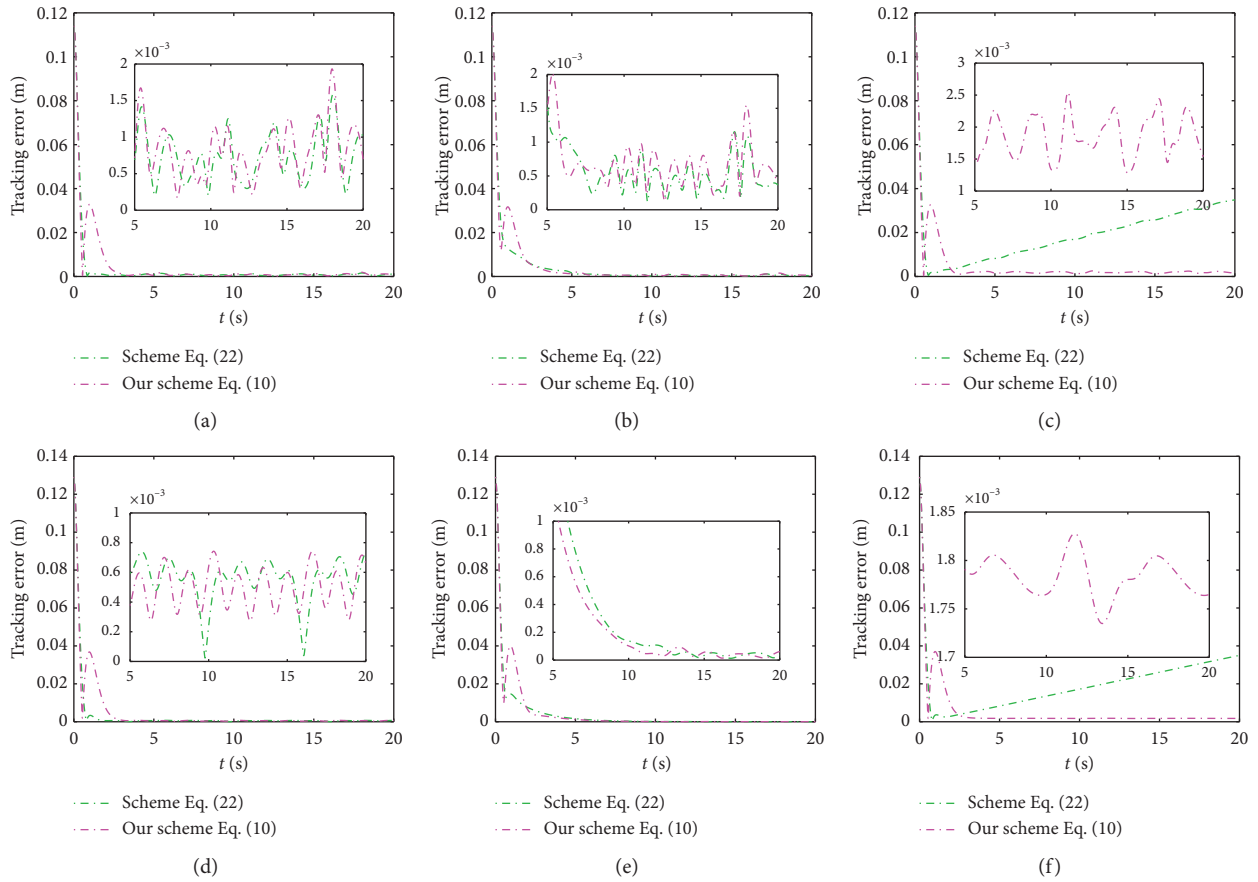


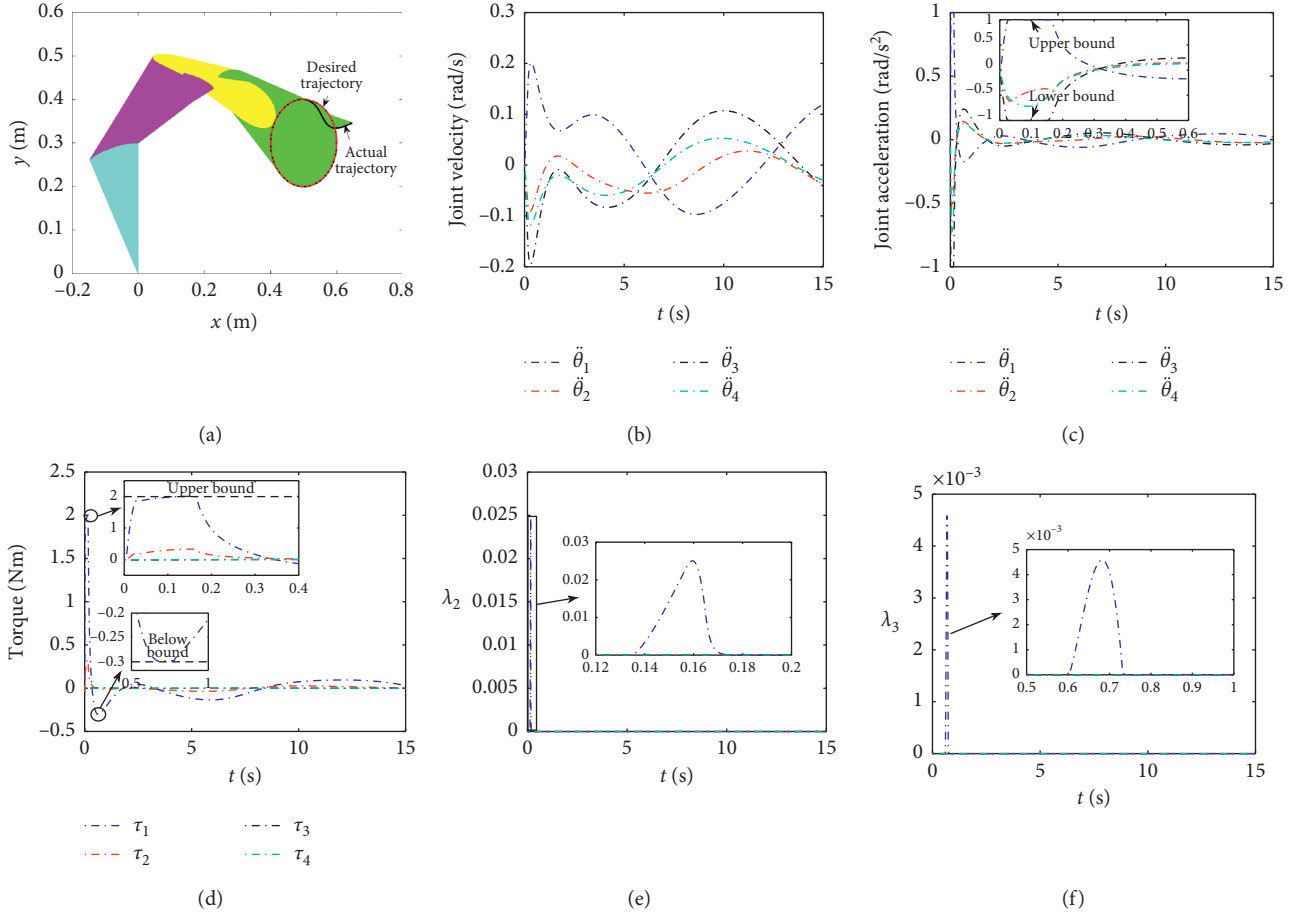
FIGURE 4: (a–f) Tracking error comparison for butterfly and triangle trajectory tracking tasks under sine-form noise, exponential noise, and linear noise, respectively. Top: butterfly trajectory. Below: triangle trajectory.

trajectories generated by the manipulator deviate from the desired position. This is to say that equation (22) fails in rejecting these two kinds of noise. On the contrary, although disturbed by the noise, our scheme equation (10) accurately and stably generates the desired triangle and butterfly trajectories.

Figure 4 shows the tracking error profiles achieved by equations (22) and (10) disturbed by the other three time-varying external noises, where the tracking error is defined as $\sqrt{e_x^2 + e_y^2 + e_z^2}$. Linear disturbances, sine disturbances, and exponential disturbances are formulated as $[0.03t, 0.02t, 0.01t]^T$,

TABLE 4: The D-H parameter of the planer redundant manipulator with 4-DOFs.

Link	a (m)	α (rad)	d (m)
1	0.3	0	0
2	0.3	0	0
3	0.2	0	0
4	0.2	0	0

FIGURE 5: Simulative results achieved by equation (10) when the manipulator is expected to follow a circle path. (a) Tracking result. (b) Joint speed profiles. (c) Joint acceleration profiles. (d) Joint torque profiles. (e) State variables λ_2 . (f) State variables λ_3 .

$[0.01 \sin(2t), 0.01 \cos(0.5t), 0.01 \sin(t)]^T$, and $[0.35 \exp(-0.5t), 0.15 \exp(-0.8t); 0.1 \exp(-t)]^T$, respectively. Based on the simulative results, we conclude that under control of the schemes equations (10) and (22), the manipulator disturbed by the sine-form noise and exponential-form noise can track the desired butterfly and triangle trajectories. The tracking error can reach 10^{-3} level. For linear disturbance, the tracking errors achieved by the scheme equation (22) would not converge to zero, as shown in Figures 4(c) and 4(f). In contrast, the tracking errors corresponding to our proposed scheme equation (10) would evaluate toward zero quickly and maintain a bounded value.

4.2. Planer Manipulator Experiments. The 4-DOFs planer redundant manipulator is employed in this experiment,

which is assumed to track a circle trajectory with the following definition:

$$r_d = \begin{bmatrix} 0.5 + 0.1 \cos(0.5t) \\ 0.3 + 0.1 \sin(0.5t) \end{bmatrix}. \quad (23)$$

Table 4 gives the DH parameters corresponding to the used planer manipulator in this study. Among the simulation setup, $\theta(0) = [(\pi/2), -(\pi/3), -(\pi/4), 0]^T \text{ rad}$, $\tau^+ = 2 \text{ Nm}$, $\tau^- = -0.3 \text{ Nm}$, $\dot{\theta}(0) = 0 \in \mathbb{R}^4 \text{ (rad/s)}$, and $\ddot{\theta}(0) = 0 \in \mathbb{R}^4 \text{ (rad/s)}^2$, $\theta^+ = -\theta^- = 1$. Other parameters remain the same as the previous experiment. Simulation duration is 15 s.

Simulative results achieved by the proposed hybrid scheme equation (10) are illustrated in Figure 5 when the manipulator is expected to follow a circle path. Similar to the

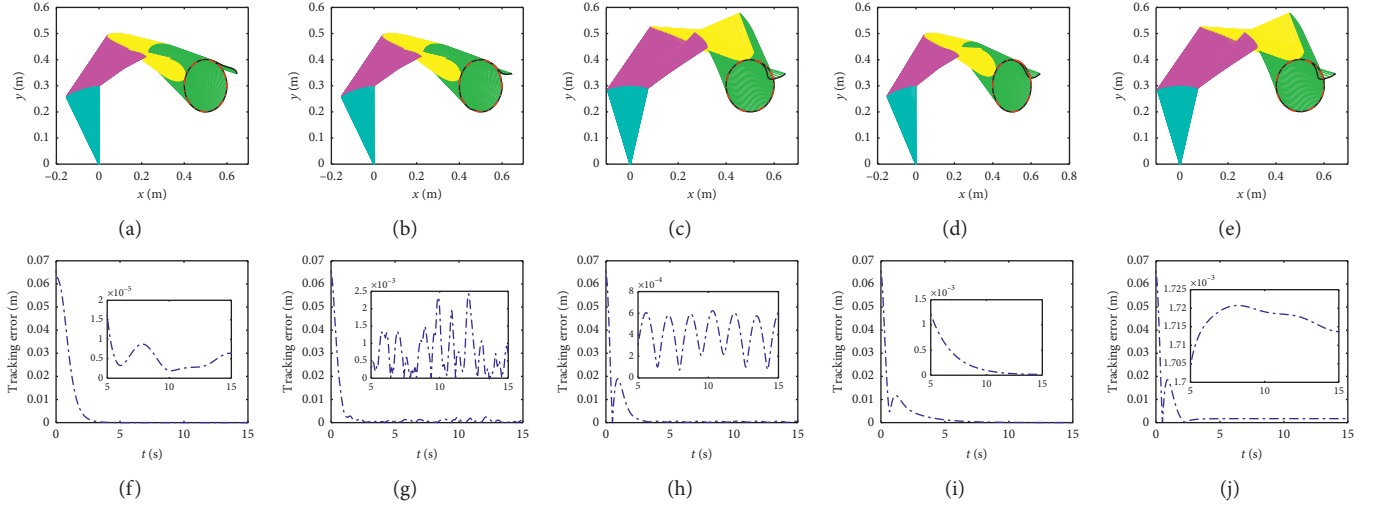


FIGURE 6: Circle trajectory tracking results and tracking errors achieved by equation (10) disturbed by the external disturbances. (a) Constant-valued noise $\omega = 1$. (b) Random noise $\omega \in \{0, 1\}$. (c) Sine-form noise $\omega = [0.01 \sin(2t), 0.01 \cos(0.5t)]^T$. (d) Exponential-form noise $\omega = [0.35 \exp(-0.5t), 0.15 \exp(-0.8t)]^T$. (e) Linear noise $\omega = [0.03t, 0.02t]^T$.

previous experiment, the actual trajectory achieved by the planar manipulator quickly coincides with the desired path, simultaneously avoiding high joint velocity, acceleration, and torque values. Following Figures 5(c) and 5(d), when joint acceleration and joint torque vectors exceed their respective bound, they would be restrained within the upper or lower bound. Figures 5(e) and 5(f) show the process profiles corresponding to state variables λ_2 and λ_3 . When the inequality constraints equations (10c) and (10d) are not satisfied, i.e., joint torque bounds of the manipulator are not reached, λ_2 and λ_3 would remain on zero. On the contrary, λ_2 and λ_3 would be greater than zero. Figure 6 shows circle trajectory tracking results achieved by equation (10) disturbed by five external noises and the corresponding tracking errors. It can be seen from it that for the considered five noises, the manipulator successfully tracks the desired circle trajectory with different tracking accuracies. Compared to other types of noises, the influence of random noise on the trajectory tracking error of the manipulator is stronger.

Based on Figures 1–6, we can conclude that under the proposed hybrid scheme equation (10), the manipulator can accurately track the desired trajectory, simultaneously avoiding high joint velocity, acceleration, and joint torque. In addition, equation (10) is robust against the constant-valued external disturbance and four types of time-varying external disturbances.

5. Conclusion

In this paper, a robust multiobjectives-integrated hybrid scheme equation (10) has been proposed and used for controlling a redundant manipulator to track the desired trajectory. Under equation (10), high joint velocity, acceleration, and joint torque may exceed the manipulator actuators' ability that can be avoided. They all are restrained on respective constraint bound. In addition, integrated-enhanced feedback has been generalized into the inverse

kinematics of the manipulator to reject the influence of external disturbances on the tracking tasks. Numerical experiments have been performed on the PUMA 560 manipulator and the planar manipulator, validating the effectiveness and robustness of the hybrid scheme equation (10).

Multiple robots cooperative control is being investigated and has made success in environmental monitoring [42] and source seeking [43]. In future work, we will be devoted to the motion-force hybrid cooperative control of multiple robots at dynamics level.

Appendix

In this appendix, the authors give the derivation process of equation (9). The velocity level inverse kinematics of the manipulator is usually described as follows [6, 9, 12]:

$$\dot{r} = \dot{r}_d - \mu(r - r_d), \quad (\text{A.1})$$

where $\mu > 0$ is a constant, which is used to scale accuracy of the trajectory tracking task.

Define a new error function $\ell = \dot{r} - \dot{r}_d + \mu(r - r_d)$. Based on [38, 39], the time-derivative evolution formulation of the error function ℓ can be constructed as follows:

$$\dot{\ell} = -\mu_1 \ell - \mu_2 \int_0^t \ell dt, \quad (\text{A.2})$$

where the same as $\mu, \mu_1, \mu_2 > 0 \in \mathbb{R}$. Differentiating ℓ , we obtain that

$$\dot{\ell} = \ddot{r} - \ddot{r}_d + \mu(\dot{r} - \dot{r}_d). \quad (\text{A.3})$$

Combining equations (A.2) with (A.3), we have the following:

$$\begin{aligned} \ddot{r} &= \ddot{r}_d - \mu(\dot{r} - \dot{r}_d) - \mu_1(\dot{r} - \dot{r}_d) - \mu_1\mu(r - r_d) \\ &\quad - \mu_2 \int_0^t (\dot{r} - \dot{r}_d) dt - \mu_2\mu \int_0^t (r - r_d) dt. \end{aligned} \quad (\text{A.4})$$

It can be further rewritten as follows:

$$\ddot{r} = \ddot{r}_d - (\mu + \mu_1) \left(\dot{r} - \dot{r}_d \right) - (\mu_1 \mu + \mu_2) (r - r_d) - \mu_2 \mu \int_0^t (r - r_d) dt. \quad (\text{A.5})$$

Because μ, μ_1, μ_2 are constants >0 , therefore, equation (A.5) can be converted as follows:

$$J\ddot{\theta} = \ddot{r}_d - J\dot{\theta} - \alpha(r - r_d) - \beta(J\dot{\theta} - \dot{r}_d) - \gamma \int_0^t (r - r_d) dt, \quad (\text{A.6})$$

by using α, β, γ to replace $\mu_1 \mu + \mu_2, \mu + \mu_1, \mu_2 \mu$, respectively. Li et al. have proved the inherent noise tolerance of equation (A.2) to constant-valued and time-varying disturbances in [38]. Therefore, robustness proof of equation (A.6) disturbed by external disturbances (i.e., equation (9)) is omitted here.

Data Availability

No data were used to support this study.

Conflicts of Interest

Zhenzhen He is employed at the company “Guangdong OPT Co., LTD.” The remaining authors declare that there are no conflicts of interest.

Acknowledgments

This work was supported by the Dongguan Social Science and Technology Development Project (Grant no. 2019507140209), Project of Education Department of Guangdong Province (2018KTSCX224), Department of Education of Guangdong China (Grant no. 2017KZDXM082), Guangdong Province Key Areas R&D Program of Guangdong Province (Grant nos. 2019B090919002 and 2020B090925001), Key Technology Research Project of Foshan City (Grant no. 1920001001148), the Natural Science Foundation of Guangdong Province (Grant no. 2020A1515010631), Basic and Applied Basic Research Project of Guangzhou City (Grant no. 202002030237), National Natural Science Foundation of China (Grant no. 62003102), and GDAS’ Project of Science and Technology Development (2017GDASCX-0115).

References

- [1] N. Wang, C. Chen, and A. Di Nuovo, “A framework of hybrid force/motion skills learning for robots,” *IEEE Transactions on Cognitive and Developmental Systems*, p. 1, 2020.
- [2] C. Yang, C. Zeng, Y. Cong, N. Wang, and M. Wang, “A learning framework of adaptive manipulative skills from human to robot,” *IEEE Transactions on Industrial Informatics*, vol. 15, no. 2, pp. 1153–1161, 2018.
- [3] C. Zeng and C. Yang*, H. Cheng, Y. Li, and S.-L. Dai, “Simultaneously encoding movement and sEMG-based stiffness for robotic skill learning,” *IEEE Transactions on Industrial Informatics*, p. 1, 2020.
- [4] Y. Zhang, S. Li, S. Kadry, and B. Liao, “Recurrent neural network for kinematic control of redundant manipulators with periodic input disturbance and physical constraints,” *IEEE Transactions on Cybernetics*, vol. 49, no. 12, pp. 4194–4205, 2018.
- [5] Y. Zhang, J. Wang, and Y. Xia, “A dual neural network for redundancy resolution of kinematically redundant manipulators subject to joint limits and joint velocity limits,” *IEEE Transactions on Neural Networks*, vol. 14, no. 3, pp. 658–667, 2003.
- [6] Z. Xu, S. Li, X. Zhou, W. Yan, T. Cheng, and D. Huang, “Dynamic neural networks based kinematic control for redundant manipulators with model uncertainties,” *Neurocomputing*, vol. 329, pp. 255–266, 2019.
- [7] S. Li, J. He, Y. Li, and M. U. Rafique, “Distributed recurrent neural networks for cooperative control of manipulators: a game-theoretic perspective,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 2, pp. 415–426, 2016.
- [8] L. Jin, S. Li, X. Luo, Y. Li, and B. Qin, “Neural dynamics for cooperative control of redundant robot manipulators,” *IEEE Transactions on Industrial Informatics*, vol. 14, no. 9, pp. 3812–3821, 2018.
- [9] X. Li, Z. Xu, S. Li, H. Wu, and X. Zhou, “Cooperative kinematic control for multiple redundant manipulators under partially known information using recurrent neural network,” *IEEE Access*, vol. 8, pp. 40029–40038, 2020.
- [10] Z. Zhang, S. Chen, J. Xie, and S. Yang, “Two hybrid multi-objective motion planning schemes synthesized by recurrent neural networks for wheeled mobile robot manipulators,” *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, p. 1, 2019.
- [11] D. Guo and Y. Zhang, “A new inequality-based obstacle-avoidance MVN scheme and its application to redundant robot manipulators,” *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 6, pp. 1326–1340, 2012.
- [12] Z. Xu, X. Zhou, and S. Li, “Deep recurrent neural networks based obstacle avoidance control for redundant manipulators,” *Frontiers in Neurorobotics*, vol. 13, p. 47, 2019.
- [13] L. Xiao and Y. Zhang, “Acceleration-level repetitive motion planning and its experimental verification on a six-link planar robot manipulator,” *IEEE Transactions on Control Systems Technology*, vol. 21, no. 3, pp. 906–914, 2013.
- [14] Z. Zhang and Z. Yan, “Hybrid-level joint-drift-free scheme of redundant robot manipulators synthesized by a varying-parameter recurrent neural network,” *IEEE Access*, vol. 6, pp. 34967–34975, 2018.
- [15] Y. Zhang, S. S. Ge, and T. H. Lee, “A unified quadratic-programming-based dynamical system approach to joint torque optimization of physically constrained redundant manipulators,” *IEEE Transactions on Systems, Man and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 5, pp. 2126–2132, 2004.
- [16] S. Liu and J. Wang, “Bi-criteria torque optimization of redundant manipulators based on a simplified dual neural network,” in *Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN)*, pp. 2796–2801, Montreal, Canada, August 2005.
- [17] J. Hollerbach and K. Ki Suh, “Redundancy resolution of manipulators through torque optimization,” *IEEE Journal on Robotics and Automation*, vol. 3, no. 4, pp. 308–316, 1987.
- [18] K. Kazerooni and A. Nedungadi, “Redundancy resolution of robotic manipulators at the acceleration level,” in *Proceedings of the World Congress on Theory of Machines & Mechanisms (IFTOMM)*, Sevilla, Spain, September 1987.

- [19] W. S. Tang and J. Wang, "Two recurrent neural networks for local joint torque optimization of kinematically redundant manipulators," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 30, no. 1, pp. 120–128, 2000.
- [20] Y. Zhang and J. Wang, "A dual neural network for constrained joint torque optimization of kinematically redundant manipulators," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 32, no. 5, pp. 654–662, 2002.
- [21] Y. Zhang, N. Tan, and C. Lai, "Bi-criteria torque minimization of redundant robot arms with schemes, models and methods compared," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 2397–2402, Guilin, China, December 2009.
- [22] Z. Xu, S. Li, X. Zhou, S. Zhou, and T. Cheng, "Dynamic neural networks for motion-force control of redundant manipulators: an optimization perspective," *IEEE Transactions on Industrial Electronics*, p. 1, 2020.
- [23] X. Wu, J. Liu, C. Huang, M. Su, and T. Xu, "3-D path following of helical micro swimmers with an adaptive orientation compensation model," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 2, pp. 823–832, 2020.
- [24] S. Li, S. Chen, B. Liu, Y. Li, and Y. Liang, "Decentralized kinematic control of a class of collaborative redundant manipulators via recurrent neural networks," *Neurocomputing*, vol. 91, pp. 1–10, 2012.
- [25] S. Li, H. Cui, Y. Li, B. Liu, and Y. Lou, "Decentralized control of collaborative redundant manipulators with partial command coverage via locally connected recurrent neural networks," *Neural Computing and Applications*, vol. 23, no. 3–4, pp. 1051–1060, 2013.
- [26] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, "Robot learning system based on adaptive neural control and dynamic movement primitives," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 30, no. 3, pp. 777–787, 2019.
- [27] H. Huang, C. Yang, and C. L. P. Chen, "Optimal robot-environment interaction under broad fuzzy neural adaptive control," *IEEE Transactions on Cybernetics*, p. 1, 2020.
- [28] L. Jin, S. Li, H. M. La, and X. Luo, "Manipulability optimization of redundant manipulators using dynamic neural networks," *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 4710–4720, 2017.
- [29] T. Xu, J. Yu, C.-I. Vong, B. Wang, X. Wu, and L. Zhang, "Dynamic morphology and swimming properties of rotating miniature swimmers with soft tails," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 3, pp. 924–934, 2019.
- [30] T. Xu, Y. Guan, J. Liu, and X. Wu, "Image-based visual servoing of helical microswimmers for planar path following," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 1, pp. 325–333, 2020.
- [31] S. Li, Y. Lou, and B. Liu, "Bluetooth aided mobile phone localization: a nonlinear neural circuit approach," *ACM Transactions on Embedded Computing Systems*, vol. 13, no. 4, 2014.
- [32] D. Guo, F. Xu, and L. Yan, "New pseudoinverse-based path-planning scheme with PID characteristic for redundant robot manipulators in the presence of noise," *IEEE Transactions on Control Systems Technology*, vol. 26, no. 6, pp. 2008–2019, 2017.
- [33] L. Jin, Y. Zhang, S. Li, and Y. Zhang, "Noise-tolerant ZNN models for solving time-varying zero-finding problems: a control-theoretic approach," *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 992–997, 2017.
- [34] D. Chen and Y. Zhang, "Robust zeroing neural-dynamics and its time-varying disturbances suppression model applied to mobile robot manipulators," *IEEE Transactions on Neural Networks & Learning Systems*, vol. 29, no. 9, pp. 1–13, 2017.
- [35] D. Chen, S. Li, Q. Wu, and X. Luo, "New disturbance rejection constraint for redundant robot manipulators: an optimization perspective," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 4, pp. 2221–2232, 2020.
- [36] D. Guo and Y. Zhang, "Acceleration-level inequality-based MAN scheme for obstacle avoidance of redundant robot manipulators," *IEEE Transactions on Industrial Electronics*, vol. 61, no. 12, pp. 6903–6914, 2014.
- [37] D. Guo, Q. Feng, and J. Cai, "Acceleration-level obstacle avoidance of redundant manipulators," *IEEE Access*, vol. 7, pp. 183040–183048, 2019.
- [38] X. Li, J. Yu, S. Li, and L. Ni, "A nonlinear and noise-tolerant ZNN model solving for time-varying linear matrix equation," *Neurocomputing*, vol. 317, pp. 70–78, 2018.
- [39] X. Li, J. Yu, S. Li, Z. Shao, and L. Ni, "A non-linear and noise-tolerant ZNN model and its application to static and time-varying matrix square root finding," *Neural Processing Letters*, vol. 50, pp. 1687–1703, 2018.
- [40] L. Jin, Y. Zhang, and S. Li, "Integration-enhanced Zhang neural network for real-time-varying matrix inversion in the presence of various kinds of noises," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 27, no. 12, pp. 2615–2627, 2016.
- [41] S. Li, Y. Zhang, and L. Jin, "Kinematic control of redundant manipulators using neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2243–2254, 2017.
- [42] S. Li, Y. Guo, and B. Bingham, "Multi-robot cooperative control for monitoring and tracking dynamic plumes," in *Proceedings of the IEEE International Conference on Robotics & Automation (ICRA)*, Hong Kong, China, June 2014.
- [43] S. Li and Y. Guo, "Distributed source seeking by cooperative robots: all-to-all and limited communications," in *Proceedings of the IEEE International Conference on Robotics & Automation (ICRA)*, pp. 1107–1112, Saint Paul, MN, USA, May 2012.

Research Article

Bottom Detection Method of Side-Scan Sonar Image for AUV Missions

Huapeng Yu ¹, Ziyuan Li,^{1,2} Dailin Li,³ and Tongsheng Shen ¹

¹National Innovation Institute of Defense Technology, Beijing, China

²Science and Technology on Underwater Vehicle Laboratory, Harbin Engineering University, Harbin, China

³College of Science, China University of Petroleum (East China), Qingdao, China

Correspondence should be addressed to Huapeng Yu; hpyu_qtxy@163.com and Tongsheng Shen; shents_bj@126.com

Received 18 September 2020; Accepted 6 October 2020; Published 17 October 2020

Academic Editor: Guang Li

Copyright © 2020 Huapeng Yu et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to obtain the measurement parameters of the sea bottom geomorphology or underwater objects, the first step in side-scan sonar (SSS) image processing is bottom detection. Due to the complexity of the marine environment, the acoustic signals received by SSS are usually polluted by noises, which affect its image quality and make the extraction of image features difficult. To address this problem, this study proposes an automatic detection method for the sea bottom line based on the actual experimental acquisition of SSS images, which is supposed to support the autonomous underwater vehicle (AUV) for intelligent target detection and classification. The proposed method comprises four main steps. First, the raw SSS data is analyzed to obtain a grayscale image, and the blind zone boundary of the image is obtained using the threshold method. Then, the noise characteristics of the image are analyzed and the denoising algorithm is optimized to effectively remove high-frequency noise. Next, spatial-temporal matching calculations are performed on each ping port and starboard data, and the accurate coordinates of first bottom returns are obtained through extreme value detection. Finally, automatic and accurate detection of the bottom line is realized according to the smooth processing of the coordinate sequence of first bottom returns. The experiments have demonstrated the effectiveness of the proposed method. As the method does not require human intervention in adjusting parameters during operation, the proposed method with a certain time window imposed during image acquisition will be suitable for AUV missions when the SSS is determined.

1. Introduction

Side-scan sonar (SSS) is an active sonar system, which mainly consists of a transducer array, peripheral auxiliary sensors, and a real-time data acquisition processing module. The transducer array is the core component of the SSS, which is usually designed to be receive-transmit combined line array [1]. SSS uses the principle of echo bathymetry to record scattered echo intensity from the sea bottom and generate sonar images according to echo intensities [1–3]. The SSS images can be utilized to observe changes in parameters such as sea bottom geomorphology, presence of obstructions, and types of sea bottom substrate, and its features have important applications in fields such as marine scientific research, marine engineering, and marine military.

In a SSS image, there is a clear junction line between the water column region and the seabed image region—called the sea bottom line, which is the distance from SSS to the sea bottom and an important parameter for measurements of the sea bottom topography or underwater objects, slant range correction, and SSS image grayscale equalization [1, 2, 4, 5].

Al-Rawi et al. [2] proposed two bottom detection methods: one to take the logarithm of the SSS echo data and use cubic spline regression algorithm for fitting and the other to filter the SSS data by sliding average filtering within a certain time window. Based on the data processed using these two methods, the center track line of the water column region is detected, and then, the first bottom return of each Ping data is obtained using the threshold method. Zhang

et al. [6] solved the problem of the weak echoes of the sea bottom line in the actual SSS data by introducing the Laplacian of the Gaussian function, which is comparable to the bottom detection by commercial SSS data processing software. However, when the water noises are rather strong, the noises will be mistakenly extracted as the sea bottom line and it is difficult to obtain accurate bottom detection results, as the Gaussian smoothing filter can only eliminate the scales of intensity changes less than the smoothing factor. In view of the interferences of the SSS image in the water column region by the transmitting pulse, the sea surface echo, the wake, and the large area suspended matter, Zhao et al. [7, 8] proposed a last-peak detection method of the bottom line. Further, to tackle with ping data loss and ping observation anomaly that often exist in SSS measurements, Zhao proposed a Kalman filtering based sea bottom line repair method, which uses the asymptotic nature of the sea bottom variation and sea bottom line symmetry on SSS both sides. Ku et al. [9] measured single beam bathymetry and real-time depth of towed body using GPS time synchronization, and then, the bottom line was obtained by combination with these two kinds of data through interpolation and smoothing. Using actual measured data, the advantages of this method compared with the method proposed by Zhao et al. [7, 8] were verified. Moreover, the bathymetry measurement on this method was relatively less influenced by the environment and its accuracy was higher. According to the spatial characteristics of the continuous and dense distribution of the sea bottom line, Wang et al. [10] constructed a point set including the bottom points along the roughly parallel direction of the track and then clustered on the chain densely distributed along the track by the “density-based spatial clustering of applications with noise (DBSCAN)” algorithm for chain searching to extract sea bottom line. Compared with the traditional bottom detection method, this method also has good stability and anti-interference ability in relatively complex environment. Yan et al. [11] used a one-dimensional convolution neural network to traverse and identify the backscatter intensity sequence in the SSS data and established a complete processing method for real-time sea bottom line detection and tracking, which has strong robustness in the field test environment.

An autonomous underwater vehicle (AUV) is capable of carrying detection equipment to accomplish automatic tasks such as marine surveys and seabed mapping [12–14]. Due to its advantages of high autonomy and low risk, AUVs are gradually being widely used, and their combinations with the SSS for underwater operations will become more frequent so that the superiority of SSS in target positioning and identification can be fully exploited [2, 12, 14–16]. However, few of the published research studies have mature engineering applications for the automatic bottom detection in SSS images suitable for AUV missions.

One of the main objectives of this study is to develop an automatic bottom detection method in SSS images suitable for AUV missions. The remainder of this paper is arranged as follows. Section 1 briefly introduces the characteristics of an SSS image and detailedly analyzes the published research studies and results of bottom detection. In Section 2, the

current problems of bottom detection in SSS image for AUV missions are stated. Section 3 elaborates the proposed bottom detection method and verifies its effectiveness using the SSS actual test data on an AUV. Section 4 summarizes this study.

2. Preliminaries and Problem Statements

For AUV missions, there are two main deficiencies in the current research on SSS bottom detection.

- (1) Under the hypothesis that there is either small and negligible or no blind zone between the port and starboard sonar transducer arrays, the installation of SSS is always taken as an ideal geometry. However, in actual working environments, the sonar transducers transmit sound waves to both sides, so there is a blind zone in the sonar image, and the space occupied by the blind zone in the sonar image should not be ignored. To avoid destroying the true geometry of the seabed image, it is necessary to consider eliminating the effects of the blind zone in the water column area.
- (2) The published research results were obtained through complex postprocessing processes [7–9] or high online computing capability for AUV [10, 11]. Moreover, due to the complex influence factors of SSS images, manual intervention is often required to set the initial value according to the situation, and most mature SSS image processing software, such as Triton and Prism, can only achieve reliable semiautomatic bottom detection [7, 10, 12, 17].

3. Automatic SSS Bottom Detection

3.1. Automatic Bottom Detection Procedure. Figure 1 shows the designed automatic bottom detection procedure for the SSS suitable for AUV missions. The procedure is described in four main steps.

3.1.1. Step 1. Obtaining the Blind Zone Boundary according to the SSS Parameters. In view of the geometry problem described in Section 2, first, it is necessary to calculate the parameters of the blind zone.

Since the blind zone is located directly below the SSS, the echo intensities of each ping are mutated from the blind zone to the outside, and the numerical values drop precipitously (see Figure 2). According to this feature, blind zone boundary detection can be realized. In general, the echo directly below the SSS is a strong positive emission wave, so a threshold can be used to detect the blind zone boundary detection.

Given a SSS, the ping data format is fixed (i.e., to 8, 16, or 32 bits). If the ping data bits of the SSS is N , then the threshold value for the blind zone boundary detection can be determined according to the measurement environment as follows:

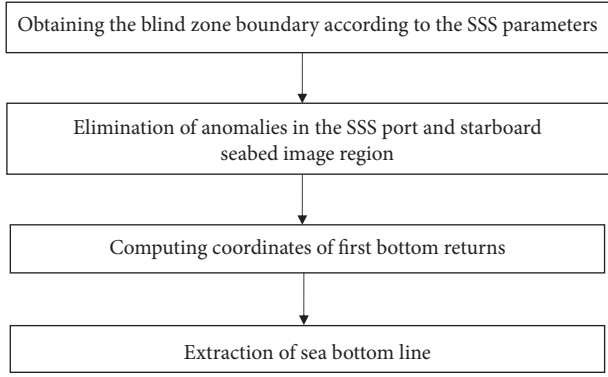


FIGURE 1: Automatic bottom detection procedure for SSS.

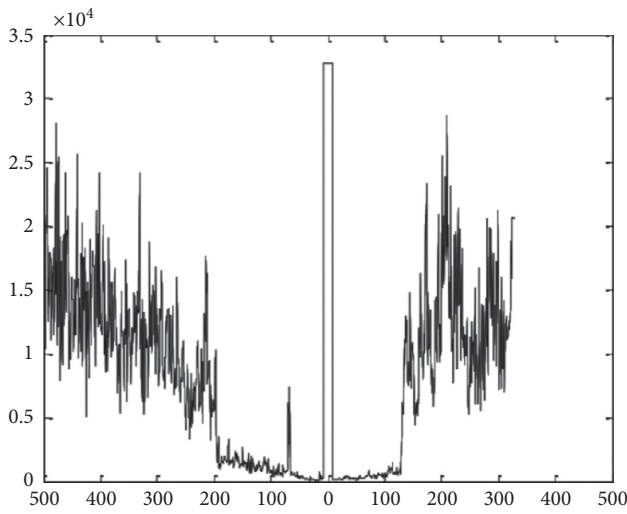


FIGURE 2: SSS waterfall image.

$$D_m = 2^{N-2}. \quad (1)$$

From the blind zone to the outside, the first echo intensities of each ping on the port and starboard sides that are less than the threshold value are denoted as K_{left} and K_{right} , respectively. As the transducer arrays are mounted symmetrically on the port and starboard sides of the AUV with respect to its center axis, the width of the blind zone of each ping, K_m , according to the principle of bilateral symmetry, is determined as follows:

$$K_m = \begin{cases} \frac{K_{\text{left}} + K_{\text{right}} - 2}{2}, & (K_{\text{left}} + K_{\text{right}}) \text{ is even,} \\ \frac{K_{\text{left}} + K_{\text{right}} - 1}{2}, & (K_{\text{left}} + K_{\text{right}}) \text{ is odd.} \end{cases} \quad (2)$$

For each ping, the maximum sampling numbers of AUV left and right echo data are K_{leftmax} and K_{rightmax} , respectively. Thus, the left echo data $[K_m, K_{\text{leftmax}}]$ and right echo data $[K_m, K_{\text{rightmax}}]$ of all pings, which imply the seabed image region, can be obtained.

3.1.2. Step 2. Elimination of Anomalies in the SSS Port and Starboard Seabed Image Region. The median filtering algorithm can be used to remove the anomalies in SSS echo data, that is, the high-frequency noise and outliers [6, 9]. Herein, anomalies in both port and starboard SSS echo data are eliminated with an improved median filtering algorithm, and its calculation flow is as follows:

- (1) Calculate the intensity median of every sampling echo in each ping of the port and starboard SSS data.
- (2) Calculate the absolute intensity deviation of every sampling echo in each ping and determine the intensity value for the corresponding sampling echo based on its absolute intensity deviation. When the absolute intensity deviation is greater than a set value, the intensity median of the said sampling echo is taken as its new intensity value. Otherwise, no anomaly removal is performed and the intensity value for the said sampling echo is its sampling measured value. The above set value is a calculated value based on the echo intensity sampling values in each ping.

For example, for the SSS port echo data $[K_m, K_{\text{leftmax}}]$, let $[i, j]$ indicate the index of the j th echo sampling measure of the i th ping, and let I_{ij} and P_{max} denote the intensity value of the $[i, j]$ th port echo data and the total number of SSS pings, respectively. Then, for all $[i, j]$ combinations satisfying $1 < i < P_{\text{max}}$ and $K_m \leq j < K_{\text{leftmax}}$, the following processing is performed:

- (1) Calculate the intensity median Z_{ij} using the $[i-1, j-1]$ th, $[i-1, j]$ th, $[i-1, j+1]$ th, $[i, j-1]$ th, $[i, j]$ th, $[i, j+1]$ th, $[i+1, j-1]$ th, $[i+1, j]$ th, and $[i+1, j+1]$ th sampling measured intensity values in port echo data, as well as the absolute intensity deviation $E_{ij} = \text{abs}(I_{ij} - Z_{ij})$.

It is obvious from the above example that the number of adjacent pings and the number of adjacent echo sampling measures in each ping are both determined according to the actual needs. The same can be said for the numerical calculation of the intensity value of every sampling echo in each ping of the starboard SSS data, which is described subsequently.

- (2) Deciding absolute intensity deviation E_{ij} [18]: if $E_{ij} > 9.78 + 1.44 \times I_{ij} - 8.86 \times 10^{-4} \times I_{ij}^2 + 2.95 \times 10^{-7} \times I_{ij}^3 - 3.26 \times 10^{-11} \times I_{ij}^4$, then Z_{ij} is used as the new intensity value for the $[i, j]$ th echo sampling measure in SSS port data; otherwise, no anomaly removal is performed.

For the SSS starboard echo data $[K_m, K_{\text{rightmax}}]$, let $[i, l]$ indicate the index of the l th echo sampling measure of the i th ping, and let I_{il} denote the intensity value of the $[i, l]$ th starboard echo data. Then, for all $[i, l]$ combinations satisfying $1 < i < P_{\text{max}}$ and $K_m \leq l < K_{\text{rightmax}}$, the following processing is performed:

- (1) Calculate the intensity median Z_{il} using the $[i-1, l-1]$ th, $[i-1, l]$ th, $[i-1, l+1]$ th, $[i, l-1]$ th, $[i, l]$ th, $[i, l+1]$ th, $[i+1, l-1]$ th, $[i+1, l]$ th, and $[i+1, l+1]$ th sampling measured intensity values in starboard echo data, as well as the absolute intensity deviation $E_{il} = \text{abs}(I_{il} - Z_{il})$.
- (2) Deciding absolute intensity deviation E_{il} : if $E_{il} > 9.78 + 1.44 \times I_{il} - 8.86 \times 10^{-4} \times I_{ij}^2 + 2.95 \times 10^{-7} \times I_{ij}^3 - 3.26 \times 10^{-11} \times I_{ij}^4$, then Z_{il} is used as the new intensity value for the $[i, l]$ th echo sampling measure in SSS starboard data; otherwise, no anomaly removal is performed.

3.1.3. Step 3: Computing Coordinates of First Bottom Returns.

According to the principle of bilateral symmetry of the sea bottom line, the first bottom returns received by the transducer arrays on the port and starboard sides of the SSS come from directly below AUV, the distances of the first bottom return echoes are equal, and the detected sea bottom lines on both sides are symmetrical to the AUV track [7, 9–11]. The above principle allows for better sea bottom line detection.

Herein, calculation of the coordinates of first bottom returns involves the following flow:

- (1) The two spatial-temporal echo sampling measure sequences in each ping of SSS port and starboard data are matched and aligned.

The issue of unequal maximum numbers, K_{leftmax} and K_{rightmax} , of the port and starboard echo sampling measures in one ping results in different temporal and spatial information implied by I_{ij} and I_{il} when $l=j$. To maximize the usage of the principle of bilateral symmetry of the sea bottom line and to achieve a combined comparison between the port and starboard measures, the above problem must first be solved for removing as many influences of the suspended matter as possible.

In this study, bottom detection focuses on the water column region data and the mutated part of the seabed region data from the water column region. Therefore, to maintain the robustness of the bottom detection method, the SSS port and starboard echo data in each ping can be regarded as two spatial-temporal sequences for matching and alignment.

When performing matching the alignment, consistency in the spatial-temporal information of mutations from the water column region to the seabed region should be maintained. Thus, the dynamic time warping (DTW) algorithm is used to achieve this goal [19], which includes the following processes:

- (a) For the i th ping, there are two spatial-temporal sequences, the port echo sampling measures $\{I_{ij}, K_m \leq j \leq K_{\text{leftmax}}\}$, and the starboard echo sampling measures $\{I_{il}, K_m \leq l \leq K_{\text{rightmax}}\}$. Take one echo from each of the two sequences and calculate the Euclidean distance $D(I_{ij}, I_{il})$ between them, where $K_m \leq j \leq K_{\text{leftmax}}$, $K_m \leq l \leq K_{\text{rightmax}}$. By calculating the Euclidean distances for all echoes in the two sequences, the Euclidean distance table can be constructed, as shown in Table 1.
- (b) Search for the shortest path in the above Euclidean distance table. In the above table, dynamic planning is used to search for the shortest path from node $D(I_{iK_m}, I_{iK_m})$ to node $D(I_{iK_{\text{leftmax}}}, I_{iK_{\text{rightmax}}})$. The shortest path must satisfy the condition that if the current node is $D(I_{ij}, I_{il})$, then the next node must be selected among $D(I_{ij+1}, I_{il})$, $D(I_{ij}, I_{il+1})$, and $D(I_{ij+1}, I_{il+1})$, and the chosen path must be the shortest.
- (c) According to the nodes on the shortest path through the Euclidean distance table, two discrete sequences can be obtained as aligned echo sampling measure pairs with the same sequence length recorded as K_{max} .
- (2) Perform logarithm transformation on the matched and aligned port and starboard echo sampling measures in each ping; that is, take the base 10 logarithm.
- (3) Calculate the standard deviation of the logarithmic sequence of the port and starboard echo sampling measures in each ping, and use them to construct a composite sequence of this ping.

Set a window to W , with a value as small as possible, to avoid flooding the mutant features of the first bottom return. Empirically, W can be calculated as follows:

$$W = \frac{\text{minimum height of SSS from seabed} \times \min(K_{\text{leftmax}}, K_{\text{rightmax}})}{(3 \sim 6) \times \text{slant range}} \quad (3)$$

By calculating the standard deviation of the logarithmic sequence of the port and starboard echo sampling measures in each ping, two standard deviation sequences, denoted as $\{L_{iq}, 1 \leq q \leq K_{\text{max}} - W + 1\}$ and $\{L_{is}, 1 \leq s \leq K_{\text{max}} - W + 1\}$,

are obtained, where the sequence length is $K_{\text{max}} - W + 1$.

In SSS echo data, there may be echo anomalies in the water column region, which will cause large fluctuations during the calculation of its standard

TABLE 1: Euclidean distance table.

$D(I_{iK_m}, I_{iK_m})$	$D(I_{iK_m}, I_{iK_{m+1}})$...	$D(I_{iK_m}, I_{iK_{\text{rightmax}}})$
$D(I_{iK_{m+1}}, I_{iK_m})$	$D(I_{iK_{m+1}}, I_{iK_{m+1}})$...	$D(I_{iK_{m+1}}, I_{iK_{\text{rightmax}}})$
...
$D(I_{iK_{\text{leftmax}}}, I_{iK_m})$	$D(I_{iK_{\text{leftmax}}}, I_{iK_{m+1}})$...	$D(I_{iK_{\text{leftmax}}}, I_{iK_{\text{rightmax}}})$

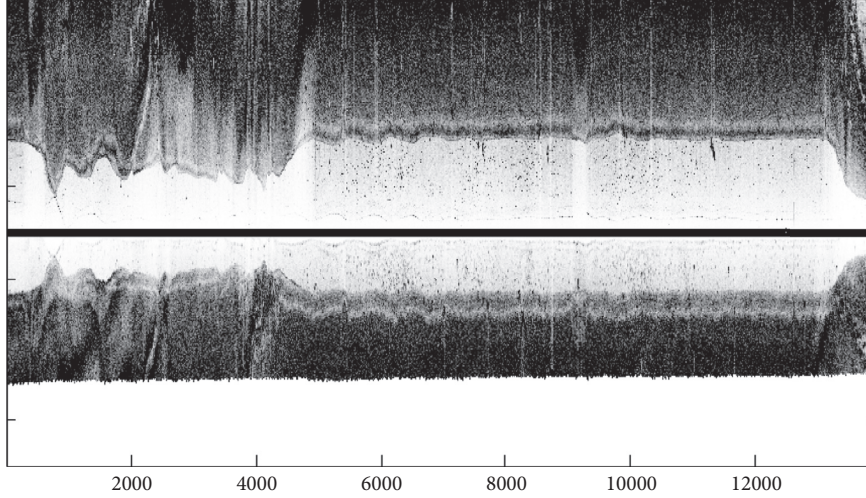


FIGURE 3: Echo data of a single ping scan line.

deviation sequence and, in turn, will affect the accuracy of subsequent extreme value detection. Therefore, to maximize the usage of the principle of bilateral symmetry of the sea bottom lines, we further take the minimum value of the port and starboard logarithmic sequence standard deviation of each ping. Consequently, for any $s = q$, $1 \leq q \leq K_{\text{max}} - W + 1$, and $1 \leq s \leq K_{\text{max}} - W + 1$, take $L_{ig} = \min(L_{iq}, L_{is})$, and a composite sequence of i th ping $\{L_{ig}, 1 \leq g \leq K_{\text{max}} - W + 1\}$ is constructed.

- (4) Perform extreme value detection on the composite sequence of each ping to obtain the extreme point coordinates.

Based on the characteristics of the SSS data, the sea bottom line comprises the first strong sea bottom echoes on the port and starboard side. Thus, the mutation of the echo intensity from the water column region to the seabed region will result in the first maximum value in the composite sequence $\{L_{ig}, 1 \leq g \leq K_{\text{max}} - W + 1\}$. Therefore, the extreme point coordinates, denoted as g_b , can be obtained by performing extreme value detection.

- (5) The coordinates of the port and starboard first bottom returns are obtained using the above extreme point coordinates.

The coordinates q_b and s_b of the standard deviation sequences corresponding to the logarithm of the obtained port and starboard echo sampling measures can be obtained using the extreme point coordinate g_b . In turn, the echo intensity $I_{b\text{left}}$ corresponding to the coordinate $q_b + W - 1$

and $I_{b\text{right}}$ corresponding to the coordinate $s_b + W - 1$ of the matched and aligned port and starboard sampling measure sequences are obtained by searching algorithms:

$\{I_{ij}, K_m \leq j \leq K_{\text{leftmax}}\}$, and the starboard echo sampling measures $\{I_{il}, K_m \leq l \leq K_{\text{rightmax}}\}$.

For each ping, by searching the port sequence $\{I_{ij}, K_m \leq j \leq K_{\text{leftmax}}\}$ and finding the first echo intensity value that equals $I_{b\text{left}}$, the coordinate of the first bottom return is derived as j_b . Similarly, by searching the port sequence $\{I_{il}, K_m \leq l \leq K_{\text{rightmax}}\}$ and finding the first echo intensity value that equals $I_{b\text{right}}$, the coordinate of the first bottom return is derived as l_b .

3.1.4. Step 4: Extraction of Sea Bottom Line. Considering the asymptotic nature of the sea bottom variation, precise sea bottom lines can be detected by sliding average filtering of the coordinates of the port first bottom returns $\{j_b\}$ and the coordinates of the starboard first bottom returns $\{l_b\}$ of all pings obtained through the above calculation [2].

3.2. Experimental Results and Discussion. In this experiment, the Sea Scan® ARC SCOUT Mk-II from the Marine Sonic Technology mounted on an AUV is used. The main operating parameters of the SSS are set to a ping frequency of 900 kHz and a slant range of 10 meters. Raw data of all pings are collected through actual measurements in the offshore test.

The waterfall plot of the SSS echo data in reverse of the image displayed at 256 grayscale levels is shown in Figure 3. The corresponding separate SSS images for the port and

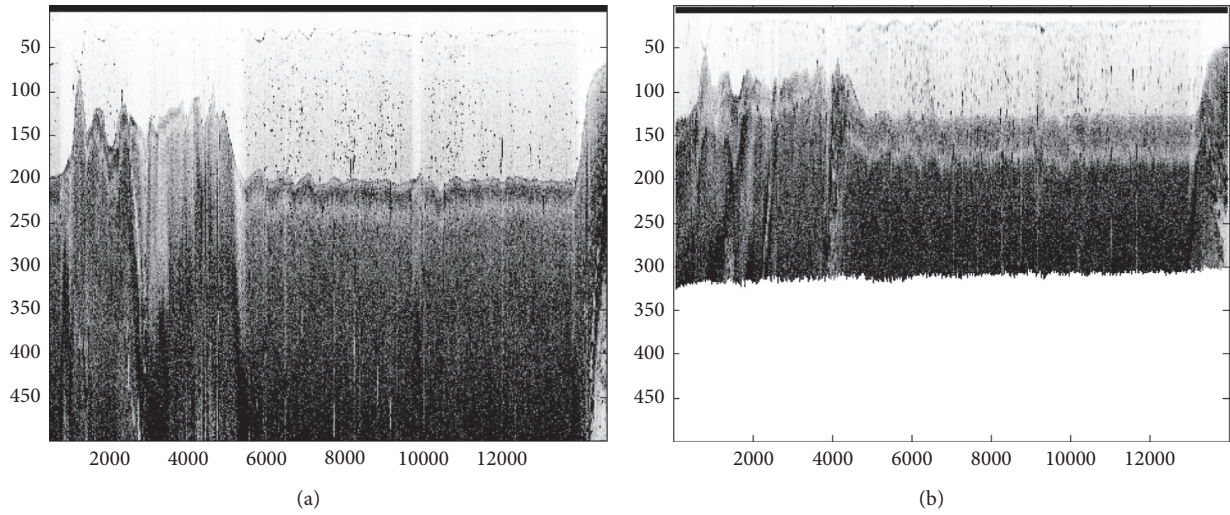


FIGURE 4: Separate SSS images for port and starboard sides with blind zone: (a) port side and (b) starboard side.

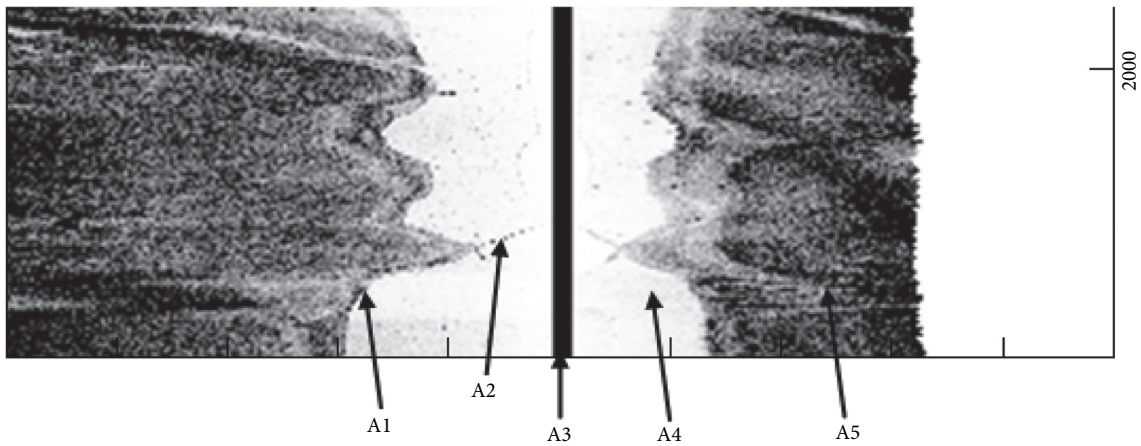


FIGURE 5: Main distribution parameters of the SSS waterfall image.

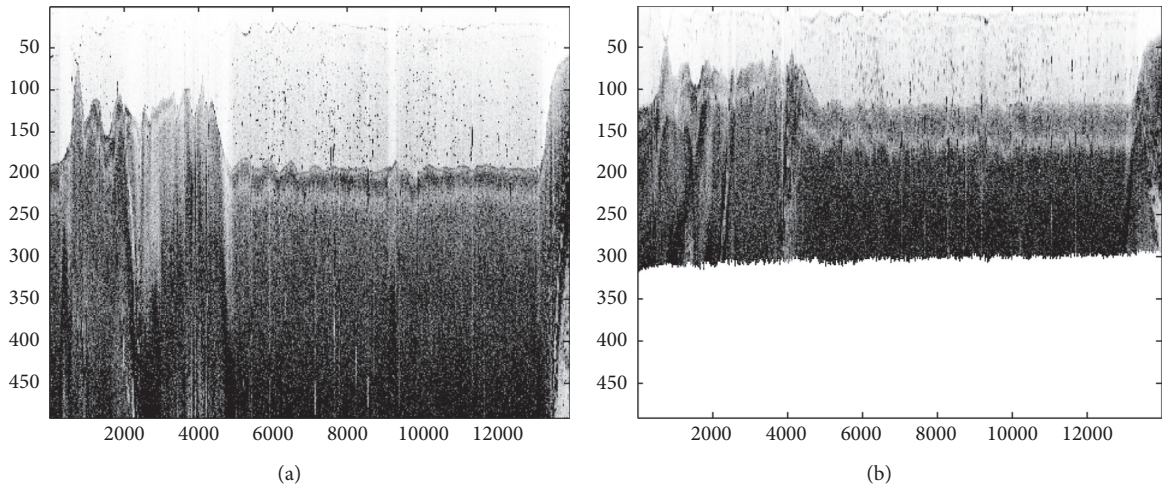


FIGURE 6: Separate SSS images of the port and starboard sides without blind zone: (a) port side and (b) starboard side.

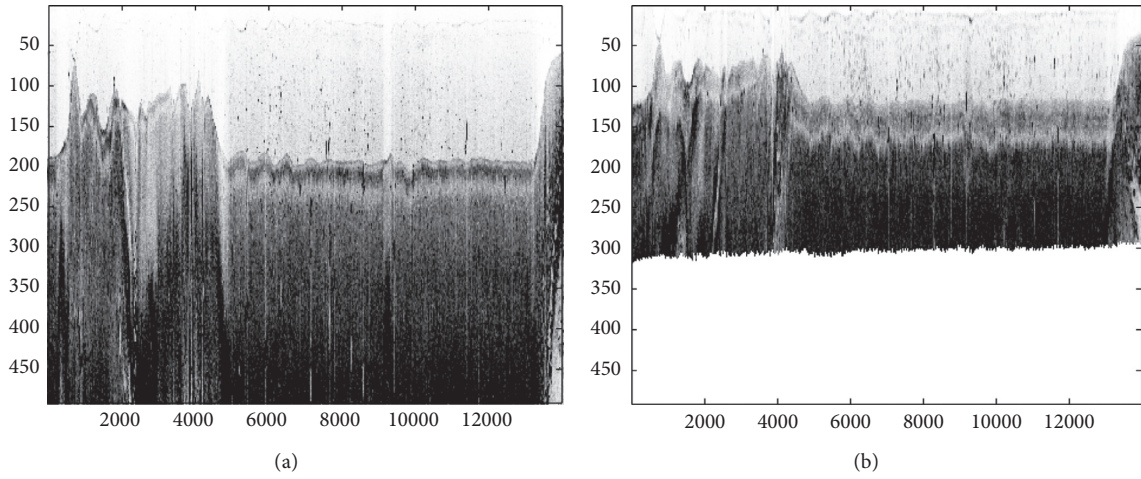


FIGURE 7: Separate SSS images of the port and starboard sides without blind zone after high-frequency noise and outliers filtering: (a) port side and (b) starboard side.

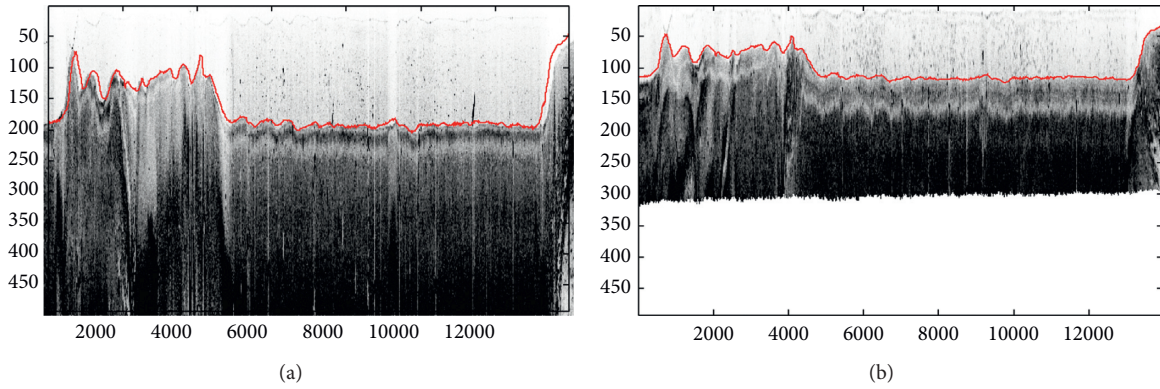


FIGURE 8: Detected sea bottom line marked separately on the port and starboard sides without showing the blind zone: (a) port side and (b) starboard side.

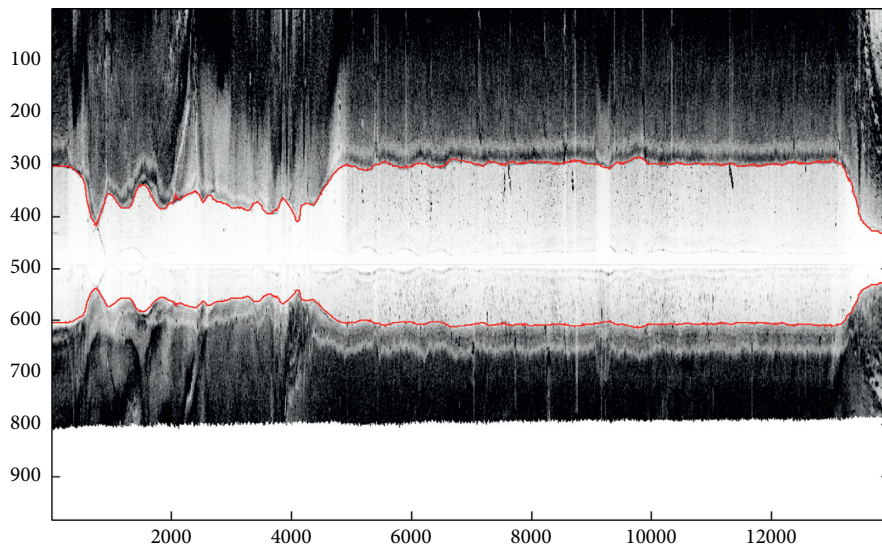


FIGURE 9: Sea bottom line marked in the waterfall of the SSS image without showing the blind zone.

starboard sides with blind zone are shown in Figures 4(a) and 4(b).

A waterfall plot of considerable echo data intercepted after the ping starting in Figure 3 is shown in Figure 5, where A1, A2, A3, A4, and A5 indicate the sea bottom line, sea surface line, blind zone, water column region, and seabed image region, respectively. It can be seen obviously that the darker the color, the stronger the echo intensity, and, conversely, the whiter (brighter) the color, the weaker the echo intensity.

Echo sampling measures of a single ping are shown in Figure 2. After removing the blind zone according to Step 1 in Section 3.1, separate SSS images of the port and starboard sides without showing the blind zone can be obtained, as shown in Figure 6.

In Figure 6, it can be seen that there are many anomalies in the echo data which may interfere with the detection of the sea bottom line and even cause false detection in complex environments, such as suspended matter present in the water column region. These echo anomalies are essentially high-frequency noise and outliers that can be effectively removed from the image during data processing according to an improved median filtering algorithm in Step 2 in Section 3.1, as shown in Figure 7.

The separate SSS images of the port and starboard sides without the blind zone after high-frequency noise and outliers filtering are shown in Figure 7. Compared to Figure 6, the echo anomalies in the water column region can be eliminated to some extent.

Processed with Step 3 and Step 4 in Section 3.1, the detected sea bottom line marked separately on the port and starboard sides without showing the blind zone is shown in Figure 8, while the sea bottom line marked in the waterfall of the SSS image without showing the blind zone is shown in Figure 9.

4. Conclusions

In this study, the automatic bottom detection method of SSS image for AUV missions is presented. The existence of blind zone in the SSS image is taken into account, and the blind zone boundary of the SSS image is calculated based on the SSS parameters conveniently. The anomalies in SSS image are eliminated effectively using an improved median filtering algorithm. Further, to perform extreme value detection according to the principle of bilateral symmetry of the sea bottom lines, the SSS port and starboard echo data sequences in each ping are retained by using the logarithm of the space-time align-matched data. Therefore, an accurate sea bottom line is obtained through a sliding average filter with the coordinate sequences of the first bottom returns. The experimental results show that, without manual intervention to set the initial values online, the proposed method can overcome the strong noise interferences and achieve high detection accuracy and good detailed continuity in the process of sea bottom line detection.

Data Availability

The underlying data supporting the results of our study can be found at https://www.researchgate.net/profile/Huapeng_Yu.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This research was supported by the project of the National Natural Science Foundation of China (Grant no. 61803381).




References

- [1] P. Blondel, *The Handbook of Sidescan Sonar*, Springer-Verlag, New York, NY, USA, 2009.
- [2] M. Al-Rawi, F. Elmgren, M. Frasher et al., "Algorithms for the detection of first bottom returns and objects in the water column in side-scan sonar images," in *Proceedings of the A Vision for Our Marine Future Oceans '17*, Aberdeen, UK, June 2017.
- [3] C. Capus, A. C. Banks, E. Coiras, I. T. Ruiz, C. Smith, and Y. Petillot, "Data correction for visualisation and classification of sidescan sonar imagery," *IET Radar Sonar and Navigation*, vol. 2, no. 3, pp. 155–169, 2007.
- [4] G. Guo, X. Wang, and H. Xu, "Review on underwater target detection, recognition and tracking based on sonar image," *Control and Decision*, vol. 2, no. 3, pp. 155–169, 2007.
- [5] P. S. Chavez, J. Isbrecht, P. Galanis et al., "Processing, mosaicking and management of the Monterey Bay digital sidescan-sonar images," *Marine Geology*, vol. 181, no. 1–3, pp. 305–315, 2002.
- [6] J. Zhang, G. Pan, and W. Ding, "Bottom track method in side-scan sonar data processing based on LOG operator," *Marine Science Bulletin*, vol. 29, no. 3, pp. 324–328, 2010.
- [7] J. Zhao, X. Wang, and H. Zhang, "An automatic bottom extracting method for side-scan sonar image," *Geomatics and Information Science of Wuhan University*, vol. 42, no. 12, pp. 1797–1803, 2017.
- [8] J. Zhao, X. Wang, H. Zhang, and A. Wang, "A comprehensive bottom-tracking method for sidescan sonar image influenced by complicated measuring environment," *IEEE Journal of Oceanic Engineering*, vol. 42, no. 3, pp. 619–631, 2017.
- [9] A. Ku, X. Zhou, F. Wang, and Q. Tang, "A bottom extracting method for side-scan sonar image with single beam sounding," *Marine Science Bulletin*, vol. 38, no. 1, pp. 47–52, 2019.
- [10] A. Wang, I. Church, J. Gou, and J. Zhao, "Sea bottom line tracking in side-scan sonar image through the combination of points density clustering and chains seeking," *Journal of Marine Science and Technology*, vol. 25, no. 3, pp. 1–17, 2019.
- [11] J. Yan, J. Meng, and J. Zhao, "Real-time bottom tracking using side scan sonar data through one-dimensional convolutional neural networks," *Remote Sensing*, vol. 12, no. 1, pp. 1–21, 2019.
- [12] P. Woock, "Side-scan sonar based SLAM for the deep sea," in *Proceedings of the Joint Workshop of Fraunhofer IOSB and Institute for Anthropomatics, Vision and Fusion Laboratory 2010*, pp. 63–74, La Bresse, France, July 2010.
- [13] T. Glotzbach, M. Schneider, and P. Otto, "Cooperative line of sight target tracking for heterogeneous unmanned marine vehicle teams: from theory to practice," *Robotics and Autonomous Systems*, vol. 67, pp. 53–60, 2015.
- [14] J. Ferrand, J.-P. Malkasse, N. L. Bouffant, and F. Florin, "Automatic mine counter measure mission control for AUV systems," *IFAC Proceedings Volumes*, vol. 40, no. 17, pp. 7–12, 2007.

- [15] J. Petrich, M. F. Brown, J. L. Pentzer, and J. P. Sustersic, "Side scan sonar based self-localization for small autonomous underwater vehicles," *Ocean Engineering*, vol. 161, pp. 221–226, 2018.
- [16] D. Moreno, A. Burguera, and G. Oliver, "SSS-SLAM: an object oriented matlab framework for underwater SLAM using side scan sonar," in *Proceedings of the Days Automatic*, pp. 1–8, Valencia, Spain, September 2014.
- [17] A. Wang, H. Zhang, X. Wang, and X. Shang, "Processing principles of side-scan sonar data for seamless mosaic image," *Journal of Geomatics*, vol. 42, no. 1, pp. 26–29, 2017.
- [18] T. P. Bas, D. C. Mason, and N. C. Millard, "TOBI image processing-the state of the art," *IEEE Journal of Oceanic Engineering*, vol. 20, no. 1, pp. 85–93, 1995.
- [19] D. J. Berndt and J. Clifford, "Using dynamic time warping to find patterns in time series," *Knowledge Discovery and Data Mining*, pp. 359–370, 1994.

Research Article

Analysis on the Impact of Data Augmentation on Target Recognition for UAV-Based Transmission Line Inspection

Chunhe Song ^{1,2,3,4} Wenxiang Xu,^{1,2,3,4,5} Zhongfeng Wang,^{1,2,3,4} Shimao Yu,^{1,2,3,4} Peng Zeng ^{1,2,3,4} and Zhaojie Ju ^{4,6}

¹State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China

²Key Laboratory of Networked Control Systems, Chinese Academy of Sciences, Shenyang 110016, China

³Institutes for Robotics and Intelligent Manufacturing, Chinese Academy of Sciences, Shenyang 110016, China

⁴Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China

⁵University of Chinese Academy of Sciences, Beijing 100049, China

⁶School of Computing, University of Portsmouth, Portsmouth, UK

Correspondence should be addressed to Peng Zeng; zp@sia.cn and Zhaojie Ju; juzhaojie@sia.cn

Received 21 July 2020; Revised 17 August 2020; Accepted 24 August 2020; Published 30 September 2020

Academic Editor: Guang Li

Copyright © 2020 Chunhe Song et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Target recognition is one of the core tasks of transmission line inspection based on Unmanned Aerial Vehicle (UAV), and at present plenty of deep learning-based methods have been developed for it. To enhance the generalization ability of the recognition models, a huge number of training samples are needed to cover most of all possible situations. However, due to the complexity of the environmental conditions and targets, and the limitations of images' collection and annotation, the samples usually are insufficient when training a deep learning model for target recognition, which is one of the main factors reducing the performance of the model. To overcome this issue, some data augmentation methods have been developed to generate additional samples for model training. Although these methods have been widely used, currently there is no quantitative study on the impact of the data augmentation methods on target recognition. In this paper, taking insulator strings as the target, the impact of a series of widely used data augmentation methods on the accuracy of target recognition is studied, including histogram equalization, Gaussian blur, random translation, scaling, cutout, and rotation. Extensive tests are carried out to verify the impact of the augmented samples in the training set, the test set, or the both. Experimental results show that data augmentation plays an important role in improving the accuracy of recognition models, in which the impacts of the data augmentation methods such as Gaussian blur, scaling, and rotation are significant.

1. Introduction

Transmission line inspection plays a very important role in ensuring the safety of the power system. In recent years, with the development of Unmanned Aerial Vehicle (UAV) technology, UAVs have been successfully applied in power transmission line inspection. In the traditional transmission line inspection method based on UAV, under the control of the operators, the UAVs fly along the specific inspection route and use the cameras to collect images or videos at certain locations. These images or videos will be transmitted back to the operators to obtain the actual flight paths, the

environment around the UAVs, and the targets to be identified. In this method, target recognition mainly depends on human experience, which limits the efficiency and accuracy of it [1].

With the progress of artificial intelligence technology, increasing attention has been paid to UAV autonomous inspection technology. As shown in Figure 1, by using GPS and intelligent image recognition technology, UAV can realize autonomous planning of inspection path, automatic obstacle avoidance, and intelligent recognition of suspicious targets. It is clear that the accuracy of image recognition directly determines the success of UAV autonomous



FIGURE 1: UAV-based autonomous transmission line inspection.

inspection. In recent years, the development of the deep learning technology has greatly improved the performance of image target recognition [2–4]. The performance of deep learning models highly rely on the number of valid training samples; unfortunately, lack of valid training samples is a common problem in deep learning model training. Due to the complexity of the environmental conditions and targets, as well as the limitations of images collection and annotation, the valid samples usually cannot cover most of all possible situations. As a result, the generalization ability of the recognition models is reduced, making them hardly usable in the practical scenes. To overcome this issue, some data augmentation methods have been developed to generate additional samples for deep learning models training [5–10]. Although these methods have been widely used, currently there is no quantitative study on the impacts of them on target recognition, which greatly limits the usage of these data augmentation methods.

In this paper, taking insulator strings as the target, the impact of a series of widely used data augmentation methods on the accuracy of target recognition is studied. The reason of choosing insulator strings as the target is that, in the transmission line, insulator strings are widely installed and have a variety of types, which are also fault prone elements. Once an insulator string fails, the transmission line cannot work normally and a large area of power failure is caused, which poses a great threat to the safe and stable operation of the power system. The recognition of insulator strings is a very important step to detect the fault of them, and the recognition accuracy of the insulator strings directly determines the result of insulator string fault detection. However, due to the complex background, the various shapes of insulator strings, uncertain shooting parameters of the cameras, and the recognition of insulator string are still a very challenging work. Although the recent advanced deep learning-based methods provide a promising approach to solve this issue, they suffer the insufficient training sample issue, similar with many other practical scenes. Therefore, data augmentation methods are usually adopted when training a deep learning-based model for insulator string recognition. Currently, the widely used data augmentation methods include histogram equalization [1, 6, 8], Gaussian blur [1, 7], translation [1, 5, 10], scaling [9, 10], and rotation [1, 9, 10]. Although these data augmentation methods have been widely used, there is still a lack of quantitative analysis of the impact of different

augmentation methods on recognition results, which seriously hinders the further improvement of data augmentation impact.

To overcome this issue, this paper studies the impact of some widely used data augmentation methods on the accuracy of target recognition, including histogram equalization, Gaussian blur, random translation, scaling, cutout, and rotation. Extensive experiments are carried out, and it is found that data augmentation plays an important role in improving the recognition performance of the model when the dataset is small. Meanwhile, Gaussian blur, scaling, and rotation have a great impact on the target recognition performance.

The rest of this paper is arranged as follows. Section 2 reviews the related works, and Section 3 gives the details of different data augmentation methods and examples. Section 4 presents extensive experiments and analyzes the impact of data augmentation methods on target recognition, and Section 5 is the conclusion.

2. Related Works

The safe and stable operation of the power system is of great significance to human life. Intelligent analysis of the power system has always been a research hotspot [11, 12]. As a key element of the power transmission line, insulator string recognition and defect detection is a research hotspot. As an important part of the transmission line, the state of insulator string directly determines the operation safety of the transmission line. Therefore, insulator string recognition and fault detection has always been a research hotspot. Insulator string recognition can be carried out using traditional image processing methods or the recent advanced deep learning-based methods. Because the insulator string in the image does not always follow certain directions, the recognition methods need to detect the insulator string along all possible direction, which is very time consuming. To overcome this issue, Zhao et al. [13] propose an insulator string recognition method based on orientation angle detection and binary shape prior knowledge. Zhao et al. [14] propose an insulator strings recognition method in infrared image based on binary robust invariant scalable keypoints (BRISK) and vector of locally aggregated descriptors (VLAD).

With the development of deep learning technology, many neural networks or convolutional neural networks (CNNs) methods have been proposed [15–20]. Zhao et al. [15] adopt the VGG16 structure and replace the last three fully connected layers with a VLAD pooling layer, and a SVM is trained for binary image classification. Sadykova et al. [9] adopt the You Only Look Once (YOLO) model for insulator string recognition. Chen et al. [18] use the You YOLO V3 algorithm for insulator strings recognition. Meanwhile, they improve the image quality using a super-resolution method based on Super-Resolution Convolutional Neural Network (SRCNN). In [20], Kang et al. use a faster R-CNN network for insulator string recognition, and a deep multitask neural network for insulator string defect detection. Miao et al. [5] use a single shot multibox detector

(SSD) for insulator string recognition. Jiang et al. [6] use SSD for insulator string defect detection from the entire image, the multi-insulator image, and the single-insulator image and then adopt ensemble learning to combine the results. In [7], Sampedro et al. first propose Up-Net, a fully convolutional network (FCN) architecture, for insulator string segmentation, and then design a Siamese convolutional neural network- (SCNN-) based method for insulator string defect detection. Ling et al. [8] and Li et al. [10] use a faster R-CNN network for insulator string recognition and a U-net for insulator string defect detection. Tao et al. [1] propose a CNN cascading architecture for insulator string recognition and defect detection.

Similar with many other practical recognition tasks based on deep learning technology, the performance of deep learning models of insulator string recognition highly rely on the number of valid training samples. However, due to the complexity of the environmental conditions and targets, as well as the limitations of image collection and annotation, the valid samples usually cannot cover most of all possible situations. To overcome this issue, some data augmentation methods have been developed to generate additional samples for deep learning model training. For example, Miao et al. [5] adopt horizontal and vertical flip; Jiang et al. [6] adopt horizontal flip and gamma correction; Tao et al. [1] adopt affine transformation, insulator and new background fusion, Gaussian blur, and brightness transformation; Ling et al. [8] adopt augmentation methods including random flip, crop, random saturation, brightness, and contract perturbation. Sadykova et al. [9] adopt many augmentation methods such as Gaussian noise, Gaussian blur, average blur, Median blur, rotation, scaling, addition, and multiplication. Li et al. [10] adopt more augmentation methods such as mirroring, rotation, affine transformation, Gaussian white noise, brightness and color transformation, and other data augment operations. Although these image augmentation methods have been widely used, at present, there is lack of quantitative study on the impacts of them on target recognition.

3. Data Augmentation

3.1. Histogram Equalization. Histogram equalization is to stretch the image nonlinearly and redistribute the pixel value of the image so that the gray histogram of the original image changes from a certain gray range to a uniform distribution in the whole range. For a RGB image in this paper, first, the image is converted from RGB space to HSV space, and then histogram equalization is performed on the V channel in HSV space as

$$p_r(r_k) = \frac{n_k}{n}, \quad k = 0, 1, \dots, L-1, \quad (1)$$

where n is the total number of pixels in the image, n_k is the number of pixels with gray level r_k , and L is the total number of possible gray levels in the image.

Meanwhile, the pixels of the gray level r_k in the image can be mapped to the corresponding pixels of the gray level s_k in the output image by

$$s_k = \sum_{j=0}^k p_r(r_k) = \sum_{j=0}^k \frac{n_j}{n}, \quad k = 0, 1, \dots, L-1. \quad (2)$$

Figure 2 gives two examples of histogram equalization, in which Figures 2(a) and 2(c) are original images, and Figures 2(b) and 2(d) are corresponding images after histogram equalization. From Figure 2, it can be seen that, compared with the original images Figures 2(a) and 2(c), the images after histogram equalization Figures 2(b) and 2(d) have a better visual impact. However, whether this impact can benefit the performance of target recognition is not clear, which will be verified in Section 4.

3.2. Gaussian Blur. Gaussian blur is to transform the weight of each pixel in an image according to the Gaussian distribution function using the weighted average value with the surrounding pixels:

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-(x^2+y^2/2\sigma^2)}, \quad (3)$$

where σ is the standard deviation, and x and y are the coordinates of pixels in the Gaussian blur kernel. Then a Gaussian blurred image I_b can be obtained by

$$I_b = I \oplus G(x, y), \quad (4)$$

where \oplus is the convolution operator.

The impact of Gaussian blur depends on the standard deviation σ and the size of the Gaussian blur kernel. In this paper, the standard deviation of Gaussian blur is fixed as 5, and the sizes of the Gaussian blur kernels are set to $3 * 3$, $7 * 7$, and $11 * 11$, respectively. Figure 3 gives an example of an image blurred by different Gaussian blur kernels, where Figure 3(a) is the original image and Figures 3(b)–3(d) are blurred images resulted from Gaussian blur with the kernels $3 * 3$, $7 * 7$, and $11 * 11$, respectively.

The Gaussian blurred image is to simulate the defocusing effect when the target is not in the focus position of the camera during exposure, which is a common phenomenon. Therefore, it may be an effective data augmentation method.

3.3. Random Translation. Random translation is to keep the size of the image unchanged and move the whole image up/down/left/right for a certain distance. In this paper, the translation distance is random, but ensures that the insulator string will not be moved out of the image. Figure 4 gives two examples of random translation, in which Figures 4(a) and 4(c) are original images, and Figures 4(b) and 4(d) are corresponding images after random translation. Note that the vacant area resulted from random translation are filled by zeroes.

3.4. Image Scaling. Image scaling is to resize the image while keeping the image aspect ratio of the length and width unchanged. This strategy is used to simulate the effect of different focal lengths of the camera on the shooting results.



FIGURE 2: Images before and after histogram equalization.



FIGURE 3: Continued.



FIGURE 3: Images burred by different Gaussian blur kernels.

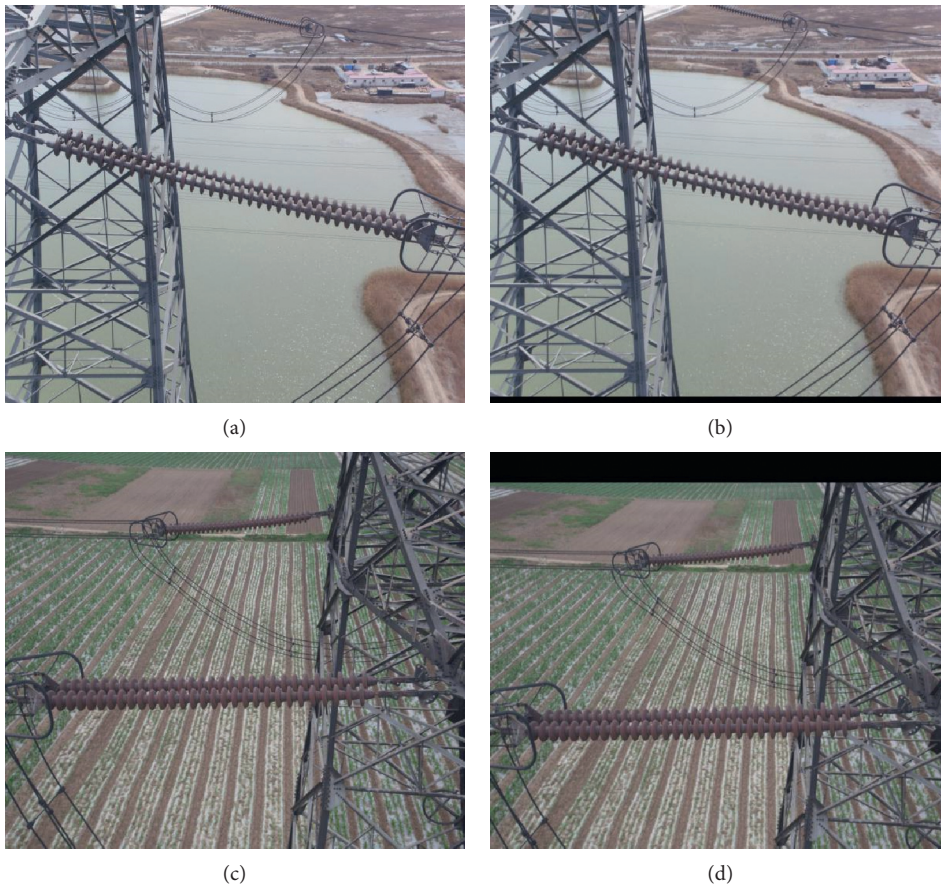


FIGURE 4: Images before and after random translation.

In this paper, two scaling ratios 0.5 and 2 are applied. Figure 5 gives two examples of image scaling, in which Figures 5(a) and 5(c) are original images and Figures 5(b) and 5(d) are corresponding images after image scaling. Similar with random translation, the vacant area resulted from image scaling are filled by zeroes. Note that, for some images, parts of insulator strings may be out of the images.

3.5. Image Cutout. Image cutout is the process of generating a new image by eliminating a certain region of the image. Cutout can simulate the situation that the target is partially occluded, which is a common phenomenon in a natural image. In this paper, image cutout is carried out by eliminating a region with the size of $100 * 100$. The region is randomly selected in any position of the image. Note that the

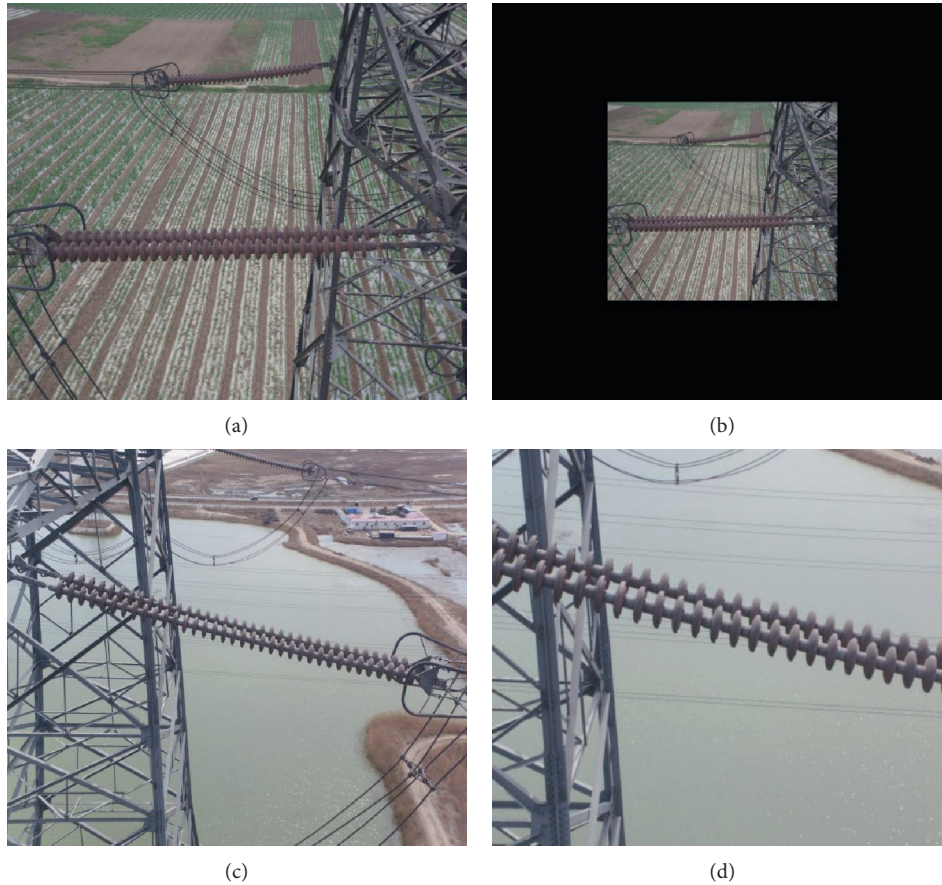


FIGURE 5: Images before and after scaling.

region may not be completely in the image, e.g., at the boundary of the image. Meanwhile, if less than 50% of the region appears in the image, then the region will be re-selected. This strategy is used to ensure that the valid area of the eliminated region will not be significantly reduced at the boundary of the image. Figure 6 gives two examples of image cutout, in which Figures 6(a) and 6(c) are original images and Figures 6(b) and 6(d) are corresponding images after image cutout.

3.6. Image Rotation. Compared with general objects in natural images, the aspect ratio of insulator string is extremely large. The algorithms based on CNN are to search the region in the current image with high similarity with the labeled region in the training samples. When the algorithm based on CNN is applied to the identification of insulator strings, the overlapping area of insulator strings at different directions is greatly reduced due to the influence of the extremely large aspect ratio. Therefore, when the direction of the insulator string in the detected image is different from that in the training sample set, the detection rate will be greatly reduced. As a result, image rotation [1, 9, 10] has been adopted to increase the coverage of insulator string directions in the training set. In this paper, the relationship between the target rotation angle and recognition rate will be analyzed quantitatively. In order to achieve this goal, first,

the image is rotated to make the insulator string in the image in a horizontal position; second, the aspect ratio of the insulator strings in the images are calculated, and the images are divided into the large class and the small class according to the aspect ratio; finally, the images belonging to the large class and the small class are used for training and testing, respectively; for the test set image, it is rotated to -87° , -84° , \dots , -3° , 0° , 3° , 6° , \dots , 87° , and 90° , respectively, to test whether it can be recognized by the trained model.

Figure 7 gives examples of image rotation, in which Figure 7(a) is the original image and Figure 7(b) is a rotated image that the insulator string in the image is in horizontal position. Figures 7(c) and 7(d) are rotated images, where the insulator strings in the images are 3° and -3° from the horizontal.

4. Experiment and Discussion

The basic dataset used in this paper contains 848 insulator string images with a resolution of $1152 * 864$, including 600 images with insulator strings and 248 images without insulator string. The dataset is divided into the training set and test set with a ratio of 4:1, and the number of images of the training set and test set is 678 and 170, respectively.

The experiments are carried out on a server with ubuntu18.04, python 3.6, and rx2080ti, and the deep learning framework is Caffe. In this paper, the software

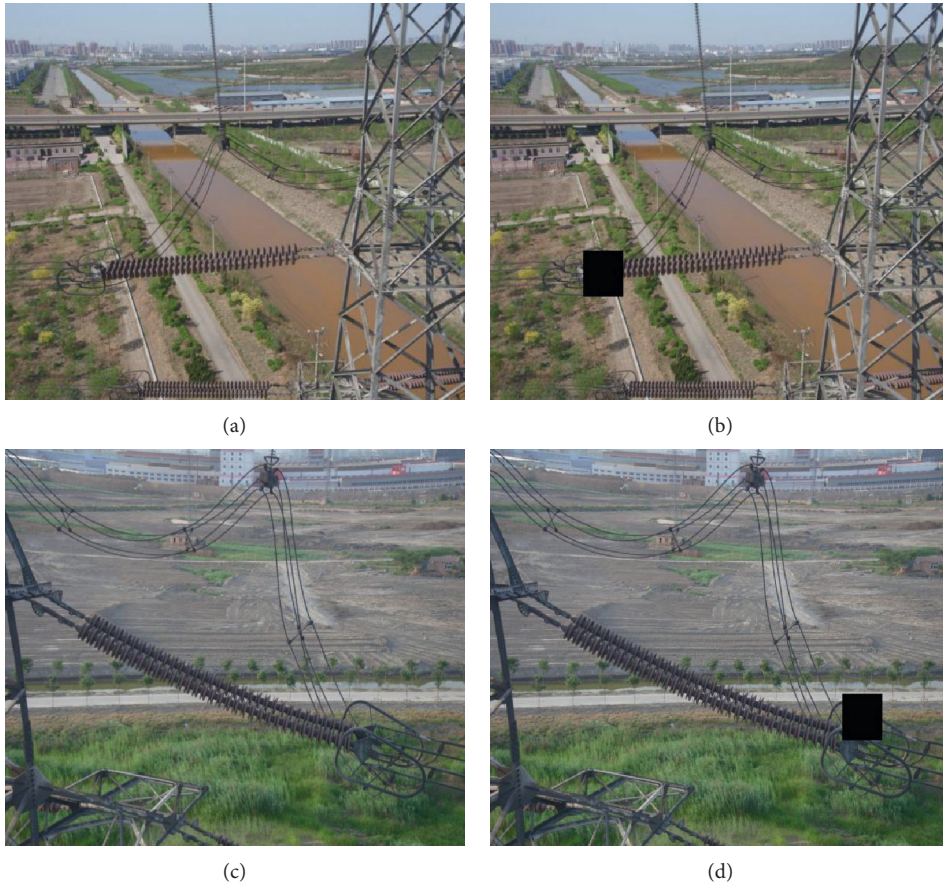


FIGURE 6: Images before and after cutout.



FIGURE 7: Continued.

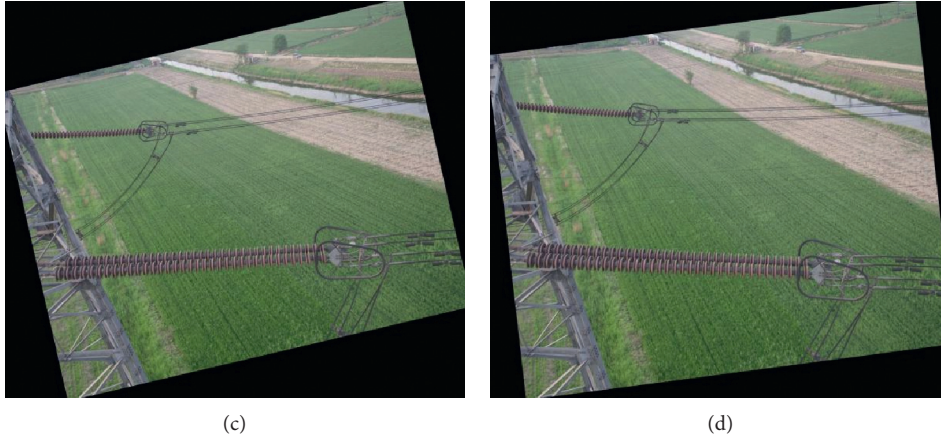


FIGURE 7: Images before and after rotation.

LabelImg is used to label the 848 images in the basic dataset. The target recognition algorithm is faster RCNN [2], and the pretrained model is ZF. For the training process, the initial learning rate is set to 0.001, the weight attenuation coefficient is set to 0.0005, and the momentum value is 0.9.

4.1. Histogram Equalization. For this test, first the training set and test set are augmented by Gaussian blur with a kernel $3 * 3$, random translation, scaling ($\times 0.5$), and cutout, respectively. After that, the training set and test set are 4 times of the basic dataset, and these data are taken as the original dataset. In order to verify the impact of histogram equalization on recognition performance, the images in the original dataset are processed by histogram equalization, and then a new training set and test set are formed for training and testing. The test results before and after histogram equalization are shown in Table 1, where Ori, Aug, and All are the test results based on the original dataset, the augmented dataset, and all dataset, respectively.

It can be seen from Table 1 that when histogram equalization is used to augment the dataset, the recognition accuracy of the test set processed by histogram equalization is improved by 0.6%. For the original test data and augmented data, the recognition accuracy of the insulator string is improved by 0.22%, which shows that histogram equalization is a useful method to improve the accuracy of insulator strings recognition, although the impact is not significant.

4.2. Gaussian Blur. For this test, first the training set and test set are augmented by histogram equalization, random translation, scaling ($\times 0.5$), and cutout, respectively. After that, the training set and test set are 4 times of the basic dataset. In order to verify the impact of different Gaussian blur kernels on the recognition performance, the dataset is blurred by Gaussian blur with different kernels, and then a new training set and test set are formed. The test results with different Gaussian blur kernels are shown in Table 2, where K1, K2, and K3 are Gaussian blur with kernels $3 * 3$, $7 * 7$, and $11 * 11$, respectively.

TABLE 1: Test results before and after histogram equalization.

	Test set		
	Ori	Aug	All
<i>Training set</i>			
Ori	0.9002	0.8932	0.8975
All	0.9003	0.8992	0.8997

TABLE 2: Test results on Gaussian blur.

	Test set				
	Ori	K1	K2	K3	All
<i>Training set</i>					
Ori	0.9014	0.8924	0.7923	0.6038	0.8013
Ori + K1	0.9021	0.9004	0.8129	0.6307	0.8108
Ori + K2	0.9030	0.8999	0.8971	0.8100	0.8889
Ori + K3	0.9010	0.9006	0.8921	0.8922	0.8978
Ori + K1 + K2	0.9021	0.9012	0.8963	0.8136	0.8942
Ori + K1 + K3	0.9001	0.8994	0.8956	0.8798	0.8970
Ori + K2 + K3	0.9035	0.8995	0.8999	0.8821	0.8981
All	0.9033	0.9016	0.8975	0.8739	0.8985

It can be seen from Table 2 that the recognition accuracy of the model trained only with the original data on the test set images with $3 * 3$, $7 * 7$, and $11 * 11$ blur kernels decreases in turn; while for the model trained with blurred images and original images or only using the original image, when the test images are blurred with small Gaussian blur kernels (e.g., $3 * 3$), the difference of the recognition accuracy among different test datasets is not significant. It means that Gaussian blur with a small kernel has little impact on the result. However, when the training set contains images blurred by large kernels, the recognition accuracy on the test dataset is improved significantly. From the above analysis it can be seen that Gaussian blur is an effective method to improve the accuracy of insulator strings recognition, and the impact is significant.

4.3. Random Translation. For this test, first the training set and test set are augmented by histogram equalization, Gaussian blur with a kernel $3 * 3$, scaling ($\times 0.5$), and

cutout, respectively. After that, the training set and test set are 4 times of the basic dataset, and these data are taken as the original dataset. In order to verify the impact of random translation on recognition performance, the original dataset is randomly translated, and then a new training set and test set are formed for training and testing. The test results before and after random translation are shown in Table 3.

It can be seen from Table 3 that when the random translation method is used to augment the dataset, the recognition accuracy rate of the test set processed by random translation is improved by 0.36%, and the recognition accuracy rate of the insulator string is improved by 0.23% for the original test data and augmented data, which indicates that the recognition performance of the insulator string can be improved by using the random translation method.

4.4. Image Scaling. For this test, first the training set and test set are augmented by histogram equalization, Gaussian blur with a kernel $3 * 3$, random translation, and cutout, respectively. After that, the training set and test set are 4 times of the basic dataset, and these data are taken as the original dataset. To verify the impact of different scaling on recognition performance, the images in the original dataset are scaled by 0.5 and 2, respectively, and then a new training set and test set are formed for training and testing. The test results before and after scaling are shown in Table 4.

It can be seen from Table 4 that when using the model trained based on the original dataset, the recognition accuracy on the test images with the scale of 0.5 and 2 is obviously lower than on the original dataset. When the training set contains the images scaled by 0.5, the recognition accuracy of all dataset is improved. Furthermore, when the training set contains the images scaled by 2, the recognition accuracy of images scaled by 2 is improved. It shows that the recognition accuracy of images with different scaling ratios can be improved when the images with different scales are used for training.

4.5. Image Cutout. For this test, first the training set and test set are augmented by histogram equalization, Gaussian blur with a kernel $3 * 3$, random translation, and scaling ($\times 0.5$), respectively. After that, the training set and test set are 4 times of the basic dataset, and these data are taken as the original dataset. In order to verify the impact of image cutout on recognition performance, the images in the original dataset are processed by cutout, and then a new training set and test set are formed for training and testing. The test results before and after histogram equalization are shown in Table 5.

It can be seen from Table 5 that when using cutout to augment the original dataset, the recognition accuracy of the test set processed by cutout is improved by 0.56%. For the original test data and augmented data, the recognition accuracy of the insulator string is improved by 0.27%, which indicates that cutout is a useful method to improve the accuracy of insulator string recognition.

TABLE 3: Test results before and after random translation.

	Test set		
	Ori	Aug	All
<i>Training set</i>			
Ori	0.8996	0.8981	0.8981
All	0.9011	0.9017	0.9004

TABLE 4: Test results before and after scaling.

	Test set			
	Ori	$\times 0.5$	$\times 2$	All
<i>Training set</i>				
Ori	0.9026	0.7955	0.6230	0.7928
Ori + ($\times 0.5$)	0.9045	0.8121	0.6539	0.8016
Ori + ($\times 2$)	0.9024	0.7240	0.9080	0.8154
All	0.9030	0.8131	0.9087	0.9010

TABLE 5: Test results before and after cutout.

	Test set		
	Ori	Aug	All
<i>Training set</i>			
Ori	0.8997	0.8965	0.8996
All	0.9031	0.9021	0.9023

4.6. Image Rotation. For the basic dataset, first, rotate the image so that the insulator string in the image is in the horizontal position. Second, based on the statistics of the aspect ratio of insulator strings, it can be seen that the aspect ratio of insulator strings is from 1 : 1 to 14 : 1. In this paper, taking the aspect ratio 8 : 1 as the threshold, the images with the insulator string's aspect ratio greater than 8 : 1 are classified as class *L*, and those with the insulator string's aspect ratio greater than 8 : 1 are classified as class *S*. If there are both class *L* and class *S* insulator strings in an image, the image belongs to class *L* and class *S* at the same time. After that the images of class *L* and class *s* are divided into training set and test set according to the ratio of 4 : 1, and then histogram equalization, Gaussian blur, random translation, scaling, and cutout are used to expand the training set and test set of these two types of images, respectively.

During the test process, for an image, first, rotate it to make the insulator strings are at -87° , -84° , -3° , 0° , 3° , \dots , 87° and 90° to the horizontal line, respectively; then, the recognition accuracies of the model for insulator strings in different angle are calculated and shown in Figure 8, where Figure 8(a) is the result of the *L* class images and Figure 8(b) is the result of the *S* class images. From Figure 8, it can be seen that the recognition accuracy of the model for insulator strings at 0° is the highest; however, with the angle deviation of 0° , the recognition accuracy of insulator string decreases dramatically. When the deviation angle is more than 12° , the model can hardly recognize the insulator string. Meanwhile, although the rotation angle is the same, the recognition

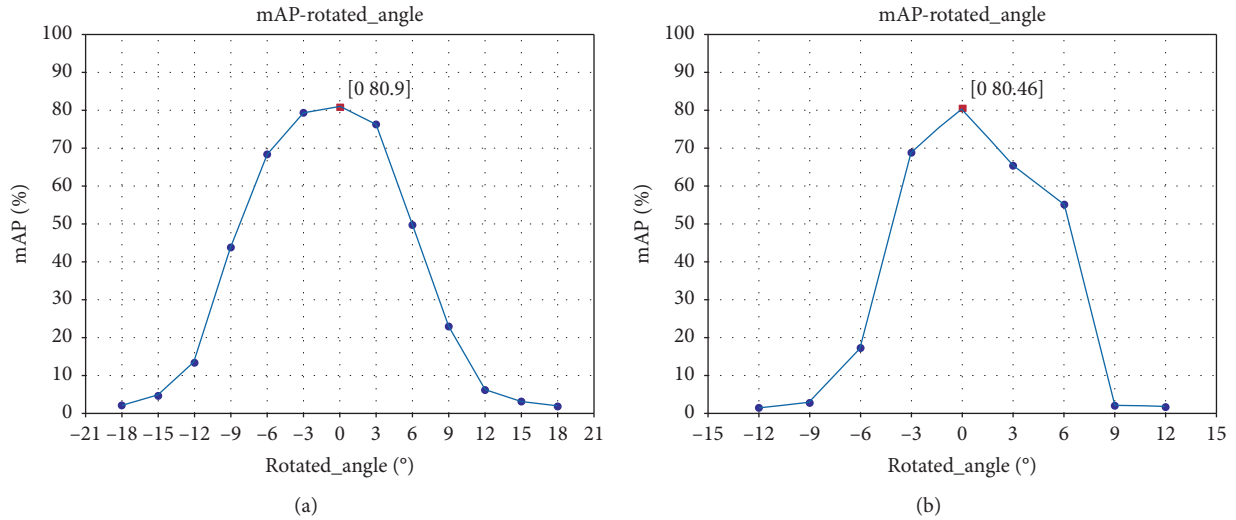


FIGURE 8: Recognition accuracy curve of two kinds of images.

accuracy is not the same due to different rotation directions. The results show that the general convolution neural network-based recognition method such as faster RCNN used in this paper cannot deal with the issue of target rotation well, and additional strategies are needed to solve the target rotation issue.

5. Conclusion

In this paper, the impact of data augmentation on target recognition for UAV-based transmission line inspection is analyzed. Based on extensive experiments on different data augmentation methods, it can be found that the data augmentation methods such as histogram equalization, random translation, and cutout can improve the target recognition accuracy, but the impact is not significant. Compared with above data augmentation methods, Gaussian blur, scaling, and rotation have a greater impact on the recognition performance of insulator strings. For rotation, the general convolution neural network-based recognition method such as faster RCNN used in this paper cannot deal with the issue of target rotation well, and additional methods should be adopted to solve this issue.

Data Availability

The data used to support the findings of the study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was partially supported by the National Key R&D Program of China under Grant 2018YFB1700200, National Nature Science Foundation of China under Grant

U1908212, State Grid Corporation Science and Technology Project (SG2NK00DWJS1800123), and State Grid Shanghai Electric Power Corporation's 2020 technology project "Research on Automatic Networking and Data Fusion Technology of Power IoT Sensors" (52090F1900BK).

References

- [1] X. Tao, D. Zhang, Z. Wang, X. Liu, H. Zhang, and D. Xu, "Detection of power line insulator defects using aerial images analyzed with convolutional neural networks," in *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 4, pp. 1486–1498, 2020.
- [2] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real time object detection with region proposal networks," in *International Conference on Neural Information Processing Systems*, vol. 50, pp. 91–99, 2015.
- [3] W. Liu, D. Anguelov, D. Erhan, and C. Szegedy, "SSD: Single shot multibox detector," *ECCV*, vol. 50, pp. 21–37, 2015.
- [4] J. Redmon, S. Divvala, R. Girshick et al., "You only look once: unified, real-time object detection," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 50, pp. 779–788, 2016.
- [5] X. Miao, X. Liu, J. Chen, S. Zhuang, J. Fan, and H. Jiang, "Insulator detection in aerial images for transmission line inspection using single shot multibox detector," in *IEEE Access*, vol. 7, pp. 9945–9956, 2019.
- [6] H. Jiang, X. Qiu, J. Chen, X. Liu, X. Miao, and S. Zhuang, "Insulator fault detection in aerial images based on ensemble learning with multi-level perception," in *IEEE Access*, vol. 7, pp. 61797–61810, 2019.
- [7] C. Sampedro, J. Rodriguez-Vazquez, A. Rodriguez-Ramos, A. Carrio, and P. Campoy, "Deep learning-based system for automatic recognition and diagnosis of electrical insulator strings," *IEEE Access*, vol. 7, pp. 101283–101308, 2019.
- [8] Z. Ling, "An accurate and real-time method of self-blast glass insulator location based on faster R-CNN and U-net with aerial images," in *CSEE Journal of Power and Energy Systems*, vol. 5, no. 4, pp. 474–482, 2019.
- [9] D. Sadykova, D. Pernebayeva, M. Bagheri, and A. James, "IN-YOLO: real-time detection of outdoor high voltage insulators

- using UAV imaging,” in *IEEE Transactions on Power Delivery*, vol. 35, no. 3, pp. 1599–1601, 2020.
- [10] X. Li, H. Su, and G. Liu, “Insulator defect recognition based on global detection and local segmentation,” in *IEEE Access*, vol. 8, pp. 59934–59946, 2020.
- [11] C. Song, W. Jing, P. Zeng, H. Yu, and C. Rosenberg, “Energy consumption analysis of residential swimming pools for peak load shaving,” in *Applied Energy*, vol. 220, pp. 176–191, 2018.
- [12] C. Song, W. Jing, P. Zeng, and C. Rosenberg, “An analysis on the energy consumption of circulating pumps of residential swimming pools for peak load management,” in *Applied Energy*, vol. 195, pp. 1–12, 2017.
- [13] Z. Zhao, N. Liu, and L. Wang, “Localization of multiple insulators by orientation angle detection and binary shape prior knowledge,” in *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 22, no. 6, pp. 3421–3428, 2015.
- [14] Z. Zhao, G. Xu, and Y. Qi, “Representation of binary feature pooling for detection of insulator strings in infrared images,” in *IEEE Transactions on Dielectrics and Electrical Insulation*, vol. 23, no. 5, pp. 2858–2866, 2016.
- [15] C. Song, P. Zeng, Z. Wang, T. Li, L. Qiao, and L. Shen, “Image forgery detection based on motion blur estimated using convolutional neural network,” *IEEE Sensors Journal*, vol. 19, no. 23, pp. 11601–11611, 2019.
- [16] C. Song, P. Zeng, Z. Wang, H. Zhao, and H. Yu, “Wearable continuous body temperature measurement using multiple artificial neural networks,” in *IEEE Transactions on Industrial Informatics*, vol. 14, no. 10, pp. 4395–4406, 2018.
- [17] Z. Zhao, X. Fan, G. Xu, L. Zhang, Y. Qi, and K. Zhang, “Aggregating deep convolutional feature maps for insulator detection in infrared images,” in *IEEE Access*, vol. 5, pp. 21831–21839, 2017.
- [18] X. Gong, Q. Yao, M. Wang, and Y. Lin, “A deep learning approach for oriented electrical equipment detection in thermal images,” in *IEEE Access*, vol. 6, pp. 41590–41597, 2018.
- [19] H. Chen, Z. He, B. Shi, and T. Zhong, “Research on recognition method of electrical components based on YOLO V3,” *IEEE Access*, vol. 7, pp. 157818–157829, 2019.
- [20] G. Kang, S. Gao, L. Yu, and D. Zhang, “Deep architecture for high-speed railway insulator surface defect detection: denoising autoencoder with multitask learning,” in *IEEE Transactions on Instrumentation and Measurement*, vol. 68, no. 8, pp. 2679–2690, 2019.

Research Article

Multimodal Multiobject Tracking by Fusing Deep Appearance Features and Motion Information

Liwei Zhang ¹, Jiahong Lai,¹ Zenghui Zhang,¹ Zhen Deng ¹, Bingwei He,¹
and Yucheng He²

¹School of Mechanical Engineering and Automation, Fuzhou University, Fuzhou, China

²The T Stone Robotics Institute, Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong, China

Correspondence should be addressed to Zhen Deng; zdeng@fzu.edu.cn

Received 6 August 2020; Revised 17 August 2020; Accepted 13 September 2020; Published 25 September 2020

Academic Editor: Guang Li

Copyright © 2020 Liwei Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Multiobject Tracking (MOT) is one of the most important abilities of autonomous driving systems. However, most of the existing MOT methods only use a single sensor, such as a camera, which has the problem of insufficient reliability. In this paper, we propose a novel Multiobject Tracking method by fusing deep appearance features and motion information of objects. In this method, the locations of objects are first determined based on a 2D object detector and a 3D object detector. We use the Nonmaximum Suppression (NMS) algorithm to combine the detection results of the two detectors to ensure the detection accuracy in complex scenes. After that, we use Convolutional Neural Network (CNN) to learn the deep appearance features of objects and employ Kalman Filter to obtain the motion information of objects. Finally, the MOT task is achieved by associating the motion information and deep appearance features. A successful match indicates that the object was tracked successfully. A set of experiments on the KITTI Tracking Benchmark shows that the proposed MOT method can effectively perform the MOT task. The Multiobject Tracking Accuracy (MOTA) is up to 76.40% and the Multiobject Tracking Precision (MOTP) is up to 83.50%.

1. Introduction

The objective of Multiobject Tracking (MOT) is to track multiple objects at the same time and estimate their current states, such as locations, velocities, and sizes, while maintaining their motion identifications. Hence, the MOT is one of the most important abilities of autonomous systems, but it remains challenging because the target objects may be obscured, or it may be interfered by objects of similar shape. Owing to the rapid development of object detectors, several tracking-by-detection methods [1–5] have been widely proposed to address the MOT problem. Typically, the existing tracking-by-detection methods involve two main computational steps: object detection and tracking. These methods first detect the location of objects and then compute the trajectories of the objects based on the results of object detection [6–8]. The accuracy of object tracking is highly related to the performance of object detection. Hence,

the important thing about the MOT is to track the new targets that appear at any time and find lost tracking target objects from detections and associate again. However, most of the tracking-by-detection methods are based on vision-based object detections. In the case of occlusion and overexposure, vision-based object detection may lead to false association with existing trajectories. For example, Figure 1(a) shows the failure of vehicle detection on the image with the occlusion of humans. Figure 1(b) shows the camera is disabled when overexposure.

The scene of autonomous driving may contain multiple objects, and the states of the objects are usually uncertain [9, 10]. In this case, the vision-based object detections are susceptible to occlusion or overexposure, which will easily lead to false checks or loss of target tracking. Besides, one major challenge of the MOT is how to reduce incorrect identity switching. Because the tracked objects often have high similarities, it is

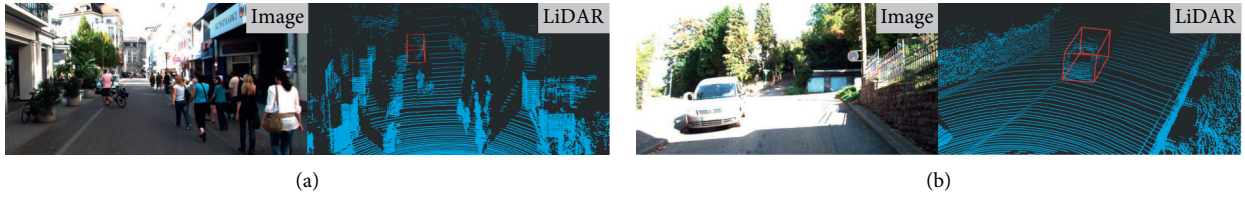


FIGURE 1: Examples of object detection in the case of occlusion and overexposure.

challenging to track objects correctly and perform correct Re-Identification(RE-ID).

Multimodal data fusion has the potential to improve the stability and accuracy of the MOT. However, a majority of traditional methods use the camera, LiDAR, or radar. These methods need to design hand-crafted features [11]. However, the hand-crafted features are often not of high precision, and it is difficult to guarantee the tracking performance. Hence, it is necessary to design a feature learning method that can automatically learn appearance features from raw visual data. Moreover, in autonomous driving systems, since the objects are moving rather than stationary, the motion information of objects should be integrated with the appearance features to achieve the MOT tasks. In addition, some MOT methods include depth information in the tracking process by using depth camera in order to improve tracking performance. For example, Mehner et al. [12] used an ordinary camera to obtain 2D information of objects and used a depth camera to obtain depth information to assist in locating the objects in world coordinates. Although it can improve the accuracy, the depth camera has a small field of view, high noise, and is easily affected by sunlight, so it is not effective as LiDAR. Moreover, they only use Kalman Filter for tracking, which does not work well in complex scenarios.

In this paper, we propose a multimodal MOT method by fusing the motion information and the deep appearance features of objects. This paper employs a 2D object detector, i.e., You Only Look Once (YOLOv3) [3] and a 3D object detector, i.e., PointRCNN [5] to process the RGB image and laser point cloud, respectively. The combination of 2D detection and 3D detection is helpful to improve the robustness of object detection. Then, the MOT is achieved by associating the motion information and the deep appearance features of the target object. A set of experiments on the KITTI Tracking Benchmark is performed to demonstrate the effectiveness of the proposed MOT method. Our contributions are summarized as follows:

- (1) The 2D object detection based on the image and the 3D object detection-based laser point cloud are combined to detect the location of objects, which is robust against light changes and occlusion.
- (2) We apply CNN that is pretrained to discriminate vehicles on a large-scale vehicle Re-Identification dataset to automatically extract the deep appearance features of the target object without manually designing features.

- (3) A multimodal MOT method is proposed by fusing the motion information and deep appearance features of the object to achieve the MOT task. In addition, the proposed method obtains competitive qualitative and quantitative tracking results on the KITTI tracking benchmark.

The rest of the paper is organized as follows. Section 2 introduces related works. Section 3 presents the proposed multimodal MOT method. Experiments and their results are presented in Section 4. Finally, the conclusion and future work are summarized in Section 5.

2. Related Works

This section provides an overview of the two related research topics: multiobject tracking and object detection.

2.1. Multiobject Tracking. The problem of the MOT first appeared in the tracking of object trajectory. For example, tracking of multiple enemy aircraft or passing missiles. With the development of computer vision, researchers have proposed several MOT methods from different aspects in the past few decades. For example, the single-object tracking method is extended to support multiple objects. According to the data association, the existing MOT methods can be divided into two categories: offline and online MOT methods. In offline methods [13–16], the detection of all frames in the sequence is combined to obtain the object trajectory robustly. These methods need to construct a global graph structure, which leads to high computational complexity. However, in the online MOT method [17–20], the target detector is only associated with the existing trajectories frame by frame. Hence, online methods are more suitable for real-time tracking.

Most of the existing MOT methods rely on motion information produced from Kalman Filter [21], Hungarian algorithm with Kalman Filter [17], Particle Filter [22], or probability hypothesis density filter [23]. However, in autonomous driving systems, due to the uncertainty of the scene, it is impossible to track objects stably only by using motion information. Therefore, more recent methods combine the motion features with the appearance features to improve the re-identification of target objects. Traditionally, the appearance features of objects are manually designed [24], which cannot provide reliable features, especially, in complex scenes. Owing to the rapid development of deep learning, deep convolutional networks [9, 25, 26] have been widely used to extract the appearance features from raw

visual data. For example, Wojke et al. [17] used CNN to extract the pedestrian image features and measure the distance between features for human detection.

2.2. Object Detection. Most of the existing 2D object detection methods are based on CNNs, which can be divided into two-stage detectors and one-stage detectors. In the two-stage detectors, such as RCNN [27], Fast RCNN [28], Faster RCNN [1], and FPN [29], they use Region Proposal Networks (RPN) to generate the candidate regions and then perform bounding-box classification and regression. For example, RCNN starts with the extraction of a set of object proposals by the selective search. Then, each proposal is rescaled to a fixed size image and fed into a CNN model that is trained on ImageNet. In this way, the presence of an object within each region is predicted and its category is recognized. Although the two-stage detectors have made great progress, their main drawback is that the redundant feature calculation of a large number of overlapping schemes results in a very slow detection speed.

The One-stage detectors have YOLO [3, 30, 31], Single Shot MultiBox Detector (SSD) [2], and RetinaNet [32]. These detectors do not need the RPN. They directly generate the categories' probability and bounding boxes of the objects. These methods only use one-stage calculation to get the final detection results. For example, the YOLO applies a single neural network to the whole image. This network divides the images into regions and predicts the bounding boxes and the probabilities for each region simultaneously. Compared with the two-stage detectors, the one-stage detectors have a higher detection speed.

Because the point-cloud data contains richer geometric features, 3D object detection has attracted more and more attention. Compared with 2D object detection, 3D object detection is more challenging because it needs to process the point clouds of the scene. Chen et al. [33] projected point cloud to the bird's view and used 2D CNNs to learn the features of point cloud for 3D boxes' generation. Song and Xiao [34, 35] divided the point cloud into equally spaced 3D voxels and used 3D CNNs to learn the features of voxels to generate 3D boxes. Shi et al. [36] used PointNet++ [37] to process the point-cloud inputs for 3D boxes' generation. Besides, some methods [38, 39] estimate 3D bounding boxes based on images.

3. Method

This section introduces the proposed multimodal MOT method that tracks multiple objects at the same time and records their trajectories. The proposed MOT method includes the four main computations: object detection with Non-maximum Suppression, motion information extraction, learning deep appearance feature, and object tracking with data association. Figure 2 shows an overview of the proposed MOT method. We combine the result of 2D object detection and 3D object detection such that the location of the object can be detected robustly. Based on this, the motion information and appearance features of objects are computed respectively. Finally, the motion information and appearance features of objects are associated to track the target object.

3.1. Object Detection with NMS. The first task of the MOT is to detect the location of objects in the scene. In this paper, we propose to combine the results of 2D object detection and 3D object detection for robust object detection. We use the 2D detector, i.e., YOLOV3 [3] that is trained on the training set of the KITTI 2D object detection benchmark and uses the 3D detector, i.e., PointRCNN [5], that is trained on the training set of the KITTI 3D object detection benchmark.

The 2D detector processes the RGB image. The output of 2D object detection is a set of detections $D_{2d}^t = \{D_{2d}^1, D_{2d}^2, \dots, D_{2d}^{n_1}\}$, where n_1 is the number of objects at frame t . The 3D detector processes the point clouds that were collected from a LiDAR. The output of 3D object detection is $D_{3d}^t = \{D_{3d}^1, D_{3d}^2, \dots, D_{3d}^{n_2}\}$, where n_2 is the number of objects at frame t . For further calculation, we project the LiDAR point in the 3D space into the 2D space according to combine camera and LiDAR calibration:

$$y = P_{\text{rect}} * R_{\text{rect}} * T_{\text{proj}} * x, \quad (1)$$

where y is the projected point in the RGB image. x denotes the 3D LiDAR point. P_{rect} and R_{rect} are the intrinsic camera parameters. The P_{rect} is the camera matrix, and the R_{rect} is the rectification matrix to make the image co-planar. T_{proj} projects the point X in the LiDAR coordinates onto the camera coordinate system. Both the intrinsic and extrinsic parameters are available in the KITTI dataset [40]. Figure 3 shows an example of point projections.

After the 3D point clouds are projected onto the image, two overlapping boxes will appear on the same object. This paper further uses the Nonmaximum Suppression (NMS) algorithm to get rid of the extra boxes. The NMS sorts all detection boxes on the basis of their scores and selects box M with the highest score. All other detection boxes with the large overlapping area with M are suppressed by using a predefined threshold N_t :

$$S_i = \begin{cases} S_i, & \text{IOU}(M, b_i) < N_t, \\ 0, & \text{IOU}(M, b_i) \geq N_t, \end{cases} \quad (2)$$

$$\text{with IOU}(M, b_i) = \frac{\text{area}(M) \cap \text{area}(b_i)}{\text{area}(M) \cup \text{area}(b_i)}$$

where b_i is the detection box to be screened, when $\text{IOU}(M, b_i)$ is greater than N_t , b_i will be removed. In our experiment, N_t is set to 0.7. Figure 4 shows a comparison result by the detection method without NMS and with NMS.

3.2. Learning Object Appearance Features. Before implementing the MOT, we need to extract the appearance features of the object. This paper employs CNN to automatically learn the deep appearance features of objects from raw visual data. The CNN is trained on a large-scale benchmark dataset [41]. The dataset contains over 50,000 images of 776 vehicles captured by 20 cameras. Figure 5 shows several samples in this dataset.

Table 1 illustrates the architecture of CNN used in this paper. The CNN model is inspired by the wide residual

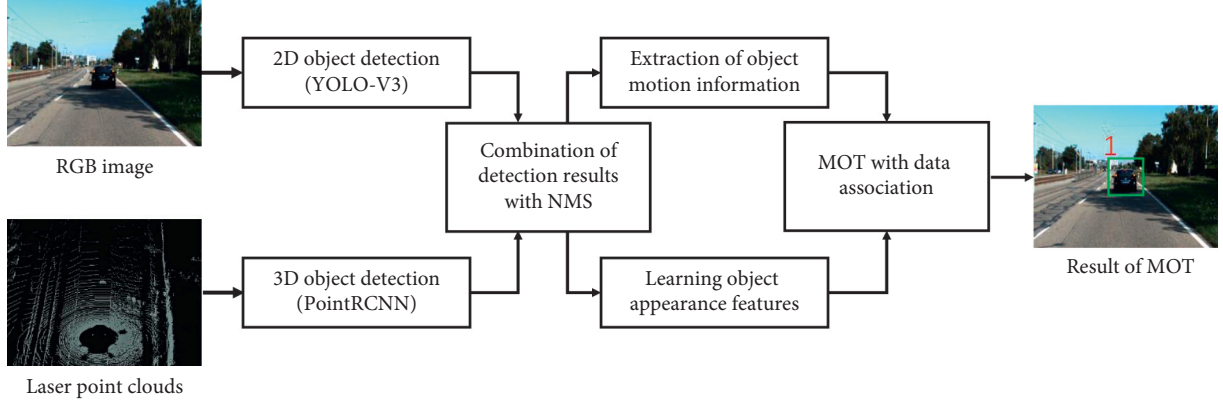


FIGURE 2: An overview of the proposed multimodal MOT method.

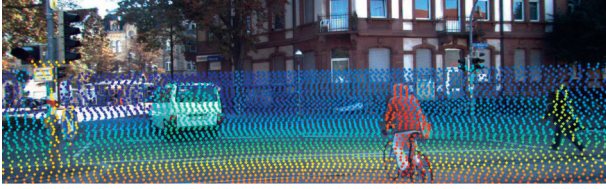


FIGURE 3: Result of point projections, where the 3D LiDAR points are projected onto the RGB image.

network [40, 42] that consists of two convolution layers and six residual blocks. The Dense layer 10 extracts a 128-dimensional global feature. The final batch and l_2 -norm layer projects feature to a unit hypersphere. By resizing the tracked vehicle image to 224×224 , then inputting it into the network. Finally, we get a 128-dimensional feature vector that is used as the deep appearance features of the object.

3.3. Extraction of Object Motion Information. Since the objects are usually moving rather than stationary, it is necessary to extract the motion information of objects for the MOT. This paper employs the Kalman filter to predict the state of the object and then extract its motion information. We use eight parameters $x_k = (a, b, \gamma, h, \dot{a}, \dot{b}, \dot{\gamma}, \dot{h})$ to describe the tracking state at frame k , where (a, b) is the bounding box center position, γ is the aspect ratio, h is the height of the bounding box, and $(\dot{a}, \dot{b}, \dot{\gamma}, \dot{h})$ represents the corresponding velocity in the image coordinate system.

Because the interval of time between each frame is very short, it can be regarded as a linear model of constant-velocity motion. We get the predicted object state at the next frame and calculate the error covariance matrix P_k^- between the predicted state and the true state:

$$\begin{aligned} x_k^- &= Ax_{k-1}, \\ P_k^- &= AP_{k-1}A^T + Q, \end{aligned} \quad (3)$$

where x_k^- is the predicted object state at frame k . A is a state transition matrix, and x_{k-1} is the object state at frame $k-1$. And Q is the covariance matrix of the predict noise. Then, we can get the Kalman gain matrix K and calculate the estimated state x_k :

$$\begin{aligned} K_k &= P_k^- H^T (HP_k^- H^T + R)^{-1}, \\ x_k &= x_k^- + K_k (z_k - Hx_k^-), \end{aligned} \quad (4)$$

where z_k is the measured value and H is the conversion matrix from x_k^- to z_k . R is the covariance matrix of the measurement noise. Finally, update the covariance matrix P_k :

$$P_k = (I - K_k H) P_k^-. \quad (5)$$

3.4. Object Tracking Based on Data Association. The next is to associate the deep appearance features and the motion information of the object for the MOT. First, this paper uses the Mahalanobis distance to compare the motion correlation between the predicted state of the Kalman Filter and the newly detected bounding boxes:

$$d^{(1)}(i, j) = (d_j - y_i)^T S_i^{-1} (d_j - y_i), \quad (6)$$

where d_j denotes the j th bounding box detection, y_i and S_i represent the mean and covariance of the i th predicted bounding box. A threshold can be adjusted to control the minimum confidence of the motion information association between objects i and j . We denote this decision with an indicator $b_{i,j}^{(1)}$, as shown in equation (7). The indicator will be equal to 1 if the Mahalanobis distance is smaller or equal to a threshold $t^{(1)}$, which is set to 9.4877 for our four-dimensional measurement space:

$$b_{i,j}^{(1)} = \begin{cases} 1, & d^{(1)}(i, j) \leq t^{(1)}, \\ 0, & d^{(1)}(i, j) > t^{(1)}. \end{cases} \quad (7)$$

Next, the above method is only a suitable related measurement index when motion uncertainty is very low. However, in the image space, only using the Kalman filter framework is a rough prediction. Therefore, this paper also adopted the second metric. It measures the smallest cosine distance of the appearance features between the i th track and j th detection as follows:

$$d^{(2)}(i, j) = \min(1 - r_j^T r_k^{(i)}), \quad (8)$$



FIGURE 4: The comparison results by the detection method without NMS (a) and with NMS (b).



FIGURE 5: Some samples in the VERI dataset. From top to bottom are cars, trucks, and buses.

TABLE 1: The architecture of the used CNN.

Name	Patch size/stride	Output size
Conv 1	$3 \times 3/1$	$32 \times 128 \times 64$
Conv 2	$3 \times 3/1$	$32 \times 128 \times 64$
Max pool 3	$3 \times 3/2$	$32 \times 64 \times 32$
Residual 4	$3 \times 3/1$	$32 \times 64 \times 32$
Residual 5	$3 \times 3/1$	$32 \times 64 \times 32$
Residual 6	$3 \times 3/2$	$64 \times 32 \times 16$
Residual 7	$3 \times 3/1$	$64 \times 32 \times 16$
Residual 8	$3 \times 3/2$	$128 \times 16 \times 8$
Residual 9	$3 \times 3/1$	$128 \times 16 \times 8$
Dense 10	—	128
Batch and l_2 normalization	—	128

where r_j is the appearance feature vector of detection d_j and $r_k^{(i)}$ represents the feature vector of the i th tracked object at the most recent frame k . In our experiment, parameter k is set to a maximum number of 100 available vectors. In addition, in order to determine whether the appearance features are related, we introduce a binary indicator, as shown in equation (9). A threshold $t^{(2)}$ is set for this indicator on a VeRi dataset:

$$b_{i,j}^{(2)} = \begin{cases} 1, & d^{(2)}(i, j) \leq t^{(2)}, \\ 0, & d^{(2)}(i, j) > t^{(2)}. \end{cases} \quad (9)$$

Then, the Mahalanobis distance determines whether the prediction position of the Kalman filter is related to the new detection, which is especially useful for short-term prediction. And the cosine distance considers the appearance of tracking objects, which is especially useful for recovering identity after a long period of occlusion. Therefore, this paper combines the two metrics using a weighted sum:

$$d_{i,j} = \lambda d^{(1)}(i, j) + (1 - \lambda) d^{(2)}(i, j), \quad (10)$$

$$\text{with } b_{i,j} = \prod_{m=1}^2 b_{i,j}^{(m)},$$

where we call an association admissible if $b_{i,j}^{(1)} = 1$ and $b_{i,j}^{(2)} = 1$. The hyperparameter λ is used to control the influence of each metric on the combined association. For example, when there is substantial object motion, the prediction of the constant-velocity motion model becomes less effective. Thus, the appearance metric becomes more significant by reducing the value of λ ; on the contrary, when there are limited vehicles on the road without long-term partial occlusions, increasing the value of λ can improve the importance of distance metric.

Finally, in our implementation, the maximum number of frames allowed to lose the target A_{\max} is considered. In order to avoid redundant computations, if a tracked object is not re-identified in the most recent A_{\max} frames passed since

its last instantiation, it will be assumed that it has left the scene. If the object is seen again, a new ID will be assigned to it. The judgement of a new track is that an object in the result of detection can never be associated with the existing MOT methods. If the prediction of the object position can be correctly correlated with the detection in the consecutive F_{\min} frames, we can confirm that a new track target has appeared.

4. Experiment

This section introduces the dataset, evaluation metric, training parameters, and experimental evaluation results in the experiments on the KITTI Tracking Benchmark.

4.1. Dataset. The proposed method was evaluated on the KITTI tracking benchmark [43]. The KITTI dataset was collected under 4 different scenarios, including city, residential, road, and campus. Some samples of the KITTI dataset are shown in Figure 6. The dataset consists of 21 training sequences and 29 test sequences. In each sequence, LIDAR point clouds, RGB images, and calibration files were provided. In the training sequences, eight different classes were labeled, including car, pedestrian, and cyclist. The objects in images were annotated with 3D and 2D bounding boxes between different frames and had a unique ID. In this work, we used all 29 testing sequences for modal validation and only used on the car subset for model evaluation because it had the most instances of all object types.

4.2. Evaluation Metric. The indexes used to evaluate the performance of the proposed MOT method were as follows:

- (1) Mostly Tracked (MT): objects are successfully tracked to at least 80% of their trajectories during their life span.
- (2) Mostly Lost (ML): objects are successfully tracked to less than 20% of their trajectories during their life span.
- (3) Identity Switches (IDS): the number of times objects' identities have changed during their life span.
- (4) Fragmentation (Frag): due to the missing detection, the number of times a trajectory is interrupted.
- (5) FP and FN: the total number of false positives and false negatives (missed targets).
- (6) Multiobject Tracking Accuracy (MOTA): it combines three error sources, i.e., FP, FN, and IDS as follows [44]. Equation (11) shows the computation of the MOTA, where t is the index of the frame and G is the number of the ground truth:

$$\text{MOTA} = 1 - \frac{\sum_t (\text{FP}_t + \text{FN}_t + \text{IDS}_t)}{\sum_t G_t}. \quad (11)$$

- (7) Multiobject Tracking Precision (MOTP): the alignment accuracy between the annotated and the predicted bounding boxes [44].

4.3. Training Parameters. This paper trained the 2D detector, i.e., the YOLOv3, on the training set of the KITTI 2D object detection benchmark [5], and trained the 3D detector, i.e., the PointRCNN, on the training set of the KITTI 3D object detection benchmark [36]. The IOU threshold N_t of the NMS module was set to 0.7. The minimum number of matched frames required to create a new trajectory F_{\min} is set to 3 and the maximum number of frames allowed to lose the target $A_{\max} = 30$. And because the prediction results of Kalman Filter is rough and there are many scenes with long-term partial occlusions in the KITTI dataset, we set $\lambda = 0.1$.

4.4. Qualitative Evaluation. We evaluated the proposed tracking method qualitatively by using the KITTI test sequence. Different scenarios including occlusions, clutter, parked vehicles, and false positives from detectors were considered in the qualitative evaluation.

Figure 7 shows an example of the test sequence 0 in the test set. Each vehicle was assigned a tracking ID as a reference. Despite the compact and messy parking of the vehicle, the proposed MOT method can continuously detect and track the vehicles. Moreover, from this figure, we can see that, since the image is easily affected by the environment, such as illumination changes and partial occlusion, the shape of the detected target will change. In addition, the scale of the target object may be very different. In this case, the proposed MOT method still obtained a relatively high tracking performance. The experimental results show that our method can locate each car well even in the cluttered and strong lighting scene and maintain the ID of the car unchanged.

Figure 8 shows another example from the test sequence 1. Figure 8(a) shows that the object detector produces a false detection result, and Figure 8(b) shows the false positive of the detector is overcome by data association. In the case of transient errors in object detection, the proposed MOT method can still track the target stably. Hence, these experimental results demonstrated the robustness of the proposed MOT method.

4.5. Benchmark Results. We further evaluated the proposed MOT method on the KITTI Tracking Benchmark. In this evaluation, we considered some published online MOT methods for comparison. The results are presented in Table 2. It can be seen that the proposed MOT method is very competitive. In particular, the proposed MOT method returns the fewest number of identity switches, while maintaining competitive MOTA scores, MOTP scores, and track fragmentations. The tracking accuracy is mainly affected by a large number of false positives. Given their overall impact on the MOTA score, the combination of the 2D and 3D object detection results can significantly improve the performance of the MOT. Besides, because we set the maximum allowed trackage and associate the object motion information and appearance features, the proposed MOT method has the fewest number of identity switches. Therefore, the proposed MOT method can generate a relatively stable trajectory of the target object.

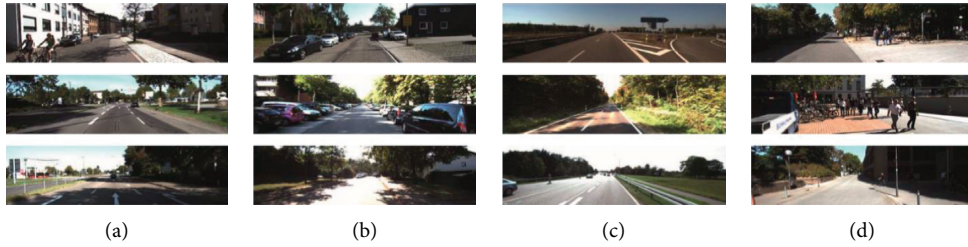


FIGURE 6: Some samples in the KITTI dataset. (a) City. (b) Residential. (c) Road. (d) Campus.



FIGURE 7: Result of the qualitative evaluation with the test sequence 0. Each vehicle is assigned a tracking ID as a reference.

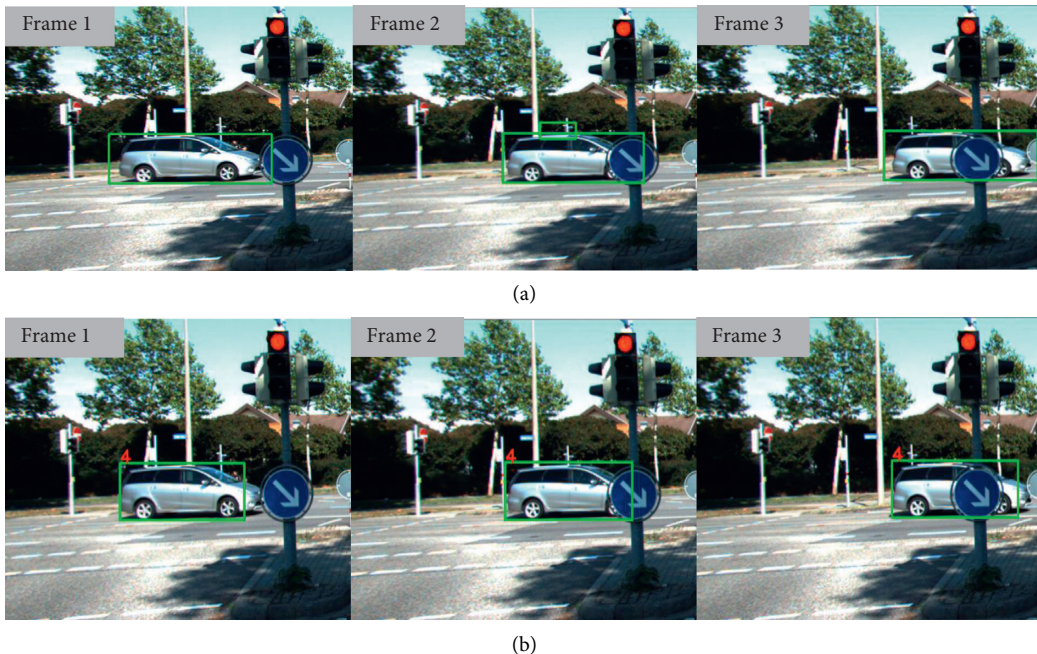


FIGURE 8: Result of the qualitative evaluation with the test sequence 1. (a) The detector generates an error detection and disappears in the next frame. (b) The proposed MOT method can track objects stably.

4.6. Ablation Study. The ablation study was to evaluate the effects of hyperparameters on the performance of the proposed MOT method. Table 3 shows the results of the ablation study on the KITTI benchmark. The hyperparameter N_t is the threshold of IOU, and the F_{\min} denotes the minimum number of matching frames required to create a new trajectory. From the table, we can be seen that when $N_t = 0.6$, this may miss some

correct detection results. That is because the number of detected objects is reduced. When $N_t = 0.8$, this may result in some wrong detection results, which is also the reason why it has the most IDS. $F_{\min} = 1$ means that track immediately when a new target is detected, which leads to more IDS and FRAG. The $F_{\min} = 5$ makes the minimum IDS, but MOTA is lower. Therefore, we finally set $N_t = 0.7$ and $F_{\min} = 3$.

TABLE 2: Comparison on KITTI tracking benchmark.

Method	MOTA (%)↑	MOTP (%)↑	MT (%)↑	ML (%)↓	IDS↓	FRAG↓
SASN-MCF nano [45]	70.86	82.65	58.00	7.85	443	975
SSP [46]	72.72	78.55	53.85	8.00	185	932
CIWT [12]	75.39	79.25	49.85	10.31	165	660
Complexer-YOLO [47]	75.70	78.46	58.00	5.08	1186	2092
DSM [48]	76.15	83.42	60.00	8.31	296	868
Ours	76.40	83.50	47.38	14.00	147	608

TABLE 3: Ablation study on the training set of KITTI tracking benchmark.

Parameter	MOTA (%)↑	MOTP (%)↑	MT (%)↑	ML (%)↓	IDS↓	FRAG↓
$N_t = 0.6$	71.27	82.92	43.73	8.42	59	409
$N_t = 0.8$	70.09	83.11	49.64	6.81	289	559
$F_{\min} = 1$	72.05	82.70	46.01	5.02	161	548
$F_{\min} = 5$	68.13	83.30	32.97	13.26	35	229
Ours	72.36	83.94	47.31	7.36	94	395

5. Conclusion

This paper proposed a multimodal MOT method by fusing the motion information and the deep appearance feature of objects. In this method, we use a Nonmaximum Suppression algorithm to combine a 2D object detector and a 3D object detector for robust object detection. Then, the deep appearance features of objects are learned by a CNN, and the motion information of objects is computed by the Kalman Filter. The MOT task is achieved by associating the appearance features and the motion information of the target object. The effectiveness of the proposed MOT method was demonstrated in a set of experiments. The proposed MOT method can track objects stably in crowded scenes and effectively avoid false detection. In the KITTI tracking benchmark, the proposed method also shows competitive results.

Although 3D object detection is used in the proposed MOT method, it is only used as the auxiliary information for 2D object detection. 3D object detection can provide accurate position and size estimation for automatic driving. Therefore, our future work will be towards the direction of 3D multitarget tracking that can adapt to a more complex environment.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest regarding the publication of this paper.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Project no. 61673115). This work was also partly funded by the German Science Foundation (DFG) and National Science Foundation of China (NSFC) in

project Cross Modal Learning under contract Sonderforschungsbereich Transregio 169.

References

- [1] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *Neural Information Processing Systems*, vol. 39, pp. 91–99, 2015.
- [2] W. Liu, "SSD: single Shot MultiBox detector," in *Proceedings of the European Conference on Computer Vision*, pp. 21–37, Amsterdam, Netherlands, October 2016.
- [3] R. Joseph and A. Farhadi, "Yolov3: an incremental improvement," 2018, <https://arxiv.org/pdf/1804.02767.pdf>.
- [4] C. Zhu, Y. He, and M. Savvides, "Feature selective anchor-free module for single-shot object detection," in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 840–849, Hubballi, India, December 2019.
- [5] D. Maturana and S. Scherer, "VoxNet: a 3D convolutional neural network for real-time object recognition," in *Proceedings of the Intelligent Robots and Systems*, pp. 922–928, Hamburg, Germany, September 2015.
- [6] S. Sharma, J. A. Ansari, J. K. Murthy, and K. M. Krishna, "Beyond pixels: leveraging geometry and shape cues for online multi-object tracking," in *Proceedings of the International Conference on Robotics and Automation*, pp. 3508–3515, Brisbane, Australia, May 2018.
- [7] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier et al., "Online multiperson tracking-by-detection from a single, uncalibrated camera," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 9, pp. 1820–1833, 2011.
- [8] H. Zhou, W. Ouyang, J. Cheng, X. Wang et al., "Deep continuous conditional random fields with asymmetric inter-object constraints for online multi-object tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 4, pp. 1011–1022, 2019.
- [9] W. Luo, B. Yang, and R. Urtasun, "Fast and furious: real time end-to-end 3D detection, tracking and motion forecasting with a single convolutional net," in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 3569–3577, Salt Lake City, UT, USA, June 2018.
- [10] N. Smolyanskiy, A. Kamenev, and S. Birchfield, "On the importance of stereo for accurate depth estimation: an

- efficient semi-supervised deep neural network approach,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 1007–1015, Salt Lake City, UT, USA, June 2018.
- [11] A. Asvadi, P. Girao, P. Peixoto, and U. Nunes, “3D object tracking using RGB and LIDAR data,” in *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, Rio de Janeiro, Brazil, November 2016.
- [12] A. Osep, W. Mehner, M. Mathias, and B. Leibe, “Combined image- and world-space tracking in traffic scenes,” in *Proceedings of the International Conference on Robotics and Automation*, pp. 1988–1995, Singapore, June 2017.
- [13] E. Levinkov, J. Uhrig, S. Tang, M. Omran et al., “Joint graph decomposition & node labeling: problem, algorithms, applications – supplement –,” in *Proceedings of the IEEE Conf. Comput. Vis. Pattern Recognit*, pp. 1904–1912, Honolulu, HI, USA, July 2017.
- [14] M. Keuper, S. Tang, B. Andres, T. Brox et al., “Motion segmentation & multiple object tracking by correlation co-clustering,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 1, pp. 140–153, 2020.
- [15] B. Wang, G. Wang, K. L. Chan, and L. Wang, “Tracklet association with online target-specific metric learning,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 1234–1241, Columbus, OH, USA, June 2014.
- [16] C. Wang, H. Liu, and Y. Gao, “Scene-adaptive hierarchical data association for multiple objects tracking,” *IEEE Signal Processing Letters*, vol. 21, no. 6, pp. 697–701, 2014.
- [17] N. Wojke, A. Bewley, and D. Paulus, “Simple online and realtime tracking with a deep association metric,” in *Proceedings of the International Conference on Image Processing*, pp. 3645–3649, Beijing, China, September 2017.
- [18] L. Zhang and L. V. Der Maaten, “Structure preserving object tracking,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 1838–1845, Cancun, Mexico, December 2013.
- [19] S. H. Bae and K. J. Yoon, “Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning,” in *Proceedings of the Computer Vision and Pattern Recognition (CVPR)*, Columbus, OH, USA, June 2014.
- [20] F. Poiesi, R. Mazzon, A. Cavallaro, and I. U. Cviu, “Multi-target tracking on confidence maps: an application to people tracking,” *Computer Vision and Image Understanding*, vol. 117, no. 10, pp. 1257–1272, 2013.
- [21] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, “Simple online and realtime tracking,” in *Proceedings of the International Conference on Image Processing*, pp. 3464–3468, Las Vegas, NV, USA, January 2016.
- [22] B. Liu, S. Cheng, and Y. Shi, “Particle filter optimization: a brief introduction,” in *Proceedings of the International Conference on Swarm Intelligence*, pp. 95–104, Springer, Cham, Switzerland, 2016.
- [23] S. A. Goli, B. H. Far, and A. O. Fapojuwo, “An accurate multi-sensor multi-target localization method for cooperating vehicles,” in *Information Reuse and Integration*, pp. 197–217, Springer, 2016.
- [24] B. D. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” in *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 674–679, Vancouver, Canada, August 1981.
- [25] L. Chen, H. Ai, R. Chen, and Z. Zhuang, “Aggregate tracklet appearance features for multi-object tracking,” *IEEE Signal Processing Letters*, vol. 26, no. 11, pp. 1613–1617, 2019.
- [26] X. Su, X. Qu, Z. Zou, P. Zhou et al., “K-reciprocal harmonious attention network for video-based person Re-identification,” *IEEE Access*, vol. 7, pp. 22457–22470, 2019.
- [27] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 580–587, Columbus, OH, USA, June 2014.
- [28] R. Girshick, “Fast R-CNN,” in *Proceedings of the International Conference on Computer Vision*, pp. 1440–1448, Santiago, Chile, December 2015.
- [29] T. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 936–944, Honolulu, HI, USA, July 2017.
- [30] J. Redmon, S. K. Divvala, R. Girshick, and A. Farhadi, “You only Look once: unified, real-time object detection,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 779–788, Las Vegas, NV, USA, June 2016.
- [31] J. Redmon and A. Farhadi, “YOLO9000: better, faster, stronger,” in *Proceedings of the computer vision and pattern recognition*, pp. 6517–6525, Honolulu, HI, USA, July 2017.
- [32] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in *Proceedings of the International Conference on Computer Vision*, pp. 2999–3007, Venice, Italy, July, 2017.
- [33] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, “Multi-view 3D object detection network for autonomous driving,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 6526–6534, Honolulu, HI, USA, July 2017.
- [34] S. Song and J. Xiao, “Deep sliding shapes for amodal 3D object detection in RGB-D images,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 808–816, Las Vegas, NV, USA, June 2016.
- [35] Y. Zhou and O. Tuzel, “VoxelNet: end-to-end learning for point cloud based 3D object detection,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 4490–4499, Salt Lake City, UT, USA, June 2018.
- [36] S. Shi, X. Wang, and H. Li, “PointRCNN: 3D object proposal generation and detection from point cloud,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 770–779, Long Beach, CA, USA, June 2019.
- [37] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, “PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space,” in *Proceedings of the NIPS*, Long Beach, CA, USA, December 2017.
- [38] J. Ku, A. D. Pon, and S. L. Waslander, “Monocular 3D object detection leveraging accurate proposals and shape reconstruction,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 11867–11876, Long Beach, CA, USA, June 2019.
- [39] Z. Deng and L. J. Latecki, “Amodal detection of 3D objects: inferring 3D bounding boxes from 2D ones in RGB-depth images,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 398–406, Honolulu, HI, USA, July 2017.
- [40] S. Zagoruyko, N. J. Komodakis, and P. Recognition, “Wide residual networks,” in *Proceedings of the BMVC*, New York, NY, USA, September 2016.
- [41] X. Liu, W. Liu, T. Mei, and H. Ma, “PROVID: progressive and multimodal vehicle reidentification for large-scale urban surveillance,” *IEEE Transactions on Multimedia*, vol. 20, no. 3, pp. 645–658, 2018.
- [42] N. Wojke and A. Bewley, “Deep cosine metric learning for person Re-identification,” in *Proceedings of the Workshop on Applications of Computer Vision*, pp. 748–756, Vancouver, Canada, September 2018.

- [43] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Providence, RI, USA, June 2012.
- [44] K. Bernardin, R. Stiefelhagen, and V. Processing, "Evaluating multiple object tracking performance: the CLEAR MOT metrics," *EURASIP Journal on Image and Video Processing*, vol. 2008, 10 pages, 2008.
- [45] G. Gunduz and TAJITOI Vehicles, *Efficient Multi-Object Tracking by Strong Associations on Temporal Window*, IEEE, Newyork, NY, USA, 2019.
- [46] P. Lenz, A. Geiger, and R. Urtasun, "FollowMe: efficient online min-cost flow tracking with bounded memory and computation," in *Proceedings of the International Conference on Computer Vision*, pp. 4364–4372, Santiago, Chile, December 2015.
- [47] M. Simon, "Complexer YOLO: real-time 3D object detection and tracking on semantic point clouds," in *Proceedings of the Computer Vision and Pattern Recognition*, p. 0, Long Beach, CA, USA, June 2019.
- [48] D. Frossard and R. Urtasun, "End-to-end learning of multi-sensor 3D tracking by detection," in *Proceedings of the International Conference on Robotics and Automation*, pp. 635–642, Brisbane, Australia, May 2018.