Wireless Communications and Mobile Computing

Conference Issue: Intelligent Media Computing Technology and Applications for Mobile Internet

Lead Guest Editor: Chien Aun Chan
Guest Editors: Ming Yan, Yun Tie, and Ivan Lee

Conference Issue: Intelligent Media Computing Technology and Applications for Mobile Internet

Wireless Communications and Mobile Computing

Conference Issue: Intelligent Media Computing Technology and Applications for Mobile Internet

Lead Guest Editor: Chien Aun Chan
Guest Editors: Ming Yan, Yun Tie, and Ivan Lee

Copyright © 2022 Hindawi Limited. All rights reserved.

This is a special issue published in “Wireless Communications and Mobile Computing.” 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

Zhipeng Cai, USA

Associate Editors

Ke Guan, China
Jaime Lloret, Spain
Maode Ma, Singapore

Academic Editors

Muhammad Inam Abbasi, Malaysia
Ghufran Ahmed, Pakistan
Hamza Mohammed Ridha Al-Khafaji, Iraq
Abdullah Alamoodi, Malaysia
Marica Amadeo, Italy
Sandhya Aneja, USA
Mohd Dilshad Ansari, India
Eva Antonino-Daviu, Spain
Mehmet Emin Aydin, United Kingdom
Parameshachari B. D., India
Kalapraveen Bagadi, India
Ashish Bagwari, India
Dr. Abdul Basit, Pakistan
Alessandro Bazzi, Italy
Zdenek Becvar, Czech Republic
Nabil Benamar, Morocco
Olivier Berder, France
Petros S. Bithas, Greece
Dario Bruneo, Italy
Jun Cai, Canada
Xuesong Cai, Denmark
Gerardo Canfora, Italy
Rolando Carrasco, United Kingdom
Vicente Casares-Giner, Spain
Brijesh Chaurasia, India
Lin Chen, France
Xianfu Chen, Finland
Hui Cheng, United Kingdom
Hsin-Hung Cho, Taiwan
Ernestina Cianca, Italy
Marta Cimitile, Italy
Riccardo Colella, Italy
Mario Collotta, Italy
Massimo Condoluci, Sweden
Antonino Crivello, Italy
Antonio De Domenico, France
Floriano De Rango, Italy
Antonio De la Oliva, Spain
Margot Deruyck, Belgium
Liang Dong, USA
Praveen Kumar Donta, Austria
Zhuojun Duan, USA
Mohammed El-Hajjar, United Kingdom
Oscar Esparza, Spain
Maria Fazio, Italy
Mauro Femminella, Italy
Manuel Fernandez-Veiga, Spain
Gianluigi Ferrari, Italy
Luca Foschini, Italy
Alexandros G. Fragkiadakis, Greece
Ivan Ganchev, Bulgaria
Óscar García, Spain
Manuel García Sánchez, Spain
L. J. García Villalba, Spain
Miguel Garcia-Pineda, Spain
Piedad Garrido, Spain
Michele Girolami, Italy
Mariusz Glabowski, Poland
Carles Gomez, Spain
Antonio Guerrieri, Italy
Barbara Guidi, Italy
Rami Hamdi, Qatar
Tao Han, USA
Sherief Hashima, Egypt
Mahmoud Hassaballah, Egypt
Yejun He, China
Yixin He, China
Andrej Hrovat, Slovenia
Chunqiang Hu, China
Xuexian Hu, China
Zhenghua Huang, China
Xiaohong Jiang, Japan
Vicente Julian, Spain
Rajesh Kaluri, India
Dimitrios Katsaros, Greece
Muhammad Asghar Khan, Pakistan
Rahim Khan, Pakistan
Ahmed Khattab, Egypt
Hasan Ali Khattak, Pakistan
Mario Kolberg, United Kingdom
Meet Kumari, India
Wen-Cheng Lai, Taiwan

Jose M. Lanza-Gutierrez, Spain
Pavlos I. Lazaridis, United Kingdom
Kim-Hung Le, Vietnam
Tuan Anh Le, United Kingdom
Xianfu Lei, China
Jianfeng Li, China
Xiangxue Li, China
Yaguang Lin, China
Zhi Lin, China
Liu Liu, China
Mingqian Liu, China
Zhi Liu, Japan
Miguel López-Benítez, United Kingdom
Chuanwen Luo, China
Lu Lv, China
Basem M. ElHalawany, Egypt
Imadeldin Mahgoub, USA
Rajesh Manoharan, India
Davide Mattera, Italy
Michael McGuire, Canada
Weizhi Meng, Denmark
Klaus Moessner, United Kingdom
Simone Morosi, Italy
Amrit Mukherjee, Czech Republic
Shahid Mumtaz, Portugal
Giovanni Nardini, Italy
Tuan M. Nguyen, Vietnam
Petros Nicopolitidis, Greece
Rajendran Parthiban, Malaysia
Giovanni Pau, Italy
Matteo Petracca, Italy
Marco Picone, Italy
Daniele Pinchera, Italy
Giuseppe Piro, Italy
Javier Prieto, Spain
Umair Rafique, Finland
Maheswar Rajagopal, India
Sujan Rajbhandari, United Kingdom
Rajib Rana, Australia
Luca Reggiani, Italy
Daniel G. Reina, Spain
Bo Rong, Canada
Mangal Sain, Republic of Korea
Praneet Saurabh, India
Hans Schotten, Germany
Patrick Seeling, USA
Muhammad Shafiq, China
Zaffar Ahmed Shaikh, Pakistan
Vishal Sharma, United Kingdom
Kaize Shi, Australia
Chakchai So-In, Thailand
Enrique Stevens-Navarro, Mexico
Sangeetha Subbaraj, India
Tien-Wen Sung, Taiwan
Suhua Tang, Japan
Pan Tang, China
Pierre-Martin Tardif, Canada
Sreenath Reddy Thummaluru, India
Tran Trung Duy, Vietnam
Fan-Hsun Tseng, Taiwan
S Velliangiri, India
Quoc-Tuan Vien, United Kingdom
Enrico M. Vitucci, Italy
Shaohua Wan, China
Dawei Wang, China
Huaqun Wang, China
Pengfei Wang, China
Dapeng Wu, China
Huaming Wu, China
Ding Xu, China
YAN YAO, China
Jie Yang, USA
Long Yang, China
Qiang Ye, Canada
Changyan Yi, China
Ya-Ju Yu, Taiwan
Marat V. Yuldashev, Finland
Sherali Zeadally, USA
Hong-Hai Zhang, USA
Jiliang Zhang, China
Lei Zhang, Spain
Wence Zhang, China
Yushu Zhang, China
Kechen Zheng, China
Fuhui Zhou, USA
Meiling Zhu, United Kingdom
Zhengyu Zhu, China

Contents

Wireless Communications and Mobile Computing Multitrack Music Generation Network Based on Music Rules
Yun Tie, Tao Wang, Cong Jin, Xiaobing Li, and Ping Yang
Research Article (7 pages), Article ID 5845689, Volume 2022 (2022)

Objective Evaluation Method of Broadcasting Vocal Timbre Based on Feature Selection
Chaohui Lv, Hua Lan, Ying Yu, and Shengnan Li
Research Article (17 pages), Article ID 7086599, Volume 2022 (2022)

Attention-Based Bi-DLSTM for Sentiment Analysis of Beijing Opera Lyrics
Cong Jin, Zhen Song, Jiaqi Xu, and Huiyue Gao
Research Article (8 pages), Article ID 1167462, Volume 2022 (2022)

Multiple HD Screen-Based Virtual Studio System with Learned Mask-Free Portrait Harmonization
Jimao Jiang, Jiaxin Lin, Yuntao Su, Li Fang, and Long Ye
Research Article (10 pages), Article ID 6014795, Volume 2022 (2022)

A Renovated CNN-Based Model Enhances KGC Task Performance
Fang Miao, Xueting Wang, Feng Feng, Cong Jin, and Libiao Jin
Research Article (12 pages), Article ID 5968047, Volume 2022 (2022)

Flexible Light Field Angular Superresolution via a Deep Coarse-to-Fine Framework
Qian Wang, Li Fang, Long Ye, Wei Zhong, Fei Hu, and Qin Zhang
Research Article (10 pages), Article ID 4570755, Volume 2022 (2022)

MTAL: A Novel Chinese Herbal Medicine Classification Approach with Mutual Triplet Attention Learning
Wangli Hao, Meng Han, Shancang Li, and Fuzhong Li
Research Article (9 pages), Article ID 8034435, Volume 2022 (2022)

ALBLP: Adaptive Load-Balancing Architecture Based on Link-State Prediction in Software-Defined Networking
Junyan Chen, Yong Wang, Xuefeng Huang, Xiaolan Xie, Hongmei Zhang, and Xiaoye Lu
Research Article (16 pages), Article ID 8354150, Volume 2022 (2022)

Relative Pose Estimation for RGB-D Human Input Scans via Implicit Function Reconstruction
Pengpeng Liu, Tao Yu, Zhi Zeng, Yebin Liu, Guixuan Zhang, and Zhen Song
Research Article (9 pages), Article ID 4351951, Volume 2022 (2022)

DFAN: Dual Feature Aggregation Network for Lightweight Image Super-Resolution
Shang Li, Guixuan Zhang, Zhengxiong Luo, and Jie Liu
Research Article (12 pages), Article ID 8116846, Volume 2022 (2022)

Dynamics of 2SIH2R Rumor-Spreading Model in a Heterogeneous Network
Yan Wang, Feng Qing, and Ming Yan
Research Article (13 pages), Article ID 7398387, Volume 2022 (2022)

A Novel Vehicle Detection Framework Based on Parallel Vision
Ying Zhuo, Lan Yan, Wenbo Zheng, Yutian Zhang, and Chao Gou
Research Article (11 pages), Article ID 9667506, Volume 2022 (2022)

Knowledge Graph Construction for Intelligent Media Based on Mobile Internet
Jianhua Dai and Jingxin Xu
Research Article (14 pages), Article ID 4867220, Volume 2022 (2022)

A Robust Invariant Local Feature Matching Method for Changing Scenes
Di Wang, Hongying Zhang, and Yanhua Shao
Research Article (13 pages), Article ID 8927822, Volume 2021 (2021)

A Collaborative Cache Strategy in Satellite-Ground Integrated Network Based on Multiaccess Edge Computing
Li Yang, Xiangguang Kong, Yaowen Qi, and Chengsheng Pan
Research Article (14 pages), Article ID 8121509, Volume 2021 (2021)

"Avatar to Person" (ATP) Virtual Human Social Ability Enhanced System for Disabled People
Zichun Guo, Zihao Wang, and Xueguang Jin
Research Article (10 pages), Article ID 5098992, Volume 2021 (2021)

Enhancing Feature Point-Based Video Watermarking against Geometric Attacks with Template
Zhongze Lv, Hu Guan, Ying Huang, Shuwu Zhang, and Yang Zheng
Research Article (9 pages), Article ID 3364680, Volume 2021 (2021)

Dynamic Mathematical Model of Information Spreading on News Platform
Guohui Song, Yongbin Wang, and Yugang Li
Research Article (11 pages), Article ID 2174190, Volume 2021 (2021)

SAL-Net: Self-Supervised Attribute Learning for Object Recognition and Segmentation
Shu Yang, JingWang, Sheeraz Arif, Minli Jia, and Shunan Zhong
Research Article (13 pages), Article ID 2891303, Volume 2021 (2021)

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 5845689, 7 pages
https://doi.org/10.1155/2022/5845689

Research Article

Wireless Communications and Mobile Computing Multitrack Music Generation Network Based on Music Rules

Yun Tie,[1,2] Tao Wang,[1] Cong Jin,[3] Xiaobing Li,[2] and Ping Yang[4]

[1]*School of Information and Engineering, Zhengzhou University, 450001, China*
[2]*Department of Music Artificial Intelligence, Central Conservatory of Music, 100031, China*
[3]*School of Information and Communication Engineering, Communication University of China, 100024, China*
[4]*Beijing Polytechnic, 100029, China*

Correspondence should be addressed to Xiaobing Li; lxiaobing@ccom.edu.cn

Received 26 September 2021; Accepted 19 July 2022; Published 29 December 2022

Academic Editor: Peng Li

Copyright © 2022 Yun Tie 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.

Multitrack music generation technology is becoming more and more mature, but the existing generation technology cannot reach the desired effect in terms of harmony and matching degree, and most of the generated music does not conform to the music theory knowledge. In order to solve these problems, we propose a multitrack music generation network based on transformer to produce music with high musicality under the guidance of music theory rules. This paper uses an improved version of transformer to learn the information inside a single-track sequence and between different tracks. Then, a combination of music theory rules and crossentropy loss is used to guide the training of the generated network, and the well-designed loss objective function is optimized while the discrimination network is trained. Compared with other multitrack music generation models, the validity of our model is proved.

1. Introduction

As the human demand for music increased, intelligent composition technology emerged. Generally, musical instruments can be divided into single-track and multitrack types, while music generation models can be divided into symbolic [1, 2] and audio techniques [3, 4]. However, piano, guitar, and bass serve as the primary instruments in contemporary music. As such, this study investigated the generation of multitrack symbolic music.

End-to-end sequence models, such as recurrent neural networks (RNNs), long short-term memory (LSTM), and hierarchical RNNs, are a common technique used for intelligent music composition in previous studies. The music models proposed for multitrack music generation include HRNN [5], MiDiNet [6], and MuseGAN [1]. A large number of experiments have found that these models only allow the network to learn the relationship between note features from the real music data, but not the harmony and rules that the composer needs to follow from the whole music. As a result, the resulting music seems to lack harmony and to be incongruous with human hearing habits.

Therefore, in order to solve all the problems encountered above, we propose a novel network, which is improved on the basis of transformer [7] to get a crosstrack transformer network which can learn the information between different tracks well and combined with discrimination network to produce multitrack music in line with the public's musical literacy under the guidance of music rules. Finally, a set of music evaluation indexes is proposed. Through the evaluation, it is found that the model proposed by us is closer to the real music works than the benchmark and other multitrack music generation models.

Our main contributions are as follows: (1) based on transformer network, a generative network based on music theory knowledge is proposed to guide the generation of confrontation network in line with human music literacy. (2) In view of the importance of the internal information of single-track sequence and the information between different track sequences in the generation of multitrack music,

transformer is improved to produce works satisfying the correlation of internal information of single-track and the harmony between different tracks. (3) In view of the importance of music theory rules in music, a new discriminant method combining music rule mathematical model and discrimination network is put forward.

2. Related Work

Many methods of music generation have been proposed by researchers. For example, in 2016, Mogren proposed a continuous recurrent neural network (C-RNN-GAN) model with confrontation training based on an RNN, to generate melody [8]. In 2018, Roberts et al. established a hierarchical RNN to generate 16-bar musical notes [9]. However, common RNN and LSTM networks cannot solve the problem of long-term dependence between contexts. As such, Huang et al. modified the relative attention mechanism in a transformer sequence model used for text translation [10] or text continuation [11] and generated a musical clip with the same pitch, length, and interval structure [12].

In the generation of multitrack music, researchers have started to use neural network with VAE, GAN, and transformer for multitrack music generation [8, 13, 14]. In 2016, Chu et al. proposed an RNN-based hierarchical model (HRNN), in which the lower structure generated melody while the high-level structures generated chords and percussion for accompaniment; compared with the traditional music generation method, the rhythm has been greatly improved [5]. In 2019, Zhang proposed a technique to generate multitrack music using a decoding structure with a transformer serving as the generator and an encoding structure functioning as the discriminator [15]. However, these models have no regular limits in terms of melody and rhythm, and the resulting samples are not ideal. Therefore, in 2020, Jin et al. proposed the MTMG model method that could well learn the relationship between different sound tracks [16]. However, the existing models of music generation are deficient in melody, rhythm, overall harmony, and matching degree, and most of the generated music does not conform to the basic knowledge of music theory.

Therefore, based on the existing achievements, this paper proposes a multitrack music generation model guided by music rules, combining transformer model and discrimination network according to the process of human music creation, and proves the effectiveness of the model through experiments.

3. Proposed Method

3.1. Data Representation. The inputs and outputs used by the model are MIDI files. In order to adapt the MIDI file to the generation task of this model, it is necessary to extract eight features of MIDI file and encode them into event sequences according to the features (see Figure 1), where each event is represented as a tuple, namely, bar, position, chord, tempo value, tempo class, note on, note velocity, and note duration [17]. Bar is the number of bars, position represents the position of each event type, chord represents the set chord

```
Event(name=Bar, time=None, value=None, text=1)
Event(name=Position, time=0, value=1/16, text=0)
Event(name=Chord, time=0, value=N:N, text=N:N)
Event(name=Position, time=0, value=1/16, text=0)
Event(name=Tempo Class, time=0, value=mid, text=None)
Event(name=Tempo Value, time=0, value=30, text=None)
Event(name=Position, time=480, value=5/16, text=480)
Event(name=Tempo Class, time=480, value=slow, text=None)
Event(name=Tempo Value, time=480, value=0, text=None)
Event(name=Position, time=960, value=9/16, text=960)
Event(name=Chord, time=960, value=G:maj, text=G:maj)
```

Figure 1: An example of a section of a MIDI file being converted into a sequence of events.

progression, tempo class represents tempo type (fast, moderate, and slow), tempo value represents the value that quantifies the rhythm type, note in indicates the start time of the pitch (pitch quantization range "0-127"), note velocity indicates the perceived loudness of notes (quantization range "0-127") and note duration indicates the length of each note.

3.2. Overall Framework. Based on transformer, this paper is oriented by music theory knowledge rules and combined with discrimination network to generate multitrack music (see Figure 2). Firstly, three tracks are encoded into time sequence, respectively, and the internal information of single-track sequence is learned through three generators, and the state y_t of the next moment is generated. Secondly, six CT-transformer modules are used to learn the sound track sequence in pairs, and the piano sequence after learning the guitar track sequence and the bass track sequence is pieced together to obtain the piano sequence containing the information of the other two tracks. The learning of the guitar track sequence and the bass track sequence is the same as that of the piano track sequence. Finally, the real sample sequence and the generated sample sequence were discriminated by discriminator D_φ, and the generation was guided by music theory rules.

3.3. Generation Network. In the generation stage, the model needs to learn two parts: first, single-track sequence information learning and generation (G_p, G_g, and G_b). Second is information learning and generation between multitrack sequences (CT-transformer).

In the single-track sequence information learning section, only the decoding portion of the transformer [10] was used in the single-track generation network. The input feature sequence was mapped to embedding through using a learning embedding matrix, used to group information after the kth step through Ng self-attention blocks (Ng is equal to 5). This masking mechanism ensures that characters refer only to information prior to time k. The output of the last self-attention block is then mapped to a vocabulary space and activated by a softmax layer to produce the output feature distribution (see Figure 3). In the pretraining stage, the single-track generator is trained to minimize the cross-entropy loss between the predicted character and the input characters. In the generation phase, characters are produced individually in an autoregressive manner.

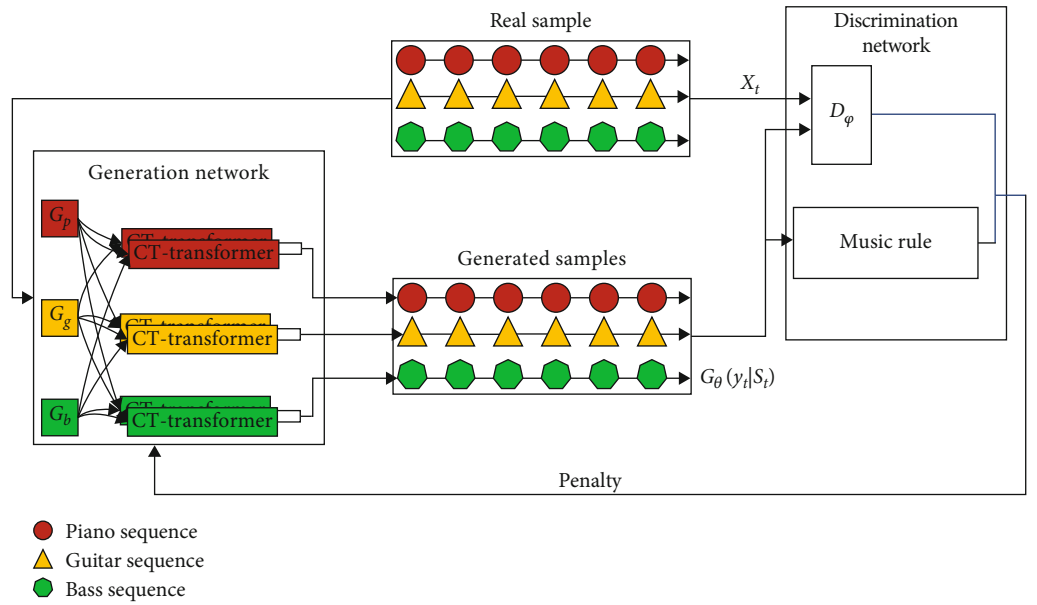


Figure 2: Multitrack music generation network framework.

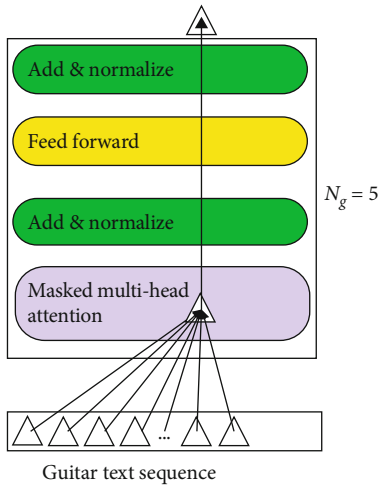


Figure 3: Single-track music generation structure.

In the part of information learning and generation between multitrack sequences, this module improves the self-attention mechanism based on transformer, namely, CT-transformer, and the core part is the crosstrack attention mechanism. The piano sequence is taken as the learning object, and the guitar sequence is taken as the learning object, respectively, represented as $X_p \in R^{T_p \times d_p}$ and $X_g \in R^{T_g \times d_g}$. $T_{(.)}$ and $d_{(.)}$ represent sequence length and feature dimension, respectively. Let us define the query as $Q_{(p)} = X_p W_{Q_p}$ and the key-value pairs as $K_{(g)} = X_g W_{K_g}$ and $V_{(g)} = X_g W_{V_g}$, where $W_{Q_p} \in R^{d_p \times d_k}$, $W_{K_g} \in R^{d_g \times d_k}$, and $W_{V_g} \in R^{d_g \times d_v}$ are weights (see Figure 4). $Z_{p \longrightarrow g}^{[i]}$ is the state obtained at the moment i, $i = [1, 2, \cdots, 6]$. At time i, the character gets $\widehat{Z}_{p \longrightarrow g}^{[i]}$ after passing through the crosstrack learning module CT-transformer. After layer normalization, it gets $f_{\theta_{p \longrightarrow g}^{[i]}}(\mathrm{LN}(\widehat{Z}_{p \longrightarrow g}^{[i]}))$ through the feedforward layer and sums with the sequence normalized by the layer to get the state $Z_{p \longrightarrow g}^{[i]}$ at the next time, as shown in Equation (1), where $Z_{p \longrightarrow g}^{[i]}$ is the piano sequence learning the guitar sequence, and the multihead crosstrack attention output sequence through layer i is shown in Equation (2); $\widehat{Z}_{p \longrightarrow g}^{[i]}$ is the sum of the previous time state after the previous time state is normalized with the layer through crosstrack learning.

$$Z_{p \longrightarrow g}^{[i]} = f_{\theta_{p \longrightarrow g}^{[i]}}\left(\mathrm{LN}\left(\widehat{Z}_{p \longrightarrow g}^{[i]}\right)\right) + \mathrm{LN}\left(\widehat{Z}_{p \longrightarrow g}^{[i]}\right), \quad (1)$$

$$\widehat{Z}_{p \longrightarrow g}^{[i]} = \mathrm{CT}\left(\mathrm{LN}\left(Z_{p \longrightarrow g}^{[i-1]}\right), Z_{p \longrightarrow g}^{[0]}\right) + \mathrm{LN}\left(Z_{p \longrightarrow g}^{[i-1]}\right), \quad (2)$$

$$\mathrm{CT}\left(X_p, X_g\right) = W[\mathrm{head}_1, , \mathrm{head}_h], \quad (3)$$

$$\mathrm{Head}_h = \mathrm{softmax}\left(\frac{Q_p K_g^T}{\sqrt{d_k}}\right) V_g. \quad (4)$$

After getting multihead crosstrack attention, in order to make the output sequence and the input sequence have the same dimension, the output sequence is normalized by layer, and then, input the feedforward sublayer to make residual connection with the normalized output sequence, so as to get the output sequence $Z_{p \longrightarrow g}^{[i]}$ after the learning module of layer i. Similarly, take the piano track and learn the bass track information $Z_{p \longrightarrow b}^{[i]}$.

Finally, $Z_{p \longrightarrow g}^{[i]}$ and $Z_{p \longrightarrow b}^{[i]}$ 144 are spliced to obtain the piano sequence Z_p containing the information 145 of guitar sequence and bass sequence, as shown in Equation (5).

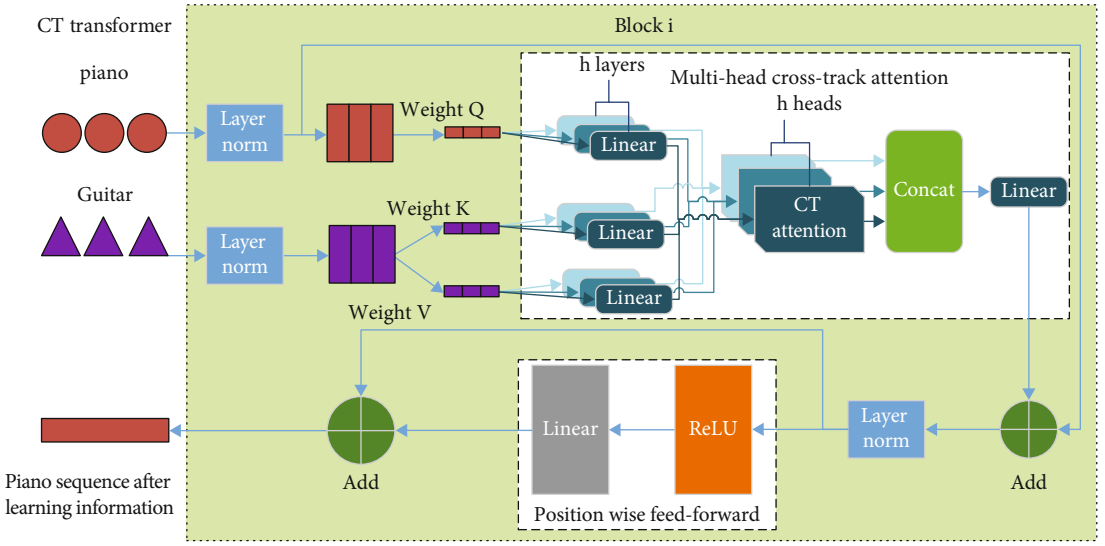


Figure 4: Piano sequence learning guitar sequence module.

Guitar sequence Z_g and bass sequence Z_b are similar to piano sequence Z_p.

$$Z_p = \text{Concat}\left(Z_{p\longrightarrow g}^{[i]}, Z_{p\longrightarrow b}^{[i]}\right). \tag{5}$$

3.4. Discrimination Network. After the corresponding predicted token $\hat{y}_t$ is obtained by generating the network, this paper takes the note vector X_{t+1} input at the next moment as the target value $\hat{y}_{t+1}$ at the current moment; that is, it forms a supervised learning environment and updates the model parameters according to the predicted value and the original sample. In this paper, softmax layer is taken as the output layer, that is, the probability distribution of the output notes, so crossentropy is used to construct the loss function. As shown in Equation (6), when multitrack music is generated, the parameters can be greatly optimized and the quality of music works can be improved through cross-entropy training of the model.

$$D_\varphi = -\frac{1}{I}\sum_{i=1}^{I}[y_t \log \hat{y}_t + (1-y_t)\log(1-\hat{y}_t)]. \tag{6}$$

The emotion represented by each mode in the music is different. When composing music, the composer needs to set the range of notes in a music in advance. Once there are notes higher or lower than the range, the quality and emotion of a music will be greatly reduced. In order to improve the quality of generated music, this paper adds this restriction to the music theory rules. In pop music, the pitches of the piano, guitar, and bass tracks are within the C2-C6, E2-E3, and E2-C4 ranges, respectively. In Equation (7), $y_{\min}$ and $y_{\max}$ are the lowest and highest notes set in advance according to the musical mode, y_t is the pitch at time$t$, and $R^{m_1}(S_{1:t}, y_t)$ represents the reward value of the state at time$t$ (we set different reward values according to the impact on the music itself caused by conforming to this rule and not conforming to this rule. The reward value in subsequent rules is also different. The setting of the value is as follows: first, find the rule that has the greatest impact on the quality of generated music, set its reward value to +1 or -1, and then get its reward value according to the comparison between each rule and this strongest rule).

$$R^{m_1}(S_{1:t}, y_t) = \begin{cases} 0.1, & y_t \in [y_{\min}, y_{\max}], \\ -0.6, & y_t \notin [y_{\min}, y_{\max}]. \end{cases} \tag{7}$$

Furthermore, the number of notes for the piano, guitar, and bass is fewer than 8, 6, and 1, respectively. In as Equation (8), a_t represents the number of notes at time t, n represents the maximum number of notes sounded, and $R^{m_2}(a_t)$ represents the bonus value for the number of notes within the required range.

$$R^{m_2}(a_t) = \begin{cases} 0.2, & a_t \le n, \\ -0.5, & a_t > n. \end{cases} \tag{8}$$

The chords set in this paper are triads, such as F-G-Am-F. For chord notes, in strong music, the strong beat is basically in the odd beat position. Assume that C_1^t, C_2^t, and C_3^t are chord notes at the moment t, and y_t represents the notes selected by the generation network at that moment t, as shown in Equation (9), where $R^c(S_{1:t}, y_t)$ is the reward value provided by this rule (we set the corresponding reward value according to the compliance with this rule, and for subsequent rules, the reward value is determined by the degree of impact on the quality of the generated music).

$$R^C(S_{1:t}, y_t) = \begin{cases} 0.7, & y_t \in \left(C_1^t, C_2^t, C_3^t | t\%2 = 1\right), \\ 1, & y_t \notin \left(C_1^t, C_2^t, C_3^t | t\%2 = 1\right). \end{cases} \tag{9}$$

Different weight ratios are assigned according to the importance of different rules in music, as shown in Equation (10), where R_G represents the reward value for meeting the set rules and α_i represents the weight of rule i, $i = 1, 2, 3$.

$$R_G = \alpha_1 R^{m_1}(S_{1:t}, y_t) + \alpha_2 R^{m_2}(a_t) + \alpha_3 R^C(S_{1:t}, y_t). \quad (10)$$

The objective function of the model is obtained by assigning different weights to the reward function and the crossentropy loss function through the discrimination network, as shown in Equation (11), where J_{G_θ} is the objective function value after assigning weight values β_1 and β_2 to the reward value and the loss function value.

$$J_{G_\theta} = \beta_1 R_G + \beta_2 D_\varphi. \quad (11)$$

The Adam optimizer used in the experiment sets the learning rate as 0.0001. The number of iterations set in the training process is 20000. If the number of iterations of training is less than 20000, the value of loss function converges, and the model training is terminated immediately. If the number of iterations of training has reached 20000, but the loss function value has not converged, the model training will end automatically. In the process of model training, when the model obtains the reward value of identifying network feedback at each step, it will automatically update the network parameters, so as to maximize the long-term reward of the objective function. This section uses the objective function constructed by Equation (11) to update the gradient of the generated network parameter θ of the model, as shown in Equation (12). The network parameters θ of the generated network can be optimized according to Equation (13). After the model is trained, the test set of 500 MIDI files is used to test the model. The event of one section at the beginning of each MIDI file is used as the input, and the predicted section event is used as the input of the next round of prediction, and the model is continued in turn.

$$\nabla_\theta J_{G_\theta} = \sum_{t=1}^{T} G_\theta(y_t | S_{1:t}) * R_G(S_{1:t}, y_t), \quad (12)$$

$$\theta \longleftarrow \theta + \nabla_\theta J_{G_\theta}. \quad (13)$$

4. Experiment and Analysis

4.1. Data Set and Implementation Details. All experiments in this paper used the Lakh MIDI data set [18], which included 176,581 different multitrack MIDI files. The music containing piano, guitar, and bass was screened out from the data set, and then, these three tracks were extracted through the pretty MIDI library and combined to obtain 55,213 MIDI files. Finally, the MIDI files of 4/4 beats are selected. At this point, the data set contains only 34,610 MIDI files. We used 24,610 MIDI files as the training set and 10,000 MIDI files as the test set. The generation and discrimination networks in the proposed model were trained using an Adam optimizer and a reward network, to minimize the crossentropy level and optimize the output. The learning rate ε was set to 0.0002, and the number of iterations was 10000. This section tests the MuseGAN model, the MultINN network, and MTMG using our data set and compares the results generated by our proposed network with the same character length generated by all four methods.

Table 1: Human vs. AI evaluation results.

	Average (pro)	Average (all)
Human	8.02	7.93
AI	7.83	8.11

Table 2: Score results of four models.

Indicators	Our	MuseGAN	MTMG	HRNN
Rhythm	8.05	5.79	6.36	5.33
Melody	8.63	7.75	8.21	6.47
Emotion	7.88	8.36	7.29	6.36
Harmony	8.22	5.96	7.48	4.50

Table 3: To quantitatively compare the different modes of multitrack music generation.

Method	Chord matching	Harmony	BLEU
Our	0.681	0.765	0.660
MTMG	0.527	0.731	0.523
MuseGAN	0.579	0.699	0.641
HRNN	0.363	0.525	0.592

4.2. Analysis of Experimental Results

(1) Subjective evaluations: we divided all participants into two groups, professional composers and noncomposers. Participants in the professional groups are those with degrees in music creation or electronic music creation and production education, including the Central Conservatory of Music, Communication University of China, and Zhengzhou University

Human and AI. We prepared a mix of five pieces of music by professional human composers and five pieces created by our model for people to decide whether they were created by humans or by AI [19]. Forty professional composers were asked to rate each piece of music they heard in terms of musical creation theory, while 60 noncomposers were asked to rate their subjective feelings. Each listener will evaluate and score the test samples (points 1 to 10).

Among professional composers, the average score for human music was higher than AI (see Table 1). However, across all participants, our AI music scored higher than real human works (8.11 vs. 7.93), indicating that the quality of our AI music creation was quite close to that of human composers. According to a few single ratings, there are even works that transcend 8 human work. Interestingly, for all

Table 4: Comparison of music theory characteristics.

Method	Ours	MTMG	MuseGAN	HRNN	Ours (without music rules)
Note repetition	19.4%	52.6%	0.660	25.4%	35.1%
The notes are out of tune	5.1%	13.4%	0.523	7.2%	8.7%
Unique maximum note	54.6%	47.6%	0.641	51.7%	52.3%
Unique minimum note	57.1%	47.9%	0.592	48.3%	59.2%

the reviewers, the music of the human composers was considered to have been produced by artificial intelligence.

Contrast experiment. Our second test was to compare the generated samples with the three baseline models. We did a hearing test comparing MTMG, MuseGAN, and HRNN. Participants (15 composers and 30 noncomposers) received 20 pieces of music from four different models, each of which generated five pieces of music, but were given the same starting notes and instrumental timbers. The participants were then asked to rate and rate the music in terms of melody, harmony, rhythm, and emotion. After summarizing the scores, the participants were given another round of assessment, which was repeated three times in turn to summarize the final score.

Compared with the three generation models of MuseGAN, MTMG, and HRNN, the overall quality of our model has been significantly improved (see Table 2). Except for emotion, the scores of other indicators are significantly higher than those of the other three models, indicating that the music we generate is more in line with the requirements and rules of composition. It reflects the need to strengthen research on emotion in future work.

(2) Objective evaluations: in this paper, a test set containing 500 MIDI files is used to analyze and evaluate our model, MuseGAN, MTMG, and HRNN models from three aspects of harmony degree, chord accuracy, and BLEU score (the BLEU score is used to measure the similarity between the test set and the generated sample [20]). The model parameters and training set are the same. In order to test whether the model proposed in this paper improves chord accuracy, chord accuracy is defined to evaluate the accuracy between the chords of the samples generated by the model and the specified chords, as shown in Equation (12), where P is the number of segments, $\tilde{y}_m$ is the m chord detected in the generated melody, y_m is the corresponding m chord in the given chord progression, and $E(y_m, \tilde{y}_m)$ represents the error value between $\tilde{y}_m$ and y_m:

$$\text{Chord matching} = \frac{\sum_{m=1}^{P} E(y_m, \tilde{y}_m)}{P}, \quad E(y_m, \tilde{y}_m) = \begin{cases} 1, & \tilde{y}_m = y_m, \\ 0, & \tilde{y}_m \neq y_m \end{cases} \tag{14}$$

The harmony of music is the basic standard to evaluate the quality of music, so it is meaningful to evaluate the harmony degree. We also analyzed the harmony degree of the samples generated by the four models and defined that the two tracks have similar chord progression; that is, the two tracks are harmonious, as shown in Equation (15), where P and K, respectively, represent the number of segments generated by music and the number of instrument tracks and C_P^K is the chord corresponding to the P section of the K instrument track.

$$\text{Harmony} = \frac{\sum_{p=1}^{P} \delta \cap {}_1^k C_P^K}{P}, \tag{15}$$

$$\delta(a) = \begin{cases} 1, & a \neq \varnothing, \\ 0, & a = \varnothing. \end{cases} \tag{16}$$

The model we proposed is higher than the other three models in terms of chord matching degree, harmony degree, and BLEU, indicating that the introduction of rules into the discrimination network can guide the generation of music to a certain extent, and CT-transformer can also be of great help in learning information between different tracks (see Table 3).

Comparison of music theory characteristics: in order to verify whether the music rules and their rewards and punishments in our model play a guiding role in music generation, this section quantifies their expression forms according to the set music rules and carries out a series of comparative experiments on the music theory rules. This experiment compares 500 music works generated by three models: MTMG, MuseGAN, HRNN, and our model without music theory rules. In order to ensure the fairness of the experiment, the parameters of the three models, the number of bars, and starting notes of the generated music are the same when generating samples. Select a series of effective feature information from the music samples of the above three models, compare the music theory rules, and summarize the specific statistics, as shown in Table 4.

It can be seen from Table 4 that our model effectively reduces the repetition of notes compared with the music works generated by other models. Compared with the model without music theory rules, the complete model has certain advantages in many indicators, which also shows that the music theory rules after mathematical modeling play a certain guiding role in generating music.

5. Conclusions

In this paper, we propose a novelty model for multitrack music generation. It combines sequence-to-sequence

generation and multitrack learning techniques in a unified framework to achieve optimal convergence of multitrack learning and codecs. In this model, we combine with the discrimination network on the basis of transformer to produce multitrack music in line with the music literacy of the public under the guidance of music rules. The experimental results show that this model has significant advantages over some existing techniques in terms of rhythm, audibility, fluency, and compliance with music rules. In the future, we will strengthen the research on emotion and more than three tracks.

Data Availability

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

Conflicts of Interest

There is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work is supported by Key Projects of the National Key Research and Development Program of China (2018YFB14039002), National Natural Science Foundation of China (61631016), Beijing Outstanding Young Scientist Program (BJJWZYJH01201910048035), and Fundamental Research Funds for the Central Universities (CUC210B011).

References

[1] H. W. Dong, W. Y. Hsiao, L. C. Yang, and Y. H. Yang, "MuseGAN: multi-track sequential generative adversarial networks for symbolic music generation and accompaniment," *Thirty-Second AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.

[2] C. Jin, Y. Tie, Y. Bai, X. Lv, and S. Liu, "A style-specific music composition neural network," *Neural Processing Letters*, vol. 52, no. 3, pp. 1893–1912, 2020.

[3] S. Mehri, K. Kumar, I. Gulrajani et al., "SampleRNN: an unconditional end-to-end neural audio 291 generation model," 2016, http://arXiv.1612.07837.

[4] T. Wang, J. Liu, C. Jin, J. Li, and S. Ma, "An intelligent music generation based on variational autoencoder," in *2020 International Conference on Culture-oriented Science & Technology (ICCST)*, pp. 394–398, Beijing, China, 2020.

[5] H. Chu, R. Urtasun, and S. Fidler, "Song from PI: a musically plausible network for pop music generation," 2016, http://arXiv.1611.03477.

[6] L. C. Yang, S. Y. Chou, and Y. H. Yang, "MiDiNet: a convolutional generative adversarial network for symbolic-domain music generation," 2017, http://arXiv.1703.10847.

[7] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," *Advances in neural information processing systems*, vol. 30, pp. 5998–6008, 2017.

[8] O. Mogren, "C-RNN-GAN: continuous recurrent neural networks with adversarial training," 2016, http://arXiv.1611.09904.

[9] A. Roberts, J. Engel, C. Raffel, C. Hawthorne, and D. Eck, "A hierarchical latent vector model for learning long-term structure in music," *International Conference on Machine Learning*, vol. 80, pp. 4364–4373, 2018.

[10] M. Dehghani, S. Gouws, O. Vinyals, J. Uszkoreit, and Ł. Kaiser, "Universal transformers," 2018, http://arXiv.1807.03819.

[11] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. V. Le, and R. Salakhutdinov, "Transformer-XL: attentive language models beyond a fixed-length context," 2019, http://arXiv.1901.02860.

[12] C. Z. A. Huang, A. Vaswani, J. Uszkoreit et al., "Music transformer: generating music with long-term structure," 2018, http://arXiv.1809.04281.

[13] M. Akbari and J. Liang, "Semi-recurrent CNN-based VAE-GAN for sequential data generation," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 2321–2325, Calgary, AB, Canada, 2018.

[14] C. Donahue, H. H. Mao, Y. E. Li, G. W. Cottrell, and J. McAuley, "LakhNES: improving multi-instrumental music generation with crossdomain pre-training," 2019, http://arXiv.1907.04868.

[15] N. Zhang, *Learning adversarial transformer for symbolic music generation*, IEEE Transactions on Neural Networks and Learning Systems, Piscataway, NJ, USA, 2020.

[16] C. Jin, T. Wang, S. Liu et al., "A transformer-based model for multi-track music generation," *International Journal of Multimedia Data Engineering and Management (IJMDEM)*, vol. 11, no. 3, pp. 36–54, 2020.

[17] S. Ji, J. Luo, and X. Yang, "A comprehensive survey on deep music generation: multi-level representations, algorithms, evaluations, and future directions," 2020, http://arXiv.2011.06801.

[18] C. Raffel, *Learning-based methods for comparing sequences, with applications to audio-to-midi alignment and matching [Ph.D. thesis]*, Columbia University, Ann Arbor, MI, USA, 2016.

[19] X. Wu, C. Wang, and Q. Lei, "Transformer-XL based music generation with multiple sequences of7 time-valued notes," 2020, http://arXiv.2007.07244.

[20] D. Jeong, T. Kwon, Y. Kim, and J. Nam, "Graph neural network for music score data and modeling expressive piano performance," in *Proceedings of the 36th International Conference on Machine Learning*, pp. 3060–3070, Long Beach, California, USA, 2019.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 7086599, 17 pages
https://doi.org/10.1155/2022/7086599

Research Article

Objective Evaluation Method of Broadcasting Vocal Timbre Based on Feature Selection

Chaohui Lv[1,2] **Hua Lan**[1] **Ying Yu**[1] **and Shengnan Li**[1]

[1]*School of Information and Communication Engineering, Communication University of China, Beijing 100024, China*
[2]*Key Laboratory of Acoustic Visual Technology and Intelligent Control System, Ministry of Culture and Tourism, Beijing 100024, China*

Correspondence should be addressed to Chaohui Lv; llvch@cuc.edu.cn

Received 18 September 2021; Revised 9 January 2022; Accepted 7 May 2022; Published 26 May 2022

Academic Editor: Ghufran Ahmed

Copyright © 2022 Chaohui Lv 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.

Broadcasting voice is used to convey ideas and emotions. In the selection process of broadcasting and hosting professionals, the vocal timbre is an important index. The subjective evaluation method is widely used, but the selection results have certain subjectivity and uncertainty. In this paper, an objective evaluation method of broadcasting vocal timbre is proposed. Firstly, the broadcasting vocal timbre database is constructed on Chinese phonetic characteristics. Then, the timbre feature selection strategy is presented based on human vocal mechanism, and the broadcast timbre characteristics are divided into three categories, which include source parameters, vocal tract parameters, and human hearing parameters. Finally, the three models of hidden Markov model (HMM), Gaussian Mixture Model-General Background Model (GMM-UBM), and long short-term memory (LSTM) are exploited to evaluate the timbre of the broadcast by extracting timbre features and four timbre feature combinations. The experiments show that the selection of timbre features is scientific and effective. Moreover, the accuracy of the LSTM network using the deep learning algorithm in the objective evaluation of the broadcast timbre is better than the traditional HMM and GMM-UBM, and the proposed method can achieve about 95% accuracy rate in our database.

1. Introduction

Speech recognition and synthesis have made breakthroughs in recent years, and more and more attention has been paid to the speech timbre research. However, the objective evaluation of broadcasting vocal timbre belongs to the specialist areas. The vocal timbre database can be divided into three categories, which include reading language, fluent language, and spontaneous language. Some studies have shown that broadcasting knowledge can be used in the database construction. New rhythmic features are put forward to the analysis of relevant discussions in broadcasting, such as vocal expression features, discourse stress, parataxis, and compound rhythmic phrases, and they are integrated into rhythm and text annotation specifications in database [1]. At present, PRAAT software is commonly used to analyze recorded audio with the aid of a computer [2]. In addition, the relationship between vocal timbre and emotional perception in words is also researched, and the results show vocal timbre can convey the emotional meaning [3].

From the results of speech feature extractions, the evolution of audio features can be divided into time-domain, frequency-domain, time-frequency-domain joint features and depth features. Since the end of 1950s, time-domain features have played an important role in audio analyses and classifications. Frequency-domain features appeared in order to analyze the frequency spectrum of audio signals, such as tone and formant. By the end of 1960s, joint time-frequency feature extraction algorithm appeared. With the development of deep learning, depth features began to be widely used in audio signal processing algorithms [4]. Since 2010, depth features have been applied to speech enhancement, separating speech and music audio signals, and automated audio classification [5–7].

From the perspective of the speech recognition model, the application of speech signal can be roughly divided into three categories, including vocal print recognition, speech recognition, and emotion recognition [8]. The classifiers for speech recognition tasks include traditional classifiers and deep learning algorithms, involving HMM, Gaussian Mixture Model (GMM), support vector machine (SVM), and extreme learning machine (ELM) [9–11]. At present, the role of acoustic parameters is analyzed in the objective evaluation of artistic vocal, and methods of it are proposed based on error back propagation (BP) and learning vector quantization (LVQ) [12]. Piano sound is also studied on keys influencing and its quality evaluation. And a double neural network composed of Hopfield and general regression neural network (GRNN) is implemented to evaluate the piano audio. Signals are processed and weight ratios of the output are calculated by the least square method [13].

Figure 1 shows the basic framework of objective evaluation of vocal timbre proposed in this paper. The main three steps are as follows:

(1) Construction of broadcasting vocal timbre database: this paper focuses on selections of vowels and auxiliary sounds, voiceless, and voiced, as well as words and sentences which can reflect the difference from the broadcasting vocal timbre and daily spoken. Software PRAAT is used to mark the phonetic layer, phonological layer, and prosodic layer

(2) Feature optimization based on the selection strategy of timbre features considering the mechanism of human vocalization: the selected feature parameters are required to reflect the human vocal mechanism as far as possible. Therefore, human vocal timbre features are classified by the feature selection module combining principal component analysis (PCA), and three feature categories are obtained including source parameters, vocal tract parameters, and human hearing parameters. To remove the correlations between features, PCA is exploited to reduce the dimension of the above three feature categories, and finally, 74-dimensional timbre feature parameters are obtained [14, 15]

(3) Timbre grading by the objective evaluation method: HMM [16], GMM-UBM, LSTM, and other models are used to compare different objective evaluation methods [17–20]

The main contributions of this paper are three parts. (1) Based on the broadcasting knowledge and Chinese phonetic characteristics, a database of broadcasting vocal timbre was constructed and annotated. (2) Combining the mechanism of human vocalization and PCA technology, we screened out the characteristic parameters of broadcast timbre and divided them into three categories which include source parameters, vocal tract parameters, and human hearing parameters. (3) We conducted fusion and nonfusion experiments on the three types of timbre features, respectively. Among them, we used four timbre feature combinations for fusion and conducted comparative research on objective evaluation of broadcast timbre on models such as HMM, GMM-UBM, and LSTM.

2. Related Work

Language is an important communication tool between people, and artistic language pays more attention to the expressiveness of language. Announcers employ the vocal organs to send out voices with personal characteristics, and listeners get different perceptions through the auditory organs and then achieve the purpose of conveying thoughts and spreading emotions. Traditional broadcast vocal timbre evaluation relies on manual evaluation, which is susceptible to the personal preferences, professional qualities, and life experience of the participants, and has greater subjectivity and uncertainty. In this context, this paper is devoted to develop a set of scientific and complete objective evaluation system of broadcasting vocal timbre.

The construction of the speech database has been booming in recent years, and various speech databases for different research needs have appeared. Due to different environment and background noises, Bang et al. directly captured broadcast data from broadcast channels, including news, current affairs, culture, documentaries, dramas, children, and entertainment, and manually constructed an English phonetic database of 7 types of evaluation data [21]. He et al. built CUCBNC, a Chinese language broadcast news speech database. By analyzing the related discourses of broadcasting, they proposed new prosodic features, including voice expression features, text stress, paraphrase groups, compound prosodic phrases, and integrated these prosodic features into the database [1]. Not the same as research in [1, 21], the data in this paper came from a clean environment, there was only one speaker with a clear voice, and the noise was small. In addition, the voice data was based on the theoretical knowledge of broadcasting and human body acoustics, focusing on selecting vowels, consonants, unvoiced, and voiced sounds that can reflect the tone of the broadcast voice, and the three-level evaluation was conducted by judges with professional qualities in the field of broadcast hosting and used PRAAT software to mark the pinyin layer, phonological layer, and prosodic layer of the speech data.

In the field of speech, different features are selected according to different research tasks. Al-Qaderi et al. used prosodic features and short-term spectral features, combined with GMM and SVM to achieve speech recognition in human-computer interaction [22]. Al-Kaltakchi et al. utilized four feature combinations based on Mel frequency cepstral coefficient (MFCC) and power normalized cepstral coefficient (PNCC), as well as seven fusion methods, combined with *i*-vector and ELM to identify the speaker [23]. The above research can find that it is not that the more features are selected, the higher the accuracy rate is. The characteristic parameters selected in this paper required that the characterization of timbre characteristics was as complete as possible and could accurately reflect the

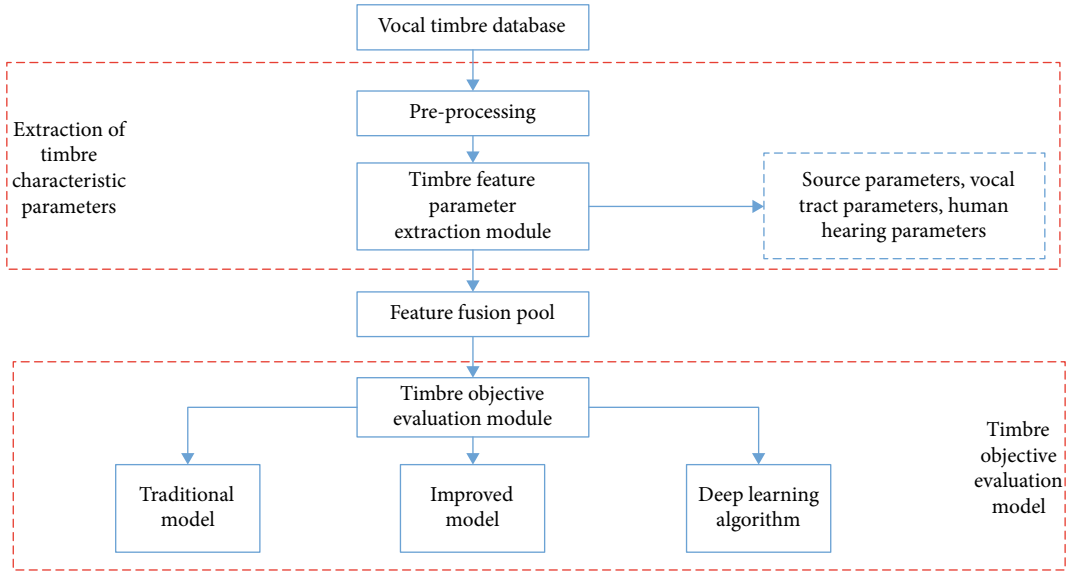


Figure 1: Basic framework of objective evaluation of vocal timbre.

mechanism of human vocalization. Therefore, from dozens of pairs of characteristic parameters, PCA and the principle of human vocalization were used for features selection, and the source parameters, vocal tract parameters, and human hearing parameters were obtained after dimensionality reduction.

In addition to extracting discriminative features from the speech signal, the choice of the classification model is also crucial for the speech recognition system. Al-Kaltakchi et al. proposed the simultaneous use of PNCC and MFCC, together with acoustic modeling using a GMM-UBM, and realized speech recognition under different signal to noise ratios [24]. Furthermore, deep learning networks have been widely developed in the field of speech recognition. Jahangir et al. employed a deep neural network (DNN), taking MFCC and time domain features as input to build a speech recognition model. DNN obtained better classification results compared with five machine learning algorithms that were recently utilized in speaker recognition [25]. Same as the speech recognition system, in the process of constructing the objective evaluation system of the broadcast timbre, it is not only necessary to extract the features of the timbre characteristics but also to select an appropriate classification model. The broadcast speech signal is a complex time-varying signal. When inputting the speech signal into the traditional shallow acoustic model, only a few levels of linear or nonlinear transformation are usually carried out. In this paper, in addition to the typical HMM and GMM-UBM models, the LSTM model is also utilized as an objective evaluation model. LSTM is a variant of recurrent neural network (RNN). By adding a gating mechanism to control the information transmission in the recurrent neural network, it can handle complex time-varying signals and has stronger modeling capabilities for the nonlinear dynamic process of human speech.

3. Construction of the Broadcasting Vocal Database

The speech samples were selected from the raw data of Communication University of China (CUC). These speech data were collected during preliminary and reexamination of broadcasting and hosting major from 2012 to 2016. The material format is WMV, and the recording environment is professional recording studio. Besides Adobe software system, the hardware equipment mainly includes computer Mac Pro for data acquisition and processing, Sony ecm-nv1 super directional condenser microphone. The sampling frequency is 44.1 KHz, and the quantization bits are 16 bit. In the selected 6000 speech samples, the speaker ratio of male and female students is 1 : 1, and short articles and newsletters were selected covering political hot spots, science, and technology culture. The speakers read in the same environment. Finally, teachers made a three-level evaluation on the thickness of vocal timbre to form the initial broadcasting vocal timbre database. And all the teachers were faculty of the school of broadcasting and hosting arts of CUC.

The initial vocal timbre database was further processed to remove the uncertainty influence and improve the classification accuracy. The construction of the broadcasting vocal timbre database is mainly divided into the following five steps:

(1) Format conversion: use ProCoder software to extract audio information from original video files and save them as lossless compressed wave audio file format

(2) Middle section interception: middle section of each wave file was intercepted by MATLAB program. And each audio file was saved as 20 seconds to meet the actual requirements of human hearing in subjective evaluation

(3) Noise removal: if the speech sample is processed by noise reduction in the later stage, it will cause distortion, and part of the timbre characteristics will be lost in the feature extraction process. Therefore, the human ear recognition method is used to remove the speech samples with noises generated by irrelevant factors

(4) Phoneme coverage improvement: referring to " Mandarin and Broadcasting Pronunciation," the speech samples are selected including all vowels and vocal tones as far as possible by counting the coverage of each phoneme in speech samples

(5) Pitch marking: the phonological layer and the prosodic layer of the speech samples are labeled by PRAAT software. In addition, it is necessary for professionals to confirm the label for each selected speech sample. Thin and thickness were selected to focus on each pair of evaluation terms. Other evaluation terms, such as brightness, tightness, and other evaluation indicators, need only a small number of tags from evaluators to reconfirm and replace in the later stage. Table 1 lists the subjective evaluation terms and objective parameters of the vocal quality broadcasted by electronic equipment, which also has a strong reference significance for the subjective evaluation of broadcasting vocal timbre

Through the above steps of the sample selection and processing, a dataset is completed with unified format, replaceable tag, wave format, and 20 seconds time length. Each file size is 1.68 M with 44.1 KHz sampling frequency, and the quantization bit is 16 bit. According to the thickness of timbre, the dataset is divided into three levels, *A*, *B*, and *C*. Each level contains 100 speech samples of male and female students, and the total dataset has 600 samples. Figure 2 shows the constitution of the broadcasting vocal timbre database.

4. Objective Evaluation Models of Broadcast Vocal Timbre

4.1. HMM. The hidden Markov model is used to describe a Markov process with hidden unknown parameters. It is a double random process without memory. Memoryless means that the system is in the current state, and the future state has nothing to do with the past state. One of the double random processes refers to the implicit random process that uses a Markov chain with a finite number of states to simulate changes in the statistical characteristics of the voice signal, and the other refers to the random process of the observation sequence associated with each state of the Markov chain. In the process of normal human pronunciation, such a double random process will occur, because the speech signal can be regarded as an observable sequence, and the other unobservable sequence is the parameters of the words, sentences, and segments generated in the brain. The HMM model can just simulate such a process. On the one hand, it can characterize speech signals with different characteristic parameters; on the other hand, it can also characterize the conversion process between signals well.

When HMM is applied to the objective evaluation system of timbre, the following three problems need to be solved, and corresponding algorithms are proposed for each problem:

(1) Valuation problem: given model parameter λ and observation sequence O, calculate the maximum probability $P(O|\lambda)$ of the observation sequence O under model λ. To solve this problem, we need to use a forward-backward algorithm to calculate the probability of each HMM producing a given observation sequence O and then evaluate the optimal HMM model

(2) Decoding problem: given model parameters λ and observation sequence O, find the optimal hidden state sequence. To solve this problem, the Viterbi algorithm needs to be used to decode the hidden state sequence. The Viterbi algorithm is a general decoding algorithm based on dynamic programming and the method of finding the shortest path of the sequence

(3) Learning problems: given the observation sequence O, the model parameters are unknown, how to maximize the output probability of the observation sequence by adjusting the model parameters. This paper utilizes the maximum expectation (EM) algorithm to solve; using the recursive idea, the local maximum value is obtained after continuous iterative optimization model parameters

4.2. GMM-UBM. GMM can approximate the distribution of any data by weighting multiple Gaussian probability density functions. But in the speech recognition training process, each target speaker needs to be modeled, and then the GMM models of all the trained speakers are found and tested. GMM has a strong characterization ability for actual data, but the stronger the characterization ability is, the larger the parameter scale is, and the more data is needed to drive to obtain a more generalized general model. GMM-UBM introduces many nontarget speakers' voice and uses the probability model of the spatial distribution of these nontarget speakers' voice features to train a GMM. This GMM does not have the identity of a specific target speaker. It is a priori model of the target speaker model, and then the target speaker's speech parameters are fine-tuned on this model. With the advantage of UBM, only a small amount of correction is needed when training each target speaker model individually. The model obtained in this way not only greatly improves the recognition rate but also reduces the amount of calculation.

Before training GMM-UBM, we need to initialize the model and set the initial parameters of the model. For the determination of parameter $\lambda = \{\mu, \delta, \alpha\}$, the K-means algorithm is usually used to roughly estimate the range of the parameter. α is the weighting coefficient of the model. μ is the expected value of the model. δ is the variance of the

Table 1: Subjective evaluation terms and objective parameters of vocal quality.

Serial number	Subjective evaluation terms	Description of objective parameters
1	Wide	Advantages of wide frequency band, small distortion, large dynamic range, uniform frequency distribution, and outstanding energy in medium and low frequency band
2	Narrow	Narrow frequency band, lack of high and low frequency, medium frequency too prominent, short reverberation time
3	Bright	Moderate medium and low frequency, enough high frequency energy, small distortion, and rich harmonic decay process
4	Dark	Lack of medium and high frequency, 5-6 KHz has obvious attenuation and short reverberation time
5	Thin	Lack of medium and low frequency, 300-500 Hz has obvious attenuation, and the sound energy is insufficient
6	Thick	Medium and low frequency energy is strong, low frequency is full, high frequency is moderate, distortion is small, and reverberation is moderate
7	Soft	Low and medium frequency response range is wide, and distortion is small
8	Hard	Lack of low frequency, medium and high frequency, and high frequency harmonic decay too fast
9	Thorough	Advantages of small distortion, good transient response, enough medium and high frequency reverberation, wide and uniform frequency response
10	Paste	Adequate low frequencies, long reverberation time, and lack of medium and high frequencies
11	Pine	High frequency, low frequency, good transient response
12	Real	The medium and low frequency energy is enough, the high frequency is moderate, the distortion is small, and the loudness is high
13	Crude	Low frequency energy is large, and medium and high frequency energy is small
14	Fine	Lack of low frequency, moderate medium and high frequency, insufficient reverberation, and low loudness
15	Tip	Lack of low frequency, medium and high frequency, high frequency energy is too large, and distortion is large
16	Circular	Wide frequency band, minimal distortion, good transient response, and moderate reverberation
17	Hair	Adequate high frequency and medium high frequency components, and the distortion is large
18	Sand	The passband distortion is large and accompanied by transient distortion
19	Stuffy	Lack of medium and high frequency, serious attenuation above 3-4 KHz, and excessive low frequency energy
20	Brittle	Lack of low frequency, medium and high frequency partial, frequency response is uneven
21	Clear	Wide and uniform frequency response, rich medium and high frequency, moderate reverberation
22	Muddy	Lack of medium and high frequency, excessive reverberation, and poor transient response
23	Plump	Rich in medium and low frequencies, moderate in high frequency, and appropriate in loudness
24	Shriveled	Lack of medium and low frequency reverberation, reverberation aftersound is too short
25	Clean	Small distortion, good transient response, and moderate reverberation
26	Three-dimensional	Wide frequency response, small distortion, moderate reverberation, and large dynamic range

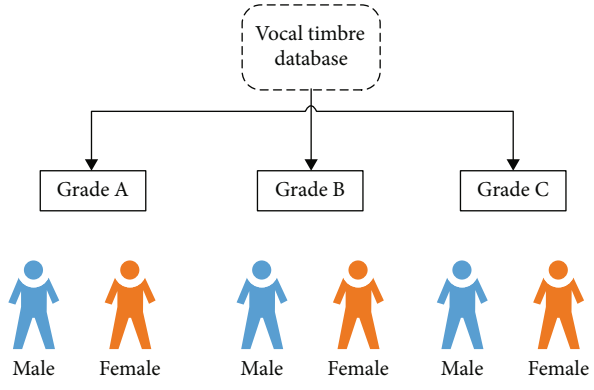


Figure 2: Broadcasting vocal timbre database.

model. After initializing the parameters, the EM algorithm is used to iteratively generate local optimal model parameters. According to the model's initial parameter λ, we train to estimate the new model parameter $\bar{\lambda}$, so that $P(X|\bar{\lambda}) \geq P(X|\lambda)$. Then, train and estimate with the new model parameter $\bar{\lambda}$ until the model meets the convergence condition.

After using the EM algorithm to train and estimate the UBM model that uses the vocal features of the nontarget speaker as background data, it is only necessary to adaptively match the trained model parameters to the GMM model of the target speaker to match the target speaker's model. The parameters are fine-tuned on the original UBM model without the need to perform GMM modeling for each speaker. The Gaussian distribution obtained by

training is fitted to the target speaker's data by using the maximum posterior probability (MAP) algorithm. The offset probability relative to the i-th Gaussian distribution obtained by UBM training is

$$P(i|x_i) = \frac{\alpha_i p_i(x_t)}{\sum_{j=1}^{M} \alpha_j p_j(x_t)}. \quad (1)$$

The characteristic parameter sequence of the target speaker is $X = \{x_1, x_2, \cdots, x_N\}$, and M is the order of the model.

After the target speaker model is obtained through adaptive training, the method of calculating the log-likelihood ratio is used to complete the recognition task by calculating the probability score of the speech to be tested on the GMM and UBM models. The log-likelihood ratio calculation equation can be defined as

$$S(X) = \log \frac{P(X|\lambda_{\text{GMM}})}{P(X|\lambda_{\text{UBM}})} = \log P(X|\lambda_{\text{GMM}}) - \log P(X|\lambda_{\text{UBM}}), \quad (2)$$

where λ_{GMM} is the target speaker GMM model parameter, and λ_{UBM} is the target speaker UBM model parameter.

4.3. LSTM. The speech signal is a complex time-varying signal. When the speech signal is input into the traditional shallow acoustic model, it usually undergoes only a few levels of linear or nonlinear transformation. Such shallow models currently have relatively mature structures and optimization algorithms. For more complex signals in speech tasks, by introducing deep models with stronger representation capabilities, the structure of the deep models is more complex and requires more levels of nonlinear transformations and has more advantages in expressing the nonlinear dynamic process of human speech. Strong modeling ability is also more suitable for processing complex time-varying signals; so, this article introduces a neural network structure LSTM with time "memory" function. LSTM controls the transmission of information in the network through a gating mechanism, which makes up for the defects of gradient disappearance and gradient explosion in the actual application of RNN, and has better performance in long-sequence speech tasks.

The gating mechanism of LSTM consists of forget gate, input gate, output gate, and memory unit. f, i, o, C represent the state of forget gate, input gate, output gate, and cell memory unit, respectively. The forget gate can determine which historical information is forgotten or kept, and the output of the forget gate is expressed as

$$f_t = \sigma\left(W_f x_t + U_f s_{t-1} + b_f\right) f_t = \sigma\left(W_f x_t + U_f s_{t-1} + b_f\right), \quad (3)$$

where W is the weight parameter from the input layer to the hidden layer, U is the weight parameter from the hidden layer to the hidden layer, b is the bias parameter, and σ is the sigmoid function. At time t, the information of the hidden state at the previous time and the current input information are simultaneously input into the sigmoid function, and the output value is between 0 and 1. The closer to 0 means the more it should be forgotten, and the closer to 1 means the more it should be reserved.

The input gate is used to update the state of the memory unit. One part is the sigmoid layer, and the output value from 0 to 1 is used to determine which information to update. The equation is

$$i_t = \sigma(W_i x_t + U_i s_{t-1} + b_i). \quad (4)$$

The other part is the tanh layer, which uses output values from -1 to 1 to generate candidate update content. The equation is as follows:

$$C_t = \tanh\left(W_c x_t + U_c s_{t-1} + b_c\right). \quad (5)$$

Then, multiply the tanh output and the sigmoid output. The sigmoid output will determine which historical information in the tanh output is important and preserve it.

The output gate is used to determine which part of the information of the memory cell state is to be output. The output gate can determine the value of the next hidden state, which contains the relevant information of the previous input. The equation is as (6).

$$o_t = \sigma(W_o x_t + U_o s_{t-1} + b_o). \quad (6)$$

Before passing the output gate, the memory unit must pass the newly obtained state to the tanh function for processing and pass the new implicit state to the next time step. The equation is as follows:

$$s_t = o_t * \tanh\left(C_t\right). \quad (7)$$

5. Feature Selection and Parameter Extraction Based on Human Vocal Mechanism

Before extracting vocal timbre features, it was necessary to preprocess the vocal samples in the database in order to reduce the effects of aliasing and higher harmonic distortion caused by the human vocal organ and the equipment that collected the vocal signal.

5.1. Preprocessing of Speech Signal. The preprocessing process mainly includes preemphasis, framing, windowing, and endpoint detection.

5.1.1. Preemphasis. In the process of vocalization, there are many reasons for the weakening of the signal. Affected by glottal excitation and lip radiation, the high frequency part of the vocal signal frequency above 800 Hz will be greatly attenuated by 6 dB/octave, and the higher the frequency, the attenuation will be the more serious it is. In order to weaken this phenomenon, it is necessary to carry out preemphasis processing on the speech signal, so that the transition of the signal from the low frequency part to the high frequency part is more stable. This article adopts finite impulse

response (FIR) high-pass filter to realize its transfer function is defined as

$$H(z) = 1 - \mu z^{-1}, \tag{8}$$

where μ is the preemphasis coefficient, and the value range is $0.9 < \mu < 1.0$; in this experiment, $\mu = 0.96$.

5.1.2. Framing and Windowing. The speech signal is a time-varying signal, but due to the inertial movement of the vocal organs, within a very short time of 10-30 ms, the spectral and physical characteristics of the speech signal can be approximately considered to be stable. In this article, the method of using frame division and windowing was used in order to obtain a short-term stable speech signal, by using a window function to smoothly slide the speech signal, and cutting the speech signal by frame. The window length is the frame length. In order to ensure the continuity between the signal frame and the frame, the continuous overlapping frame method was used, and the overlap between the two frames before and after is the frame shift, which is generally half of the window length. The frame length set in this experiment is $M = 256$, and the frame shift is 128. The window function is defined as

$$\omega(n) = \begin{cases} 0.54 - 0.46 \cos\left[\dfrac{2\pi n}{(M-1)}\right], & 0 \le n \le M-1 \\ 0, & \text{other} \end{cases}. \tag{9}$$

5.1.3. Endpoint Detection. During the recording of voice samples, few pauses and adjustments are introduced when reading news articles. Therefore, there are some silent segments and background noise segments in the vocal data. This paper used a dual-threshold endpoint detection algorithm to identify and eliminate. The principle is as follows.

Assuming that the n-th frame of speech signal is $x_n(m)$, its short-term average energy is as equation (10).

$$E_n = \sum_{m=0}^{M-1} \left| x_n^2(m) \right|. \tag{10}$$

Short-term zero crossing rate is

$$z_n = \frac{1}{2} \sum_{m=0}^{M-1} \left| \operatorname{sgn}\left[x_n(m+1) - \operatorname{sgn}\left[x_n(m)\right]\right]\right|, \tag{11}$$

where M is the frame length, and $\operatorname{sgn}[x]$ is the sign function.

When the signal-to-noise ratio of the vocal sample is large, the silent and voiced segments of the voice data can be distinguished by calculating the short-term average energy. When the signal-to-noise ratio does not meet the detection requirements, it is necessary to further calculate the average zero-crossing rate. When detecting a silent segment, a threshold is set. When the short-term energy of a frame is less than this threshold, it is judged as a silent frame; otherwise, it is a sound frame.

5.2. Parameter Extraction Based on Human Vocal Mechanism

5.2.1. Source Parameters. This paper selected the pitch frequency that can reflect the vibration effect of the vocal cord as the source parameter. We used the short-term average amplitude difference function.

$$F_n(k) = \frac{1}{M} \sum_{n=0}^{M-k-1} x_i(M+k) - x_i(k), \tag{12}$$

where $x_i(n)$ is the voice signal of the i-th frame, and k is the time delay.

5.2.2. Vocal Tract Parameters. In this paper, we chose the formant frequency and bandwidth as the vocal tract parameters, which can reflect the resonance quality and vocal tract model. Linear prediction cepstrum coefficient (LPCC) is also as the vocal tract parameters.

(1) The Formant Frequency and Bandwidth. The formant refers to the resonance frequency of the acoustic cavity. The number of formants reflects the fullness of the voice, and the position and bandwidth of the formants reflect the brightness and clarity of the voice. In this paper, the cepstrum method was used to extract the frequency and bandwidth of the first, second, and third formants. The specific steps are as follows:

(i) Perform Fourier transform on the i-th frame of speech signal $x_i(n)$. The equation is as follows:

$$X_i(k) = \sum_{n=0}^{N-1} x_i(n) e^{-j2\pi kn/N}. \tag{13}$$

(ii) Find the cepstrum of $X_i(k)$

$$\hat{x}_i(n) = \frac{1}{N} \sum_{n=0}^{N-1} \log_{10} X_i(k) e^{j2\pi kn/N}. \tag{14}$$

(iii) Cepstrum signal windowing: the equation is as follows:

$$h_i(n) = \hat{x}_i(n) \times h(n). \tag{15}$$

(iv) Find the envelope of $h_i(n)$

$$H_i(k) = \sum_{n=0}^{N-1} h_i(n) e^{-j2\pi kn/N}. \quad (16)$$

(v) Find the maximum value on the envelope of the speech spectrum and get the corresponding formant frequency and bandwidth

(2) LPCC. The generation of the speech signal can be regarded as the result of the convolution of the excitation signal and the frequency response of the vocal tract. To obtain the LPCC, it needs deconvolve by the method of homomorphic analysis and perform logarithmic processing. The specific steps for extracting LPCC are as follows:

(i) Perform Discrete Fourier Transform on the preprocessed speech signal $x(n)$. The equation is as follows:

$$X(k) = \sum_{-\infty}^{+\infty} x(n) \cdot \exp\left(-j \cdot \frac{2\pi}{N} \cdot n \cdot k\right). \quad (17)$$

(ii) Take the modulus and logarithm of the signal after discrete Fourier transform

$$C(k) = \ln\left[|X(k)|\right]. \quad (18)$$

(iii) Perform discrete cosine transform (DCT):

$$c(n) = \frac{1}{N} \sum_{k=0}^{N-1} C(k) \cdot \exp\left(j \cdot \frac{2\pi}{N} \cdot k \cdot n\right). \quad (19)$$

The calculation process of LPCC is small. It can not only describe the vowels well but also reflect the personality characteristics of the speech. However, LPCC is linearly approximate to the speech signal, which is contrary to the characteristics of human hearing and affects the performance of the system. So, it needs human hearing parameters as a supplement.

5.2.3. Human Hearing Parameters. To establish a high-performance objective evaluation system of timbre, in addition to study the sounding mechanism related to the timbre of broadcast speech, it is more important to simulate the mathematical model of timbre generation and the complex process of timbre perception based on the auditory characteristics of the human ear. Human hearing parameters mainly consider the subjective perception of human hearing and fully consider the characteristics of human hearing. They have a nonlinear logarithmic relationship with the frequency of the real speech signal, which is used to better characterize the nonlinear perception of human ears at different frequencies. Therefore, this paper selected cochlear frequency cepstrum coefficient (CFCC) and MFCC as the human hearing parameters.

(1) CFCC. The cochlea is an auditory receptor with a complex internal structure. The sensory hair cells can turn sound stimuli into nerve impulses, which are sensed through the auditory center of the brain. In addition, on the basement membrane of the cochlea, where sound peaks of different frequencies appear different, when the frequency is low, the peak value of the basement membrane appears near the cochlear foramen. As the sound frequency increases, the peak value of the basement membrane begins to move to the root, and the frequency of different sounds is logarithmically distributed on the basement membrane. The specific steps of CFCC extraction are as follows:

(i) Cochlear filter: the process of vocal signals passing through the cochlear filter simulates the frequency response of signals of different frequencies on the basilar membrane. The transformation process is as follows:

$$T(a, b) = \int_{-\infty}^{+\infty} f(t) \psi_{a,b}(t) dt, \quad (20)$$

where $f(t)$ is the speech signal function, and $\psi_{a,b}(t)$ is the cochlear filter function.

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \left(\frac{t-b}{a}\right)^{\alpha} \exp\left(-2\pi f_L \beta\left(\frac{t-b}{a}\right)\right) \bullet \cos\left(2\pi f_L \left(\frac{t-b}{a}\right) + \theta\right) u(t). \quad (21)$$

α, β, respectively, refers to the frequency domain shape and width of the cochlear filter function $\alpha = 3, \beta = 0.2$. u(t) is the unit step function. θ is the initial phase, b is the Real numbers that can change over time, and a is the scale variable and determined by the center frequency $f_c f_c$ and the lowest frequency f_L of the cochlear filter bank.

(ii) The transmission mechanism of hair cells is defined as

$$h(a, b) = [T(a, b)]^2; \forall a, b. \quad (22)$$

The hair cell function of the i-th frequency band can be expressed as follows:

$$S(i, j) = \frac{1}{d} \sum_{b=l}^{l+d-1} h(i, b), l = 1, L, 2L, \cdots; \forall i, j, \quad (23)$$

where L is the frame shift, and d is the smoothing window length of the i-th frequency band.

(iii) Convert the output energy value of the hair cell function into perceived loudness through nonlinear cube root compression

$$y(i,j) = S(i,j)^{1/3}. \quad (24)$$

(iv) In order to remove the correlation between features, CFCC parameters are obtained after discrete cosine transform

(2) MFCC. MFCC is proposed based on the characteristics of human hearing. It is based on the linear cosine transform of the logarithmic power spectrum on the nonlinear Mel scale. The extraction steps of MFCC are as follows.

(i) Perform fast Fourier transform on the preprocessed speech signal $x(n)$ and obtain the discrete power spectrum $S(n)$ in the frequency domain. The obtained power spectrum is passed through a triangular Mel filter bank, so that the linear speech signal spectrum is converted into a Mel spectrum that can reflect the characteristics of human hearing. The output after passing $M = 126$ filter banks is P_m

(ii) Calculate the logarithmic energy of each filter bank output and take the logarithm of the output of the Mel filter bank. The equation is as follows:

$$I_m = \ln(P_m), 0 \le m \le M. \quad (25)$$

(iii) The discrete cosine transform is performed on the logarithmic energy obtained above, and the correlation between the characteristic parameters is reduced to obtain the 13th-order MFCC coefficient

(iv) In addition to extracting the static characteristics of MFCC coefficients, its dynamic characteristics are also analyzed. The extraction process is

$$D_n = \frac{\sum_{i=-k}^{k} i \cdot f(n+i)}{\sqrt{\sum_{i=-k}^{k} i^2}}. \quad (26)$$

where $f(n+i)$ represents the order of the $n+i$-th cepstral coefficient, and k represents the interval of the difference frame generally taking 1 or 2.

5.3. Feature Selection Strategy Based on Human Vocal Mechanism. In the process of objective evaluation of broadcasting vocal timbre, the selection of feature parameters usually is multidimensional to ensure the evaluation. In addition, there are many organs of control and cooperation, such as respiratory organs, vocal organs, resonance organs, and nervous system [26]. Therefore, it is very important to extract the appropriate feature parameters from the complex and diverse feature parameters to characterize the timbre feature completely and accurately. Preliminary experiments show that it is not that the more dimensions of the feature parameters are selected, the higher accuracy of the resulting classification is got. On the contrary, more dimensions will increase unnecessary calculations. For the feature parameters, the redundancy and correlation should be removed as much as possible in the dimension reduction. And it is also necessary to characterize the timbre features in combination with the mechanism of human vocal production. Therefore, the optimized feature selection module is combined with human vocal generation mechanism and PCA. Figure 3 shows the feature selection process.

5.3.1. Mechanism of Human Vocal. Broadcasting and hosting vocal production is a kind of purposeful artistic vocal, which originates from the spoken vocal in daily life, but it is more professional than that. In the process of broadcasting, perfect cooperation is needed for respiratory organs, vocal organs, and resonance organs to achieve good effect of the emotional expression. In order to establish a high-performance objective evaluation system of timbre, it is necessary to study the mechanism of sound production related to timbre and the auditory feature of human ears.

(1) Respiratory Organs. Respiratory organs are composed of lung, chest, diaphragm, abdominal muscle, trachea, and bronchus, which are the "source" power in vocal production. Figure 4(a) shows the specific structure.

After inhalation, the breath is distributed to the alveoli of the left and right lobes through the trachea and bronchus and then enters the bronchus from the outlet of the lung and converges to the trachea. The vocal cords vibrate by mobilizing the muscles of the chest and abdomen. The breath exhaled from the lung is the driving force of human vocal. The strength of human vocal and the resonance state are directly related to the speed of exhaled air flow and the size of flow pressure. Therefore, the respiratory organ is the power and energy guarantee of broadcasting, and the movement of muscle groups related to respiration also plays an important role.

(2) Vocal Organs. The vocal organ is composed of the larynx and vocal cord. The larynx is a complex system of cartilage and muscle that controls the movements of vocal cord, including cricoid cartilage, thyroid cartilage, arytenoid cartilage, lower airway, and upper pharyngeal cavity. The vocal cord is in the middle of the larynx. It is two symmetrical and elastic white ligaments juxtaposed horizontally. It is regulated by the cartilage and muscle in the larynx. Figure 4(b) shows the specific structure.

The vibration process of the vocal cord is related not only to the air velocity and pressure of breathing but also to the control of cartilage and muscle groups of the larynx.

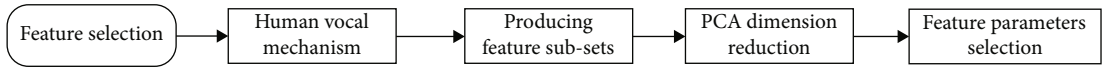


Figure 3: Feature selection process.

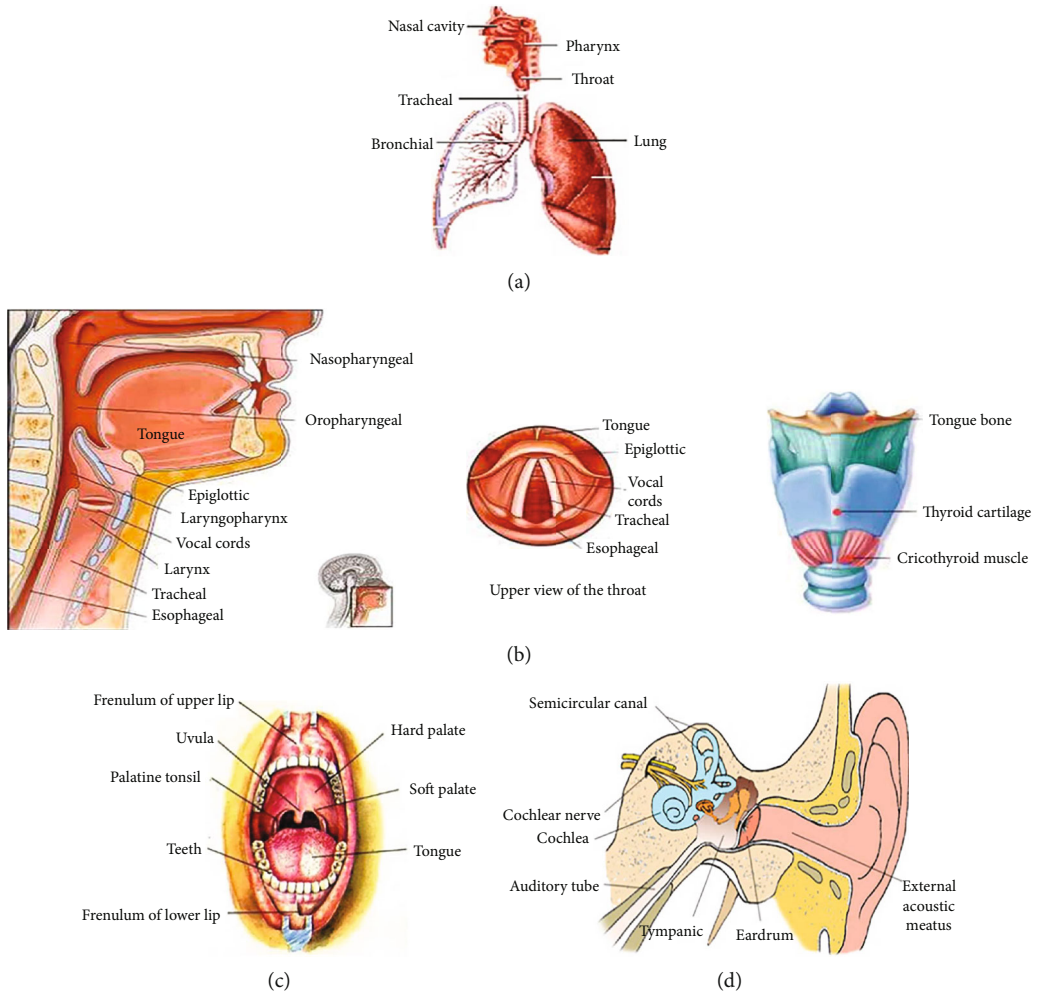


Figure 4: Composition of respiratory organs. (a) Structural diagram of respiratory organs. (b) Structural diagram of vocal organs. (c) Structure of resonance organ. (d) Structural diagram of auditory organs.

In the process of vocal broadcasting, it is necessary to fully mobilize the cartilage and muscle groups of the larynx, to control the breath and vocal organs to make the sound production stable and free.

(3) Resonance Organ. The resonance organ is composed of chest cavity, head cavity, and oral cavity. The chest cavity includes trachea, bronchus, and the whole lung. The head cavity includes nasal cavity, maxillary sinus, frontal sinus, and sphenoid sinus. The oral cavity includes upper and lower lips, upper and lower teeth, upper and lower gingiva, upper and lower jaws, tip of tongue, surface of tongue, root of tongue, and small tongue. The sound produced by vocal cords can obtain complex and colorful timbre only by adjusting the resonators. Oral cavity is the main speech resonator and manufacturer of various timbres. Figure 4(c) shows the specific structure.

(4) Auditory Organ. The auditory organ is composed of the external ear, the middle ear, and the inner ear. The outer ear includes the auricle, the external auditory canal, and the eardrum, and it is mainly used to collect sound, identify sound sources, and expand the sound of certain frequencies; the middle ear includes the malleus, incus, and stapes, which mainly transmits sound from the external auditory canal to the cochlea. The comprehensive function of the external ear and the middle ear is equivalent to a smooth band filter between 500 Hz and 6 KHz. The inner ear includes vestibular window, circular window, oval window, and cochlea. The main function of the cochlea is like a spectrum analyzer,

which can decompose complex signals into various frequency components. Figure 4(d) shows the specific structure.

According to the coordination of the related organs in vocal broadcasting, the selected features can be divided into three categories: source parameters reflecting the performance of respiratory organs, vocal tract parameters reflecting the performance of vocal organs and common vocal organs, and human hearing parameters reflecting the process of timbre perception. The parameters of feature classification are shown in Table 2.

5.3.2. Basic Mechanism of PCA. This paper mainly studies the individual characteristic parameters that can characterize the difference in timbre of the speaker during the process of broadcasting and vocalization. According to the principle of human vocalization and the auditory characteristics in the process of timbre perception, the commonly used feature parameters are divided into three categories, which include source parameters, vocal tract resonance parameters, and human hearing parameters. In the process of timbre classification, the input sample is a vector set of multiframe feature parameters of the entire speech. In the process of repeated training, there is a great correlation between frames. There is also redundant information between each feature parameter. In order to better retain the most effective and most reflective information of personalized timbre characteristics, PCA technology is utilized to perform feature optimization for selecting the effective components in the timbre characteristics. The basic mechanism of algorithm PCA is as follows:

(1) Calculate the mean value μ_i of the timbre feature vector

If each timbre feature sample dimension is set to, then each sample data can be expressed as

$$X = \left(x_1, x_2, x_3 \cdots, x_p\right)^T. \tag{27}$$

If the sample set has N pieces of sample data, the mean value of the sample set can be expressed as

$$\mu_i = \frac{1}{N}\sum_{i=1}^{N} X_i. \tag{28}$$

(2) Calculate the covariance matrix S_i of the timbre features as follows:

$$S_i = \frac{1}{N-1}\left(X_{\mathrm{i}} - \mu_i\right)^T \left(X_i - \mu_i\right). \tag{29}$$

(3) Calculate the eigenvalues $\lambda_1, \lambda_2, \lambda_3, \cdots \lambda_i$ and its corresponding eigenvector $w_1, w_2, w_3, \cdots w_i$ of the covariance matrix S_i and sort them in descending order

(4) Reduce the timbre feature dimension to k, and a new feature matrix K is formed by obtaining eigenvectors with larger eigenvalues. Matrix transformation is showed in Equation (30). The result Y_i is the set of timbre features after dimension reduction

$$Y_i = K^T X_i. \tag{30}$$

The cumulative contribution rate is used to select the principal component in determining the feature dimension. The eigenvalues $\lambda_1, \lambda_2, \lambda_3, \cdots \lambda_i$ are sorted from large to small, obtained from the abovementioned feature parameters. The ratio of the sum of the first k principal components to the total eigenvalue is the cumulative contribution rate of the first k principal components. Finally, the feature parameters are selected with the cumulative contribution rate greater than 85%.

5.4. Results of Timbre Feature Selection. Feature selections are generated according to the mechanism of human vocal generation, and all features are classified into three types, which include source parameters, vocal tract parameters, and human hearing parameters. Then, the dimensions of various feature combinations are reduced by PCA technology. Finally, the selected source parameters are pitch frequency, vocal tract parameters are formant frequency, bandwidth, and LPCC, and human hearing parameters are CFCC and MFCC. Table 3 shows the feature selection results.

In order to further verify the scientificity of feature selection, we conducted further experiments with unfiltered original features as a control group and compared the differences between filtered and unfiltered timbre feature parameters in the timbre objective evaluation model. Figure 5 shows the results.

It can be seen from Figure 5 that regardless of the male or female samples, in the timbre objective evaluation models of HMM, GMM-UBM, and LSTM, the timbre characteristic parameters after screening are more accurate than the original characteristic parameters classification. This further verifies the scientificity and necessity of selecting acoustic parameters based on the principle of human vocalization in the objective evaluation system of timbre.

6. Objective Evaluation Method

Before the experiment based on objective evaluation method, details of the experiment data, evaluation index, and experimental design were as follows.

(1) Experiment data: the speech samples in the database were processed uniformly. The preemphasis coefficient was set to 0.96, the frame length was 256 sampling points, and the frame shift was 128 sampling points. Hamming window was used to extract the vocal feature parameters of each speech sample,

Table 2: Feature classification parameters.

Source parameters	Short-term average energy, short-term average amplitude, pitch frequency, short-term energy change rate, long-term average energy, fundamental frequency change rate
Vocal tract parameters	Formant frequency, formant bandwidth, linear prediction cepstrum coefficient, line spectrum frequency
Human hearing parameters	Mel frequency cepstrum coefficient, cochlear frequency cepstrum coefficient, and perceptual linear prediction coefficient

Table 3: Feature selection results.

Source parameters	Pitch frequency
Vocal tract parameters	The frequency and bandwidth of the first, second, and third formants, linear predictive cepstrum coefficient
Human hearing parameters	Mel frequency cepstrum coefficient and cochlear frequency cepstrum coefficient

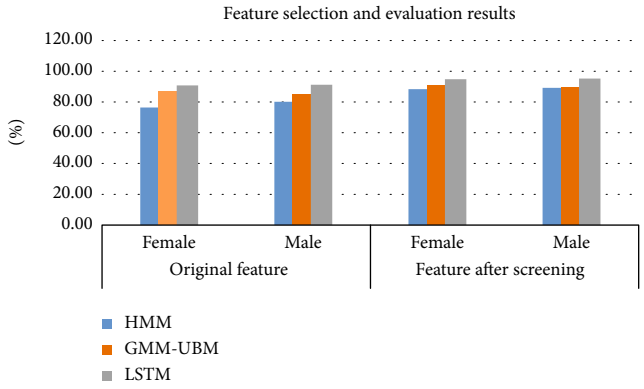


Figure 5: Feature selection and evaluation results.

and the 74-dimensional feature parameters were stored in an Excel table

(2) Evaluation index: 90% of the database was taken as the training selection, and the remaining 10% was taken as the test selection. The results were based on the speech samples of each 20 seconds test selection, and the accuracy of the broadcasting vocal timbre grading Acc is calculated by

$$\text{Acc} = \frac{Q}{U} \times 100\%, \tag{31}$$

where Q represents the number of speech samples with correct thin thickness classification in the test selection, and U represents the total number of speech samples contained in the test selection.

(3) Experimental design: the feature selection module based on the human vocal mechanism and PCA was used to divide the acoustic features into source parameters, vocal tract parameters, and human hearing parameters. The dimension of each feature type was reduced by PCA, and we conducted fusion and nonfusion experiments on the three types of optimal features, respectively. Among them, we used four timbre feature combinations for fusion. The inputs were 74-dimensional feature parameters, and the thin thickness of broadcasting vocal timbre was divided into three grades, which are *A*, *B*, and *C*. *A* is excellent, *B* is good, and *C* is pass. The accuracy of the final classification results was used to evaluate the selected three objective evaluation methods; so, this can explore a better objective evaluation method of broadcasting vocal timbre. Table 4 shows the relevant information of the experiment

6.1. Timbre Objective Evaluation Method Based on HMM

6.1.1. Experiment Workflow. Figure 6 shows the experiment workflow of the objective evaluation model based on HMM, and it mainly includes speech signal preprocessing, vocal timbre feature parameters extraction, HMM training parameters, and Viterbi algorithm recognition.

6.1.2. Analysis of Experimental Results. The source parameters, vocal tract parameters, and human hearing parameters were, respectively, analyzed on three types of characteristic parameters and their different characteristic combination types for objective evaluation of timbre. The accuracy of timbre classification was obtained by implementing the objective evaluation model based on HMM. Figure 7 shows the results.

The radar chart in Figure 7 shows that HMM obtains good accuracy in the objective evaluation of broadcast timbre. From the perspective of feature fusion, the grading effect obtained by using a single timbre characteristic parameter is not as good as the fusion timbre characteristic parameters. Among them, only the source parameters are used as the input, and the accuracy rate is only 77.25%. The classification effect obtained by the fusion of the three types of features as input is the best, and the accuracy rate can reach up to 89.28%, which further verifies the scientific nature of using the human vocal mechanism as the basis for the selection of timbre features. From the perspective of gender

Table 4: Experiments related information.

Speech language type	The number of samples	The number of training samples	The number of testing	Sampling frequency	Window type	Window size	Feature dimension	Fusion type	Classifier type
Chinese	6000	5400	600	44.1 KHz	Hamming window	256	74-dimensional	Early fusion	Discrete

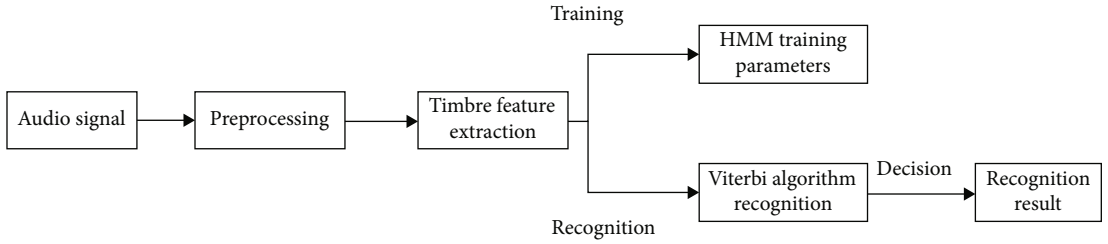


Figure 6: Workflow of the objective evaluation model based on HMM.

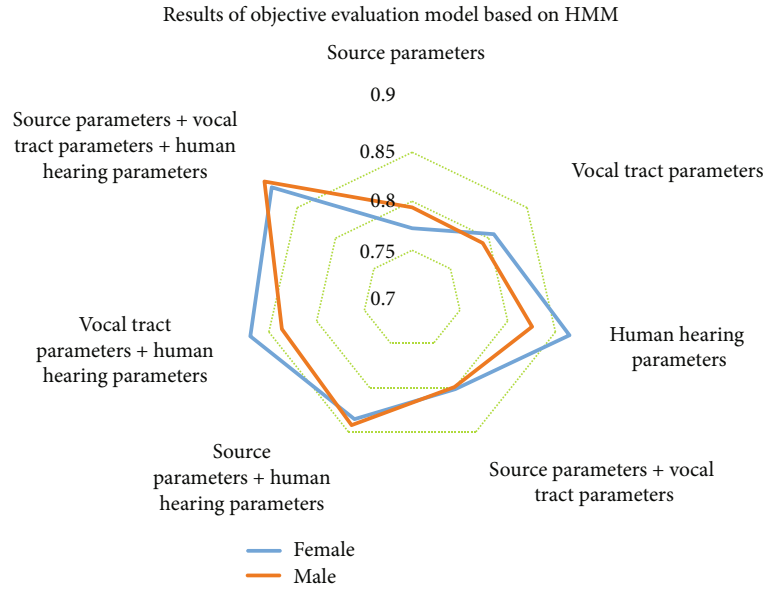


Figure 7: Results of the objective evaluation model based on HMM.

differences in timbre, when the same feature parameters combination is used as input, girls' timbre classification accuracy rate is generally higher than that of men. Especially when human hearing parameters are used as input, the accuracy rate of girls is about 4% higher than that of boys. Maybe it is because girls' timbre is more recognizable in terms of auditory perception.

6.2. Timbre Objective Evaluation Based on GMM-UBM

6.2.1. Experiment Workflow. Figure 8 shows the experiment workflow of the objective evaluation model based on the Gaussian Mixture Model-Universal Background Model (GMM-UBM), and it mainly has three important models, UBM training parameters, MAP self-adaption, and recognition result.

6.2.2. Network Parameter Setting

(i) The order of GMM model: in this experiment, 100 speech samples are selected, and 74-dimensional vocal timbre feature parameters are extracted, respectively. The training speech sample duration is 20 seconds, the testing speech sample duration is 6 seconds, and the GMM order is 4, 8, 16, 32, and 64. Table 5 shows the results

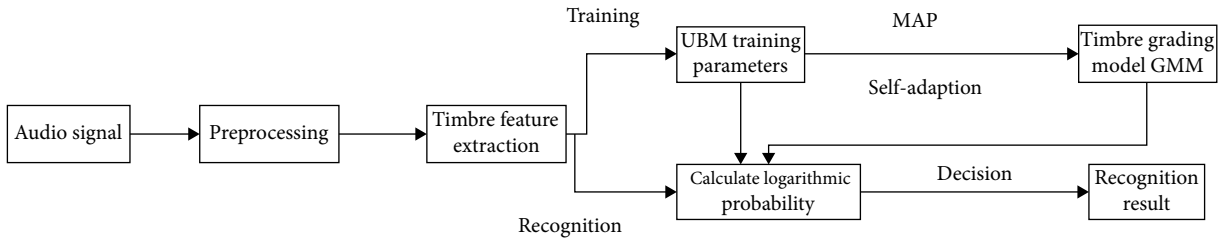


Figure 8: Workflow of the objective evaluation model based on GMM-UBM.

Table 5: Results of GMM model order.

GMM order	4	8	16	32	64
Classification accuracy	60.21%	71.35%	85.63%	89.48%	89.98%
Training duration	568.7 s	663.9 s	1035.1 s	1335.6 s	2147.3 s

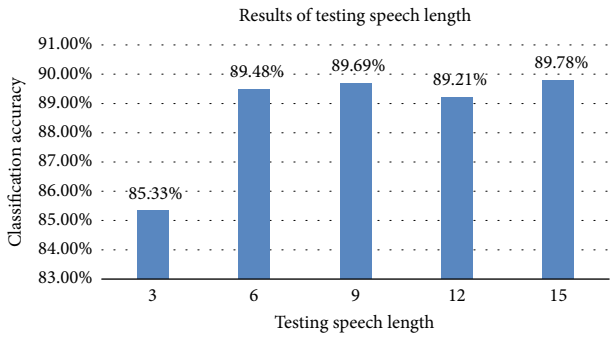


Figure 9: Results of testing speech length.

(ii) Length of the testing speech: in this experiment, 100 speech samples are selected, the GMM order is 32, the training speech sample duration is still taken as 20 seconds, and the testing speech sample is intercepted for 3 s, 6 s, 9 s, 12 s, and 15 s for comparative experiments. Figure 9 shows the results

The results show that with the increase of GMM order, the accuracy of 64 order classification is not obviously improved compared with that of 32 order classification; however, the training time is close to 2 times of that of 32 orders. In order to balance the relationship between classification accuracy and computational complexity, the order of GMM is selected as 32. When test speech length is greater than 6 s, the improvement of accuracy becomes very insignificant with the increase of test speech length. Because the test speech is too long, the model may not be able to cover and match the recognition process. Therefore, the test speech length was chosen as 6 s.

6.2.3. Analysis of the Results. The source parameters, vocal tract parameters, and human hearing parameters were, respectively, analyzed on three types of characteristic parameters and their different characteristic combination types for objective evaluation of timbre. The accuracy of timbre classification was obtained by inputting objective evaluation model based on GMM-UBM. Figure 10 shows the results.

The results show that the improved GMM-UBM generally improves the vocal timbre classification effect compared with the traditional HMM, but it still has the feature fusion problem as HMM. Overall, the fusion feature is better than the unfused feature classification effect. When source parameters, vocal tract parameters, and human hearing parameters are used as input, the accuracy of the objective evaluation of women's timbre can reach 90.5%. In addition, when human hearing parameters are used as input, the accuracy of objective evaluation of both male and female timbre reaches more than 90%. This shows the scientific nature of selecting timbre parameters based on the human vocal mechanism.

6.3. Timbre Objective Evaluation Based on LSTM

6.3.1. Experiment Workflow. The key point of the objective evaluation based on long short-term memory (LSTM) is to estimate the weight parameters of gate structure in the network training. Figure 11 shows the experiment workflow of the objective evaluation model based on LSTM.

6.3.2. Network Parameter Setting. The size of the hidden layer and the number of iterations has different influence degrees on the model classification accuracy in the training process. Therefore, this paper makes a comparative analysis of the size of the hidden layer and the number of iterations of the LSTM network through the design of preliminary experiments, to determine several important parameters affecting the network performance.

(i) Number of iterations: using the LSTM network model, the initial learning rate is 0.05, the learning rate decline cycle is 50, the decline factor is 0.2, and the dropout parameter is 0.2. The timbre feature data

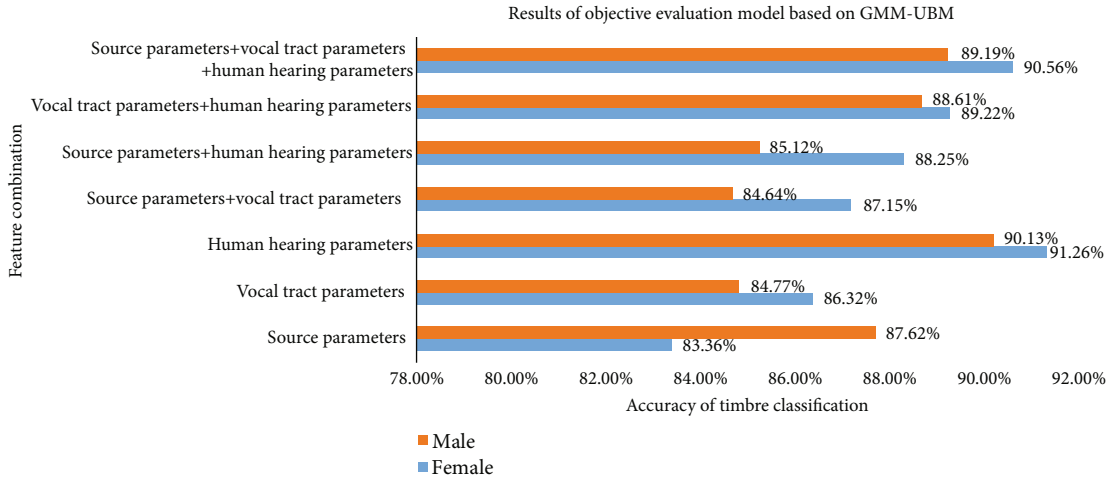


Figure 10: Results of objective evaluation model based on GMM-UBM.

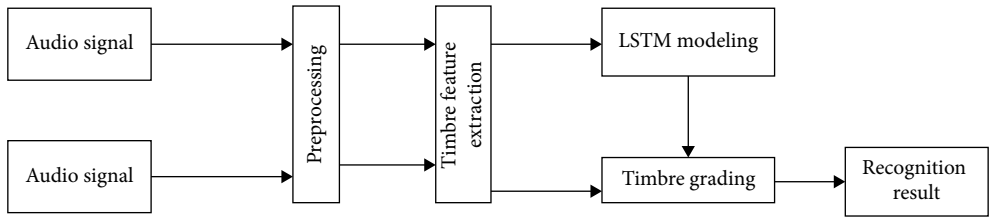


Figure 11: Workflow of objective evaluation model based on LSTM.

Table 6: Results of iteration times.

Number of iterations	25	50	75	100	125	150	175	200
Accuracy/%	89.21	91.54	92.83	93.66	94.21	94.89	94.97	95.01
Training duration/s	1903.8	2509.4	3147.9	3706.9	4387.2	5006.7	5701.9	6421.8

are input into the model, and the iteration times are set as 25, 50, 75, 100, 125, 150, 175,and 200. The evaluation indexes such as classification accuracy and training time are obtained by model training, and the appropriate iteration times are obtained by comparison. Table 6 shows the results

(ii) Hidden layer scale: using the LSTM network model, the initial learning rate is 0.05, the learning rate decline cycle is 50, the decline factor is 0.2, the dropout parameter is 0.2, and the iteration times are set to 150. The timbre feature data are input into the model, and the hidden layer nodes are set as 50, 100, 150, 200, 250, and 300. The evaluation indexes such as classification accuracy and training time are obtained by comparison. Table 7 shows the results

The results show that when the number of iterations is more than 150, the accuracy is no longer significantly improved, but the training time is still greatly increased. Considering the accuracy of classification, the system operation time is shortened as much as possible; so, the iteration number of LSTM network is selected as 150.When the number of hidden layer nodes continues to 300, the accuracy rate begins to decline to a certain extent. Because the number of neurons is more, the convergence speed of the model is slower. Considering the premise of meeting the classification accuracy, the number of hidden layer nodes is 200 in LSTM network.

6.3.3. Analysis of Experimental Results. The source parameters, vocal tract parameters, and human hearing parameters were, respectively, analyzed on three types of characteristic parameters and their different characteristic combination types for objective evaluation of timbre. The accuracy of timbre classification was obtained by inputting the objective evaluation model based on LSTM. Figure 12 shows the results.

The results show that the LSTM network using the deep learning algorithm has better classification effect than the

Table 7: Results of hidden layer node number.

Number of hidden layer nodes	50	100	150	200	250	300
Accuracy/%	93.64	93.98	94.21	94.97	94.32	93.57
Training duration/s	2631.7	3409.4	4194.2	5094.8	5901.6	8004.9

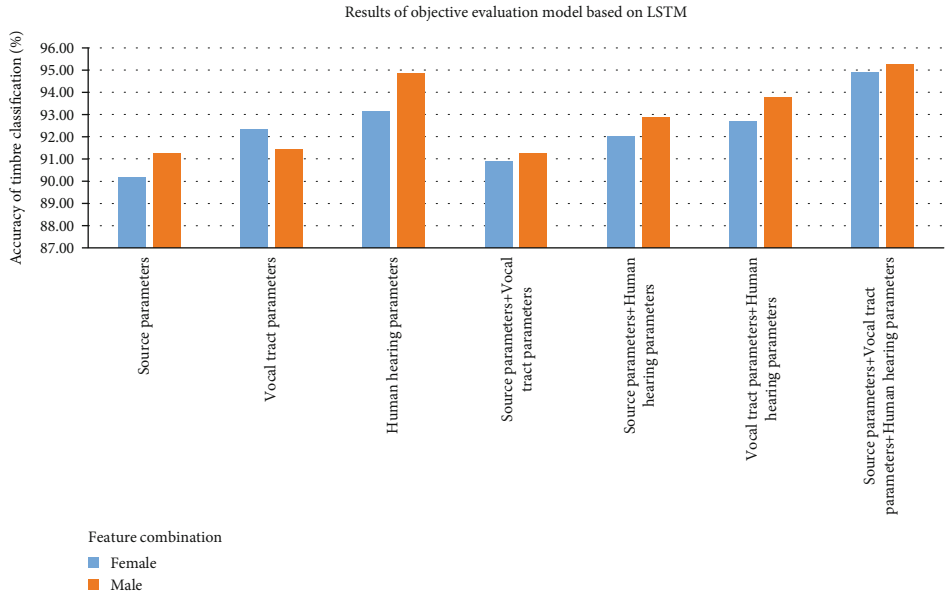


Figure 12: Results of the objective evaluation model based on LSTM.

traditional HMM and the improved GMM-UBM in the task of vocal timbre classification. Regardless of whether it is a fusion of timbre characteristic parameters or not, the accuracy of objective evaluation of LSTM-based broadcast timbre is above 90%. When the source parameters, vocal tract parameters, and human hearing parameters are used as input, the overall accuracy of objective evaluation of timbre is about 5% higher than that of HMM and GMM-UBM. This is because the LSTM network continues the advantages of the recurrent neural network in dealing with timing problems, and it still has a good expression ability for the long-term speech classification.

7. Conclusions

This paper is based on the background of the selection and training of broadcasting talents by various professional colleges and recruitment units. Linguistics, journalism, and psychology form the basis of our research. A series of studies have been carried out to get a set of objective grading methods for vocal timbre with scientific evaluation and flexible operation. Combining the mechanism of human vocalization and PCA technology, we divide the characteristic parameters of broadcast timbre into three categories, which include source parameters, vocal tract parameters, and human hearing parameters. The objective evaluation experiment of broadcast timbre is carried out by using the combination of these timbre characteristics. Experiments show that the selection of timbre parameters based on the human vocal mechanism is scientific. Furthermore, the objective evaluation method based on LSTM has better classification effect than the traditional HMM and GMM-UBM in the timbre classification. The LSTM method introduces a deep model for complex audio signals with stronger representation ability and more complex structure. It can better simulate the nonlinear dynamic process of human vocal with levels of nonlinear transformation, and it has excellent performance in the vocal timbre classification. The classification accuracy rate is about 95%.

In the future, we intend to improve the performance of the objective evaluation system of timbre through different fusion technologies, such as late fusion based on score fusion, namely, mean, maximum, and linear weighted sum fusion [24], and plan to combine the *i*-vector approach to improve the accuracy of the timbre objective evaluation system [23]. In addition, we are currently collecting a large timbre speech evaluation database containing other categories to further improve the proposed model.

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 they have no conflicts of interest.

Acknowledgments

This work was supported by the National Key R&D Program of China (2018YFB1404100).

References

[1] W. He, Y. Zou, and Y. H. Ma, "CUCBNC: a broadcast news voice database introducing broadcasting knowledge," *Journal of Tsinghua University*, vol. 51, no. 9, pp. 1313–1316, 2011.

[2] S. Rawat, T. Khan, and A. Chauhan, "PRAAT software; utilization of computerized approach for determination of variation present in recorded audios from distinct sources," *International journal of engineering trends and technology*, vol. 67, no. 3, pp. 111–114, 2019.

[3] K. L. Spreadborough and I. Anton-Mendez, "It's not what you sing, it's how you sing it: How the emotional valence of vocal timbre influences listeners' emotional perception of words," *Psychology of Music*, vol. 47, no. 3, pp. 407–419, 2019.

[4] G. Sharma, K. Umapathy, and S. Krishnan, "Trends in audio signal feature extraction methods," *Applied Acoustics*, vol. 158, article 107020, 2020.

[5] H. Purwins, B. Li, T. Virtanen, J. Schlüter, S.-Y. Chang, and T. Sainath, "Deep learning for audio signal processing," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 2, pp. 206–219, 2019.

[6] A. Pandey and D. L. Wang, "On cross-corpus generalization of deep learning based speech enhancement," *IEEE-ACM Transactions on Audio Speech and Language Processing*, vol. 28, pp. 2489–2499, 2020.

[7] F. Pishdadian, G. Wichern, and J. Le Roux, "Finding strength in weakness: learning to separate sounds with weak supervision," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 2386–2399, 2020.

[8] G. Gosztolya, "Posterior-thresholding feature extraction for paralinguistic speech classification," *Knowledge-Based Systems*, vol. 186, article 104943, 2019.

[9] T. Zoughi, M. M. Homayounpour, and M. Deypir, "Adaptive windows multiple deep residual networks for speech recognition," *Expert Systems with Applications*, vol. 139, article 112840, 2020.

[10] X. Zhang, X. Zou, M. Sun, T. F. Zheng, C. Jia, and Y. Wang, "Noise robust speaker recognition based on adaptive frame weighting in GMM for I-vector extraction," *IEEE Access*, vol. 7, pp. 27874–27882, 2019.

[11] M. T. S. Al-Kaltakchi, R. R. O. Al-Nima, and M. A. M. Abdullah, "Comparisons of extreme learning machine and backpropagation-based i-vector approach for speaker identification," *Turkish Journal of Electrical Engineering and Computer Sciences*, vol. 28, no. 3, pp. 1236–1245, 2020.

[12] S. Yang, S. Arif, M. Jia, and S. Zhong, "SAL-Net: Self-supervised attribute learning for object recognition and segmentation," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 2891303, 13 pages, 2021.

[13] M. Yan, X. Lou, and Y. Wang, "Channel noise optimization of polar codes decoding based on a convolutional neural network," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 1434347, 10 pages, 2021.

[14] A. Aggarwal, A. Srivastava, A. Agarwal et al., "Two-way feature extraction for speech emotion recognition using deep learning," *Sensors*, vol. 22, no. 6, 2022.

[15] M. B. Akçay and K. Oğuz, "Speech emotion recognition: emotional models, databases, features, preprocessing methods, supporting modalities, and classifiers," *Speech Communication*, vol. 116, pp. 56–76, 2020.

[16] M. Yan, S. Li, C. A. Chan, Y. Shen, and Y. Yu, "Mobility prediction using a weighted Markov model based on mobile user classification," *Sensors*, vol. 21, no. 5, 2021.

[17] M. Yokoyama, "Possibility of distinction of violin timbre by spectral envelope," *Applied Acoustics*, vol. 157, article 107006, 2020.

[18] C. Jin, Z. Song, J. Xu, and H. Gao, "Attention-based Bi-DLSTM for sentiment analysis of Beijing opera lyrics," *Wireless Communications and Mobile Computing*, vol. 2022, Article ID 1167462, 8 pages, 2022.

[19] R. Pradeep and K. S. Rao, "Incorporation of manner of articulation constraint in LSTM for speech recognition," *Circuits, Systems, and Signal Processing*, vol. 38, no. 8, pp. 3482–3500, 2019.

[20] Y. Zhuo, L. Yan, W. Zheng, Y. Zhang, and C. Gou, "A novel vehicle detection framework based on parallel vision," *Wireless Communications and Mobile Computing*, vol. 2022, Article ID 9667506, 11 pages, 2022.

[21] J. U. Bang, M. Y. Choi, S. H. Kim, and O. Kwon, "Automatic construction of a large-scale speech recognition database using multi-genre broadcast data with inaccurate subtitle timestamps," *IEICE Transactions on Information and Systems*, vol. -E103.D, no. 2, pp. 406–415, 2020.

[22] M. Al-Qaderi, E. Lahamer, and A. Rad, "A two-level speaker identification system via fusion of heterogeneous classifiers and complementary feature cooperation," *Sensors*, vol. 21, no. 15, p. 5097, 2021.

[23] M. T. S. Al-Kaltakchi, M. A. M. Abdullah, W. L. Woo, and S. S. Dlay, "Combined i-vector and extreme learning machine approach for robust speaker identification and evaluation with SITW 2016, NIST 2008, TIMIT databases," *Circuits, Systems, and Signal Processing*, vol. 40, no. 10, pp. 4903–4923, 2021.

[24] M. T. S. Al-Kaltakchi, W. L. Woo, S. Dlay, and J. A. Chambers, "Evaluation of a speaker identification system with and without fusion using three databases in the presence of noise and handset effects," *EURASIP Journal on Advances in Signal Processing*, vol. 2017, no. 1, Article ID 80, pp. 1–17, 2017.

[25] R. Jahangir, Y. W. Teh, N. A. Memon et al., "Text-Independent speaker identification through feature fusion and deep neural network," *IEEE Access*, vol. 8, pp. 32187–32202, 2020.

[26] M. Yan, W. Li, C. A. Chan, S. Bian, I. Chih-Lin, and A. F. Gygax, "PECS: towards personalized edge caching for future service-centric networks," *China Communications*, vol. 16, no. 8, pp. 93–106, 2019.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 1167462, 8 pages
https://doi.org/10.1155/2022/1167462

Research Article

Attention-Based Bi-DLSTM for Sentiment Analysis of Beijing Opera Lyrics

Cong Jin,[1] **Zhen Song**,[2] **Jiaqi Xu**,[2] **and Huiyue Gao**[2]

[1]*School of Information and Communication Engineering, Communication University of China, Beijing 100024, China*
[2]*Advanced Research Center for Digitalization of Traditional Drama, The Central Academy of Drama, Beijing 100006, China*

Correspondence should be addressed to Zhen Song; songzhen@zhongxi.cn

Received 28 August 2021; Revised 15 October 2021; Accepted 19 April 2022; Published 6 May 2022

Academic Editor: Carles Gomez

Copyright © 2022 Cong Jin 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.

We propose a sentiment analysis model based on Bi-DLSTM to solve the problem of sentiment analysis of Beijing Opera lyrics. A Bi-LSTM network with dilated recurrent skip connections (Bi-DLSTM) is introduced in this model, which can improve the ability to exact long-sequence information. The proposed model can learn the dependence of long sequences in different time dimensions and effectively improve the semantic extraction performance of lyrics. The attention mechanism is introduced to ensure the recognition of the more important words in the text sequence, which further improves the performance of the network. In order to solve the problem of lack of data on lyric sentiment analysis on the Internet, we build a dataset that can be used for lyric sentiment analysis. This paper completes multiple experiments on four datasets and verifies the effectiveness of the proposed model.

1. Introduction

Beijing Opera has a profound cultural heritage and literary value. Using artificial intelligence to analyze the emotions of Beijing Opera is helpful for its inheritance and development. With the development and improvement of Internet music software, various diversified services have been introduced, and intelligent song search and recommendation systems have gradually emerged. The emotional analysis of lyrics is the key technology to realize this function. Studies have shown that [1–3] lyrics are an important feature in the emotional classification of songs. When analyzing the emotional content of Beijing Opera, it is extremely valuable to consider the emotional attributes of the lyrics. At the same time, the field of natural language processing has developed rapidly, and neural networks have been widely used. Text sentiment analysis has been an important field in natural language processing, and the technology of text sentiment analysis has been very mature.

Lyric as a special form of text representation has more lyrical and artistic expressions and more complex semantics than ordinary written texts, such as lyrics: "叹只叹东风起火烧战船" and "何处悲声破寂寥." Therefore, sentiment analysis of the lyrics is more difficult. At present, the current work on sentiment analysis of lyrics is relatively scarce. When the ordinary text sentiment analysis method is used to deal with the lyrics with artistic expression, the extraction of sentiment semantics and deep features in the lyrics will be insufficient, which will affect the accuracy of lyric sentiment classification. Therefore, it is the key to excavate the emotional semantic information in lyrics accurately.

In this paper, we propose a sentiment analysis model based on Bi-DLSTM network, aiming at the problem that the traditional LSTM architecture will have a certain deviation when processing long-sequence information, because it can only capture semantics in one direction. We designed a Bi-DLSTM network composed of two layers of Dilated LSTM [4] network stack. The two layers have the same structure and opposite directions, which can capture bidirectional semantics. This variant LSTM network structure can not only realize parallel computing when the information transmission span becomes larger, but also can interpret and extract sequence information from a context perspective, which is more conducive to the retention of long-

sequence information. In the model, the Bi-LSTM network and the Bi-DLSTM network are combined to extract information in different time dimensions to achieve deep mining of textual semantic information and emotional representation. The Bi-DLSTM network alleviates the problem of long-sequence learning, but it cannot identify the more important words in the text sequence. Therefore, the model introduces an attention mechanism to further improve the performance of the network.

2. Related Work

Sentiment analysis based on texts in specific fields such as lyrics belongs to the category of text sentiment analysis and is common research in natural language processing. At present, deep learning methods can simulate human brain thinking analysis methods obtaining data feature representations through automatic learning and encode the semantic and grammatical attributes of text to provide relatively accurate text representation information.

In the early research on natural language processing, the bag-of-words representation model (BOW) is one of the most common text representation models. Later, with the rise of neural networks, the concept of word embedding was proposed to solve the problem of dimensional disaster. The initial word embedding method was not mature and was not applied on a large scale. Bengio et al. [5] first proposed the NNLM language model using neural networks, which can be trained to obtain a low-dimensional dense word vector representation. Mikolov et al. [6] proposed a new word embedding method Word2Vec, which simplifies the structure of the NNLM model, and the pretrained word vector can effectively represent the text. Later, in order to solve the shortcoming that the model usually ignores the order of words in the text, researchers introduced sequence representation models, such as recurrent neural network (RNN) [7–9] and convolutional neural network (CNN) [10, 11]. With the development of the field of natural language processing and the change of various new technologies, many novel sentiment analysis models have emerged, such as structure-enhanced LSTM [12, 13] and the combination of RNN and CNN models [14]. Attention mechanism is a novel and effective technique, which has been widely used in natural language processing in recent years. It has shown advantages in many fields such as document classification [15], sentiment classification [16], and sentence representation [17].

At present, a large number of music-based researches have shown that it is necessary to conduct in-depth research on lyrics when analyzing and categorizing music. In most song classification tasks, the performance of music classifiers incorporating text features is better than pure audio classifiers [18, 19]. Lyrics are usually easier to obtain and process than audio data. For nonprofessional musicians, when interacting with the music system, they tend to pay more attention to the lyrics than audio information [20]. Kim and Kwon [21] proposed a feature selection method based on syntactic analysis, using four syntactic analysis rules to extract emotional features from lyrics for emotional classification of lyrics. Mihalcea and Strapparava [22] used music and lyrics to characterize songs and completed the classification of song emotions. Fell and Sporleder [23] proposed a lyrics analysis method that combines n-gram model and complex features to model lyrics text from different dimensions and achieved performance improvements in three different classification tasks. Rachman et al. [24] combined lyrics and audio features, using two emotional models, classification model, and dimensional model, to complete the emotional classification of songs.

3. Learning Model

3.1. Embedding Layer. In the first stage of the model, an embedding layer is applied to encode the lyrics text and emotional tags to obtain their unified representation. The word vector tool selected is Word2Vec. Before inputting the word vector, it is necessary to train the model based on the corpus. The lyrics of 1500 Beijing Opera songs collected on the Internet are selected as the corpus for training for the model. Previous studies [25, 26] showed that the highest performance of the model can be achieved when the dimensionality of the word vector is set in 100 dimensions. In the 100 dimension, it has good convergence, and the loss value is the smallest, so 100-dimensional word vector is selected for training of Beijing Opera lyrics. The trained lyrics vector is input into the Bi-LSTM network in the form of sequence $w = [x_1, x_2, \cdots, x_n]$ to learn the semantic and syntactic information of the lyrics.

After the training is completed, the word vector library can realize the functions of printing the corresponding word vector, comparing the semantic similarity, and adding and subtracting the word vector. For example, if the input word is "离开," the system can output its corresponding sequence of synonyms. The top three are "离去," "别离," and "消失," and their similarity scores are "0.846201," "0.759595," and "0.697776." The structure of the sentiment analysis model based on the Bi-DLSTM network is shown in Figure 1.

3.2. Semantic Extraction Layer. The semantic extraction layer uses the Bi-LSTM network because it is suitable for language and time series data [27]. The LSTM network maps the input x to the output y by learning the hidden representation h_t; the formula is as follows:

$$y = f(h_{t-1}, x_t). \tag{1}$$

t represents different moments. The loss function needs to be minimized during training. The formula is as follows:

$$L(x, y) = -\frac{1}{N} \sum_{n \in N} \quad x_n \log y_n. \tag{2}$$

The LSTM network manages its weight updates through gate structures, which determine the amount of information that should be retained and forgotten at each moment. The gate structure contains three kinds of "gates": "input gate" i_t, "forgotten gate" f_t, and "output gate" o_t. The "input gate" determines how much new information is added at each

moment to update the current cell state; the "forgotten gate" determines not to keep irrelevant information, and the "output gate" determines the output and the value of the next hidden state. The calculation process is as follows:

$$\begin{aligned} i_t &= \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i),\\ f_t &= \sigma\left(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f\right),\\ c_t &= f_t c_{t-1} + i_t \tan h(W_{xc}x_t + W_{hc}h_{t-1} + b_c),\\ o_t &= \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_{t-1} + b_o),\\ h_t &= o_t \tan h(c_t). \end{aligned} \tag{3}$$

σ is the sigmoid function, c_t is the unit state, W is the weight matrix, b is the bias term, and h_t is the hidden state. Then Bi-LSTM is used to obtain information from both directions of each word to capture contextual information. It combines forward and backward hiding states. Forward hiding states $\overleftarrow{h_t}$ reading every word in the lyrics from w_i1 to w_iT, and backward hiding states $\overrightarrow{h_t}$ reading words in the lyrics from w_iT to w_i1. The formula is as follows:

$$\begin{aligned} x_i t &= W_e w_i t, t \in [1, T],\\ \overleftarrow{h_t} &= \overleftarrow{LSTM}(x_i t), t \in [1, T],\\ \overrightarrow{h_t} &= \overrightarrow{LSTM}(x_i t), t \in [1, T]. \end{aligned} \tag{4}$$

Each lyric t contains w_i words, where $w_i, i \in [0, T]$ represents the ith word in the librettos and $x_i t$ represents the word vector matrix in the embedding.

Finally, all the outputs of the forward hidden state and the backward hidden state are connected to obtain the final output o, which contains the contextual semantic information in the librettos, then input o into the Bi-DLSTM network.

3.3. Bi-DLSTM Layer. This layer of network consists of two layers of Dilated LSTM network stacks with opposite information transfer directions, and the dilation of each layer of network is set to 2, in order to achieve the effect of focusing on the extraction of sequence information in different time dimensions with the previous Bi-LSTM network layer. The most important part of the architecture is the dilated recurrent skip connection in the LSTM cell. The formula is as follows:

$$c_t^{(l)} = LSTM\left(o_t, c_{t-2}^{(l)}\right). \tag{5}$$

$c_t^{(l)}$ represents the cell state of the lth layer at time t, and o_t is the input at time t in the LSTM layer. The calculation process of the Bi-DLSTM network layer is as follows:

$$\begin{aligned} \overleftarrow{h_t} &= DLSTM\left(\overleftarrow{h_{t-1}}, s_t\right),\\ \overrightarrow{h_t} &= DLSTM\left(\overrightarrow{h_{t+1}}, s_t\right),\\ h_t &= \left[\overleftarrow{h_t}, \overrightarrow{h_t}\right]. \end{aligned} \tag{6}$$

s_t represents the state of the input at time t, $\overleftarrow{h_t}$ represents the forward hidden state, and $\overrightarrow{h_t}$ represents the backward hidden state. We obtain the semantic information of a given word by connecting the forward and backward hidden state. Finally, the final semantic layer hidden representation h_t is fed into the attention layer.

Compared with the traditional Bi-LSTM network, the Bi-DLSTM network layer proposed in this section has two advantages over the traditional Bi-LSTM network on the problem of long sequences: (1) it can focus on the extraction of sequence information in different dilated dimensions; (2) it reduces the path length of information transmission in the network and enables the network model to learn complex long-term dependencies. By combining the Bi-LSTM layer and the Bi-DLSTM layer, the model can learn the text sequence in two dilated dimensions, and extract the deep semantic information and emotional representation of the librettos text.

3.4. Attention Layer. The Bi-DLSTM layer solves the problem of long-sequence learning and performs more in-depth semantic feature extraction, but not every word or word in the lyrics sequence has the same importance and weight. Therefore, the model introduces an attention mechanism to expand this network.

The attention layer of this chapter is inspired by [14], through the attention network to find the most important words for the meaning of the lyrics. Therefore, an attention mechanism is introduced to extract words that are important to the meaning of a sentence, and the representations of these informative words are aggregated to form a sentence vector. Using the output of the Bi-DLSTM network as the input of the attention layer, the calculation formula is as follows:

$$\begin{aligned} u_{tw} &= \tan h(W_w h_t + b_w),\\ \alpha_{tw} &= \frac{\exp\left(u_{tw}^T u_w\right)}{\sum_t \exp\left(u_{tw}^T u_w\right)},\\ L_t &= \sum_i \left(\alpha_{tw} h_t\right). \end{aligned} \tag{7}$$

h_t represents the corresponding word annotation representation output by the Bi-DLSTM layer, W_w and b_w represent paranoid parameters, u_{tw} is the corresponding hidden representation obtained through word annotations h_t using a layer of MLP network and then calculates an importance weight α_{tw} by calculating the similarity between the hidden representation u_{tw} and the word-level context vector u_w, which can measure the importance of the word in the document, and L_t represents the corresponding lyrics vector.

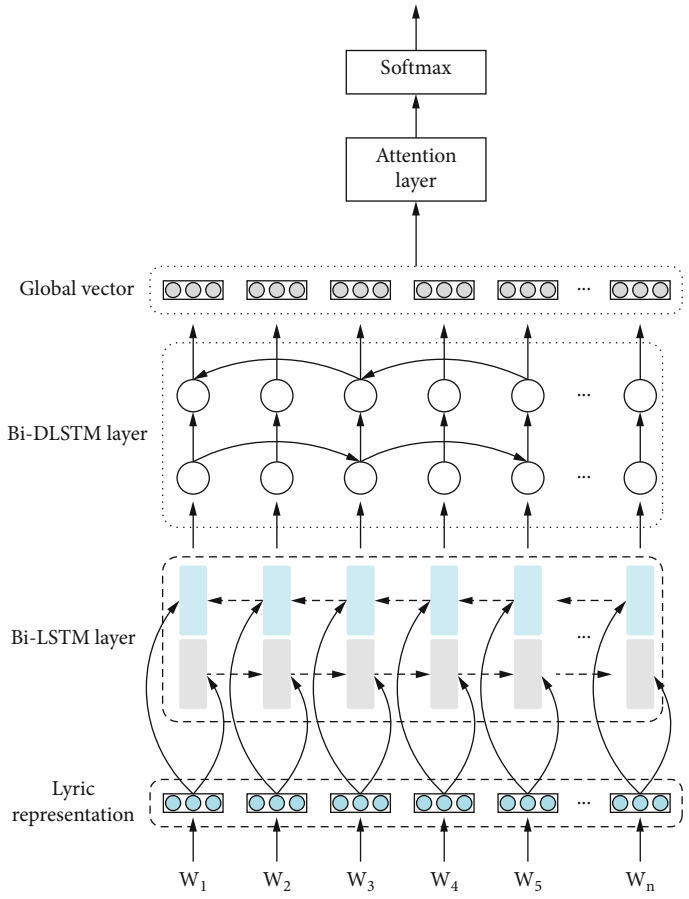


Figure 1: Model structure.

Then, the output of the attention layer obtained is calculated by the softmax function, and the classification result is obtained, and the formula is as follows:

$$P = softmax(W_t L_t + b_t). \tag{8}$$

W_t and b_t represent $softmax$ layer parameters. Finally, use the negative log likelihood of the correct label as the training loss:

$$Loss = -\sum_{L} \log p_{Ly}. \tag{9}$$

y is the emotion label of the corresponding lyric document L.

4. Self-Built Data

At present, the open-source datasets for lyrics sentiment analysis on the Internet are relatively scarce and of low quality. We combined the working mechanism of the sentiment analysis model based on deep learning and the hardware experimental environment to construct a database that can be used for sentiment analysis of Beijing Opera lyrics.

In text sentiment analysis, the distinction between sentiment categories is usually more obvious. For example, based on the six basic emotions of Ekman [59], each sentiment category contains independent emotional attributes and does not contain dependencies. In this paper, it is not feasible to divide the lyrics into six basic emotions, because although the lyrics contain rich emotions, the types of emotions expressed are more concentrated, such as happiness, praise, love, sadness, and melancholy, while the emotions in lyrics such as fear, disgust, and anger are rare and not easy to define the classification. At the same time, each emotion has a different degree of expression in different lyrics, and there are different emotions. If the data is classified according to the six basic emotions, there will be great differences in the quantity and quality of the data between each category, and the difficulty of data sorting is also a big challenge. Therefore, for the creation of the dataset for lyrics sentiment analysis, we chose to build on the basis of two major emotional categories, positive and negative.

The original data of the lyrics comes from the lyrics of more than 1500 Beijing Opera songs of different styles on the Internet. Based on these lyrics, through the steps of sentence segmentation, emotional subjective analysis, and tagging, a dataset that can be used for lyrics emotional polarity analysis is created. Lyrics are divided into two major

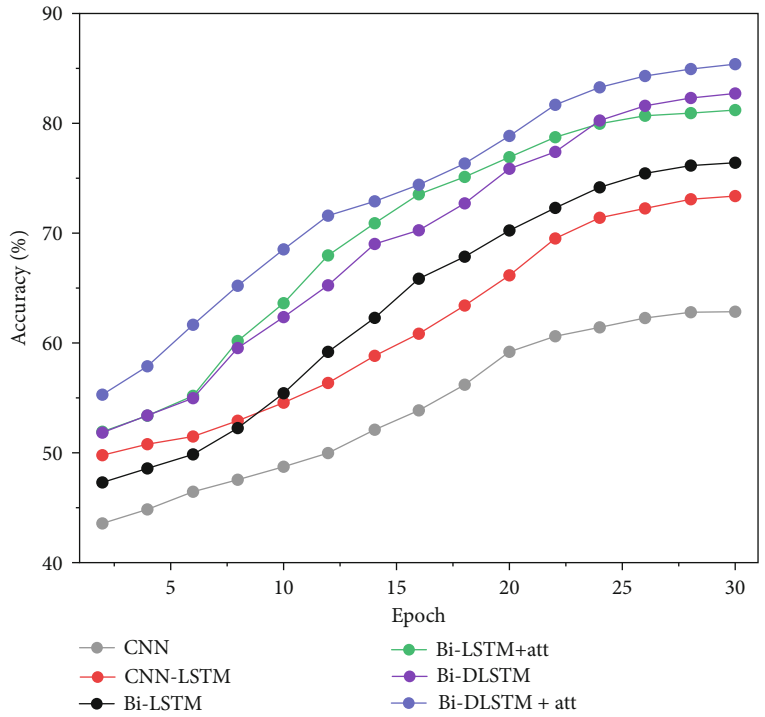


Figure 2: Classification accuracy of six models in self-built lyrics dataset.

emotional categories, positive and negative, and each degree of emotional category includes about 3500 lyrics; the number of words in each lyrics ranges from 7 to 20, and the average number of words is about 10.

5. Experiments

5.1. Comparative Data. We will use the self-built lyrics dataset and three Chinese pop songs based on datasets for comparative experiments, and the three Chinese pop songs based on datasets are divided into seven categories of emotions, including happiness, like, anger, sadness, fear, disgust, and surprise. There are some obvious expression differences between the lyrics of Beijing Opera and Chinese pop songs. For example, in Beijing Opera, "Happy" is expressed more frequently with the word "欢喜," but in Chinese pop songs, it is expressed more frequently with the word "快乐." We have already paid attention to that when collecting the data. We will use 80% of the data for training and the remaining 20% for testing.

5.1.1. The Chinese Pop Song Lyrics Dataset One. The Chinese pop song lyrics dataset one sentiment dataset contains 14000 Chinese pop songs lyrics with sentiment annotations and is based on seven emotions.

5.1.2. The Chinese Pop Song Lyrics Dataset Two. The Chinese pop song lyrics dataset two sentiment dataset contains close to 20,000 Chinese pop songs lyrics with sentiment annotations and is based on seven emotions.

5.1.3. The Chinese Pop Song Lyrics Dataset Three. The Chinese pop song lyrics dataset three contains close to 40,000 Chinese pop songs lyrics with sentiment annotations and is based on seven emotions.

Table 1: Model experiment comparison.

Learning model	Precision	Recall	F1-score
CNN	0.625	0.619	0.628
CNN-LSTM	0.731	0.743	0.741
Bi-LSTM	0.763	0.757	0.759
Bi-LSTM+attention	0.804	0.798	0.810
Bi-DLSTM	0.823	0.833	0.824
Bi-DLSTM+attention	*0.854*	*0.846*	*0.849*

The reason for collecting three Chinese pop songs lyrics dataset instead of only one is because Chinese pop songs have different genres; for instance, some of them are rock style with a fast rhythm, and some of them are romantic with a slow rhythm. So the lyrics may vary a lot according to the difference in the rhythm. Therefore, we built three datasets to reduce the influence from the different genres.

5.2. Baselines. We use five baseline models and the proposed model for comparative experiments.

5.2.1. CNN. The semantic extraction layer uses a CNN 2-D convolutional network with two fully connected layers. The size of the convolution kernel is 3, 4, and 5; the number of convolution kernels is 300; the learning rate is 0.001; batch size is 128; Adam optimizer, dropout is 0.5; and epoch is 30.

5.2.2. CNN-LSTM. The semantic extraction layer uses a network architecture that combines CNN and LSTM. The size

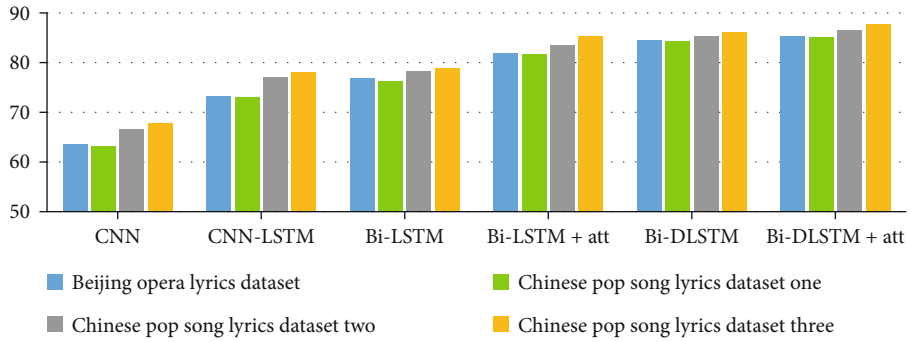


Figure 3: Comparison of classification accuracy in self-built lyrics dataset and Chinese pop song lyrics dataset.

of the convolution kernel is 3, 4, and 5; the number of convolution kernels is 300; the learning rate is 0.001; batch size is 128; Adam optimizer, dropout is 0.5; epoch is 30; and the number of LSTM hidden units is 128.

5.2.3. Bi-LSTM. The semantic extraction layer uses a Bi-LSTM network. The learning rate is 0.001, the batch size is 128, Adam optimizer, dropout is 0.5, epoch is 30, and the number of LSTM hidden units is 128.

5.2.4. Bi-LSTM+Attention. The semantic extraction layer uses a Bi-LSTM network with attention mechanism. The learning rate is 0.001, batch size is 128, Adam optimizer, dropout is 0.5, epoch is 30, and the number of LSTM hidden units is 128.

5.2.5. Bi-DLSTM. The semantic extraction layer uses a Bi-DLSTM network. The learning rate is 0.001, batch size is 128, Adam optimizer, dropout is 0.5, epoch is 30, the number of LSTM hidden units is 128, and the dilation is 2.

5.2.6. Bi-DLSTM+Attention. The semantic extraction layer uses a Bi-DLSTM network with attention mechanism. The learning rate is 0.001, batch size is 128, Adam optimizer, dropout is 0.5, epoch is 30, the number of LSTM hidden units is 128, and the dilation is 2.

5.3. Sentiment Analysis Experiment. In order to conduct a better comparison experiment, the texts marked as happy and favorite emotions in the three Chinese pop songs sentiment datasets were selected as positive sentiment categories, and other sentiments were used as negative sentiment categories and compared with the self-built lyrics dataset to evaluate the performance of the proposed sentiment analysis model.

5.4. Evaluation Indicator. The experiment uses four basic sentiment analysis evaluation indicators for evaluation: accuracy, precision, recall, and F1-score.

6. Results

As shown in Figure 2 and Table 1, the classification result of models without skip connections and attention mechanism is low. Among them, the CNN network has the lowest classification performance of all models, and the network model combining CNN and LSTM has achieved better classification results. It is experimental indicators that are equivalent to the performance of the two-way LSTM network. It is not difficult to see from the experimental data that the LSTM network with expanded jump connections shows better classification performance in sentiment analysis. After introducing a bidirectional Dilated LSTM network layer, the Bi-LSTM and Bi-LSTM+attention models have achieved about 5% improvement in classification accuracy. Similarly, the Bi-LSTM network has an accuracy increase of about 5% after the introduction of the attention mechanism, and the Bi-DLSTM network has an accuracy increase of about 3% after the introduction of the attention mechanism.

Among the accuracy, recall, and F1-score trained in the self-built lyrics dataset, the model proposed in this paper is also better than the other five comparison models. After the Bi-LSTM network introduces the two-way Dilated LSTM network layer, the three evaluation indicators are increased by 6%, 7.7%, and 6.5%, respectively. After the Bi-LSTM +attention network introduces the bidirectional Dilated LSTM network layer, the three evaluation indicators are increased by 5%, 4.8%, and 3.9%, respectively. At the same time, the Bi-LSTM network and the Bi-DLSTM network with the introduction of the attention mechanism have improved in all evaluation indicators. It fully shows that the Bi-DLSTM+attention network can improve the performance of long-sequence information extraction and sentiment analysis.

From Figure 3, we can tell that Bi-DLSTM+att has a better performance on all four datasets compared with all other models. On average, the results on Beijing Opera lyrics dataset are not as good as the outcome on the other three Chinese pop song lyrics dataset. One reason we suggest is that the context for Beijing Opera lyrics is different from Chinese pop song lyrics, and the other reason is that the sentence of Beijing Opera is shorter which may lead to less accuracy.

7. Conclusion

In this paper, we proposed a sentiment analysis model based on Bi-DLSTM network to solve the problem of semantic extraction and sentiment classification of long-sequence Beijing Opera lyrics, and construct a lyrics dataset suitable

for lyrics sentiment analysis tasks. In order to explore the sentiment analysis performance of the model in this chapter, a coarse-grained sentiment analysis experiment was carried out on four datasets including self-built lyrics datasets, and the evaluation indicators obtained by training are better than the other five baseline methods. It reflects the excellent performance of this model in semantic extraction and sentiment analysis of long-sequence texts.

Data Availability

The self-built dataset is collected from the website below: https://www.xikao.com/. The three comparative dataset (NLPCC2013/NLPCC2014/Chinese Pop Songs Emotion Corpus) is collected from the website below: https://www.mulanci.org/.

Conflicts of Interest

There is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work is supported by the Beijing Outstanding Young Scientist Program (BJJWZYJH01201910048035), National Natural Science Foundation of China (61631016), Key Projects of National Key R D Program of China (2018YFB14039002), and Fundamental Research Funds for the Central Universities (CUC210B011).

References

[1] C. Strapparava and A. Valitutti, "Word net-affect: an affective extension of word net," *LREC*, vol. 4, pp. 1083–1086, 2004.

[2] H. Corona and M. P. O. Mahony, "An exploration of mood classification in the million songs dataset," in *12th Sound and Music Computing Conference*, Dublin, Ireland, 2015.

[3] B. Rocha, R. Panda, and R. P. Paiva, "Music Emotion Recognition: The Importance of Melodic Features," in *International Workshop on Machine Learning and Music (MML)*, 2013, http://hdl.handle.net/10316/95166.

[4] A. M. Schoene, "Dilated LSTM with attention for classification of suicide notes," in *Proceedings of the Tenth International Workshop on Health Text Mining and Information Analysis (LOUHI 2019)*, pp. 136–145, Hong Kong, 2019.

[5] Y. Bengio, R. Ducharme, and P. Vincent, "A neural probabilistic language model," *The Journal of Machine Learning Research*, vol. 3, pp. 1137–1155, 2003.

[6] T. Mikolov, W. Yih, and G. Zweig, "Linguistic regularities in continuous space word representations," in *Proceedings of the 2013 conference of the north american chapter of the association for computational linguistics: Human language technologies*, pp. 746–751, Atlanta, Georgia, 2013.

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[8] A. Graves, A. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 6645–6649, Vancouver, BC, Canada, 2013.

[9] J. Chung, C. Gulcehre, K. H. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," 2014, https://arxiv.org/abs/1412.3555.

[10] A. Rakhlin, *Convolutional Neural Networks for Sentence Classification*, GitHub, 2016.

[11] Q. Liu, H. Wu, Y. Ye, H. Zhao, C. Liu, and D. Du, "Patent litigation prediction: a convolutional tensor factorization approach," in *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence (IJCAI-18)*, pp. 5052–5059, 2018.

[12] W. Li, F. Qi, M. Tang, and Z. Yu, "Bidirectional LSTM with self-attention mechanism and multi-channel features for sentiment classification," *Neurocomputing*, vol. 387, pp. 63–77, 2020.

[13] Y. Dong, Y. Fu, L. Wang, Y. Chen, Y. Dong, and J. Li, "A sentiment analysis method of capsule network based on BiLSTM," *IEEE Access*, vol. 8, pp. 37014–37020, 2020.

[14] M. E. Basiri, S. Nemati, M. Abdar, E. Cambria, and U. R. Acharya, "ABCDM: an attention-based bidirectional CNN-RNN deep model for sentiment analysis," *Future Generation Computer Systems*, vol. 115, pp. 279–294, 2021.

[15] Z. Yang, D. Yang, C. Dyer, X. He, A. Smola, and E. Hovy, "Hierarchical attention networks for document classification," in *Proceedings of the 2016 conference of the North American chapter of the association for computational linguistics: human language technologies*, pp. 1480–1489, San Diego, California, 2016.

[16] Y. Wang, M. Huang, X. Zhu, and L. Zhao, "Attention-based LSTM for aspect-level sentiment classification," in *Proceedings of the 2016 conference on empirical methods in natural language processing*, pp. 606–615, 2016.

[17] Z. Lin, M. Feng, C. N. Santos et al., "A structured self-attentive sentence embedding," 2017, https://arxiv.org/abs/1703.03130.

[18] R. Mayer, R. Neumayer, and A. Rauber, "Combination of audio and lyrics features for genre classification in digital audio collections," in *Proceedings of the 16th ACM international conference on Multimedia*, pp. 159–168, Vancouver, BC, Canada, 2008a.

[19] R. Mayer and A. Rauber, "Music genre classification by ensembles of audio and lyrics features," in *Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011)*, pp. 675–680, Miami, Florida, USA, 2011.

[20] D. Bainbridge, S. J. Cunningham, and J. S. Downie, "How people describe their music information needs: a grounded theory analysis of music queries," in *Proceedings of the International Symposium on Music Information Retrieval (ISMIR 2003)*, pp. 221-222, Baltimore, Maryland, USA, 2003.

[21] M. Kim and H. C. Kwon, "Lyrics-based emotion classification using feature selection by partial syntactic analysis," in *2011 IEEE 23rd International Conference on Tools with Artificial Intelligence*, pp. 960–964, Boca Raton, FL, USA, 2011.

[22] R. Mihalcea and C. Strapparava, "Lyrics, music, and emotions," in *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pp. 590–599, Jeju Island, Korea, 2012.

[23] M. Fell and C. Sporleder, "Lyrics-based analysis and classification of music," in *Proceedings of COLING 2014, the 25th international conference on computational linguistics: Technical papers*, pp. 620–631, Dublin, Ireland, 2014.

[24] F. H. Rachman, R. Sarno, and C. Fatichah, "Music emotion classification based on lyrics-audio using corpus based emotion," *International Journal of Electrical and Computer Engineering*, vol. 8, no. 3, 2018.

[25] J. Turian, L. Ratinov, and Y. Bengio, "Word representations: a simple and general method for semi-supervised learning," in *Proceedings of the 48th annual meeting of the association for computational linguistics*, pp. 384–394, Uppsala, Sweden, 2010.

[26] A. M. Dai, C. Olah, and Q. V. Le, "Document embedding with paragraph vectors," 2015, https://arxiv.org/abs/1507.07998.

[27] S. Hochreiter and J. Schmidhuber, "LSTM can solve hard long time lag problems," in *Advances in Neural Information Processing Systems 9*, pp. 473–479, MIT Press, Cambridge MA, 1997.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 6014795, 10 pages
https://doi.org/10.1155/2022/6014795

Research Article

Multiple HD Screen-Based Virtual Studio System with Learned Mask-Free Portrait Harmonization

Jimao Jiang[1], **Jiaxin Lin**[1], **Yuntao Su**[1], **Li Fang**[1,2] and **Long Ye**[1,2]

[1]*School of Data Science and Media Intelligence, Communication University of China, Beijing 100024, China*
[2]*Key Laboratory of Media Audio & Video (Communication University of China), Ministry of Education, Beijing 100024, China*

Correspondence should be addressed to Li Fang; lifang8902@cuc.edu.cn

Received 2 October 2021; Accepted 10 December 2021; Published 2 May 2022

Academic Editor: Ivan Lee

Copyright © 2022 Jimao Jiang 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.

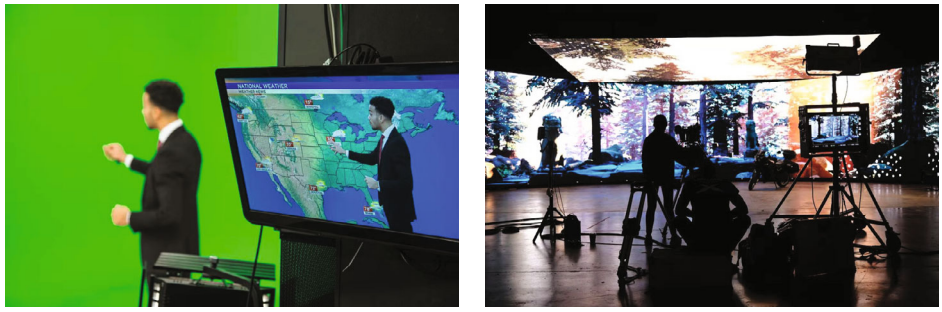
Virtual studio technology allows producers to combine live-action footage and computer-generated imagery and has promoted the development of film and television industry. However, the existing green screen-based virtual studio limits the visual effects just in postproduction. In this paper, we propose a new virtual studio system based on the recent emerging technology using multiple high-definition (HD) screens. Besides using traditional computer graphics like other similar systems, our system enables to capture panoramic video from the real world as the background and project it onto multiple HD screens, so as to recreate the scene in the studio. In addition, we propose a mask-free portrait harmonization network to make sure that the appearances of the foreground region and the background are consistent. Our portrait harmonization network does not need any auxiliary foreground mask and works in an end-to-end manner, which meets the postprocessing requirements for our virtual studio. Experimental result on real composite images shows that our network is superior to the state-of-the-art image harmonization method under the task of portrait harmonization. We also demonstrate our system with a CAVE system.

1. Introduction

Virtual studio technology based on chroma keying has set off a revolution in the film and television production industry in the past few decades. It allows the combination of people or other real objects and computer-generated environments and objects, which makes the whole studio no longer limited to traditional equipment such as site layout, lighting effects, and stage properties. Compared with the traditional studio, it requires less space but is more universal, providing producers with freer and broader creative space.

The main feature of a green screen-based studio is to film or photograph characters and objects against a single-color background (usually blue or green) and then synthesize a virtual background (shown in Figure 1(a)). However, this leads to the fact that characters cannot directly see the virtual elements in the studio, and therefore, the interaction with the virtual elements can only rely on imagination, which will inevitably cause problems such as the lack of the sense of presence of characters, resulting in unnatural program effect.

Recently, with the development of display technology, motion capture, and computer-generated imagery (CGI), multiple high-definition (HD) screens such as quality LED screens are being used as impressive replacements for traditional green screens, enabling producers to capture both live action and real-time rendered environments and objects in camera together, as shown in Figure 1(b). Furthermore, the real-time animation brought by the upgrade of the virtual engine is easy to use and scalable and gradually reduces the dependence on hardware. Nowadays, installation of HD screens in studios is becoming a trend for the filmmaking industry, and implementation of the technology will be more and more common. However, virtual production requires a 3D graphic artist and 3D computer graphics software to create the virtual background and any graphics that appear in front. For natural scenes with complex materials



(a) (b)

Figure 1: Different virtual studios. (a) Green screen-based virtual studio. The host cannot directly see the virtual elements. (b) Multiple HD screen-based virtual studio. The appearances of the foreground and the background are inconsistent.

and lighting, creating desired photorealistic models is costly. Applications that are sensitive to processing delays, such as live news broadcasting and remote face-to-face meeting, cannot use such method to rebuild the scene. Moreover, in order to create an illusion that characters and objects filmed are present in the intended background scene, the lighting in the two scenes must be a reasonable match. As can be seen in Figure 1(b), the appearances of the foreground are inconsistent with those of the background, which may cause unnatural imaging. This poses a challenge to the lighting of the studio.

In this paper, we propose a multiple HD screen-based virtual studio system that enables the use of shooting scenes from the real world as the background, as well as CGI. Specifically, our system takes panoramic videos as the background and utilizes multiple HD screens to recreate the scene in the studio. Characters in the studio seem to be on the spot, so as to get the best sense of presence. In addition, to simplify lighting requirements, we propose a deep portrait harmonization network that can make the appearances of the real part and the virtual part consistent. Unlike other image harmonization method, our network does not need to provide a foreground mask and works in an end-to-end manner, making it very suitable for our virtual studio. Experimental result shows that our network surpasses the state-of-the-art image harmonization method RainNet [1] without given any foreground mask. With the proposed virtual studio, the crew does not need to actually travel to particular locations for shooting. Lengthy and expensive location shoots can now be streamlined into one trip—capture the ideal light once then bring it back to the studio and recreate it as many times as needed in a controlled environment.

In summary, we developed a multiple HD screen-based virtual studio system. The main contributions of this paper are as follows:

(1) We developed a virtual studio system, which is complete and easy to use. Our system takes not only CGI but also panoramic videos as the background and thus provides a low-cost solution for high-fidelity scene recreation

(2) We proposed an end-to-end portrait harmonization method, which can directly process the video taken in the virtual studio without any additional foreground mask, so that the appearances of the characters in the virtual studio can be consistent with the virtual background

The rest of our paper is organized as follows. Section 2 comprehensively introduces the progress of virtual studio and some recent works of image harmonization. Section 3 presents our virtual studio system based on multiple HD screens. Section 4 presents the proposed deep portrait harmonization network for postprocessing. Section 5 is the experiments and detailed results. Section 6 concludes this paper.

2. Background and Related Work

2.1. Virtual Studio. Virtual production uses a suite of software tools to allow studios to combine live-action footage and computer graphics in real time. Digital environment can be created and rendered individually, while characters are physically working on set.

Nowadays, green screen is frequently used in most virtual studios. The BrainStorm company uses green screen combined with InfinitySet, featuring the patented TrackFree™ technology, which includes trackless and tracked camera or a combination of them [2]. CJP Broadcast also uses green screen to complete a 4K-UHD virtual studio system [3]. However, the utilization of green screen may lead to a sophisticated building process, time-consuming postprocessing, and poor sense of presence for characters.

Recently, there is an emerging trend on installation of HD screens instead of green screen. LED walls show characters what the set they are on looks like both through their eyes and in camera, so as to provide a better sense of presence. The Absen company uses advanced LED display combined with a virtual filming system, spatial positioning system, and real-time rendering system to achieve results of professional effect [4]. The Disney movie Mandalorian also chose LED display in filming [5].

2.2. Deep Learning Method for Image Harmonization. To make the synthesized output more realistic and harmonious is a major challenge in the virtual studio system, which has attracted the interest of many researchers. Traditional image

harmonization methods focus on matching the appearances of the foreground and the background using image statistics, such as color statistics [6], gradient-domain statistics [7], and semantic information [8].

With the success of deep learning in computer vision and computer graphics [9, 10], some deep learning-based image harmonization methods have been introduced. In [11], Tsai et al. proposed an end-to-end deep convolutional neural network, which can capture both the context and semantic information of the composite image during image harmonization. In [12], Cun and Pun proposed a novel attention module called Spatial-Separated Attention Module (S2AM) and combined it with the Unet [13] backbone network. In [14], Cong et al. proposed a deep image harmonization method DoveNet using a novel domain verification discriminator.

Among all these methods, RainNet [1], with the ability to make the composite image style-consistent and more realistic by learning style information from the background and adjust the style of the foreground, provides the state-of-the-art performance.

3. Proposed Virtual Studio System Based on Multiple HD Screens with Panoramic Video

To rebuild a scene, instead of 3D computer graphics made by 3D graphic artists, our virtual studio system uses panoramic videos or images shot from real scenes and projects them onto multiple screens. When shooting in the studio, both characters on set and virtual background on screens are captured by a TV camera, and the recorded video is then post processed by the portrait harmonization network resulting to the final output video. Our system is implemented using Unreal Engine 4 (UE4), and its framework is illustrated in Figure 2. In our implementation, we use an Insta360 Pro panoramic camera to capture panoramic videos. Besides, our system also includes necessary tools, such as camera tracking and motion capture, which enable characters to interact with the virtual environment. Please note that even though the major goal of our system is to allow the use of real shots as the background, traditional CGI works as well, and they can also be used together for more complex purposes.

3.1. Panoramic Video Segmentation and Projection of the Panorama onto Multiple HD Screens. The panoramic video is a type of video that breaks through the two-dimensional plane and can record everything happening in a 360-degree sphere with the panoramic camera in the center. As for the shooting of the panoramic video, we use an Insta360 Pro panoramic camera consisting of six fisheye lenses.

Since the panoramic video is spherical and cannot be played directly with planar screens, it needs to be projected first. Assuming that the panoramic video is on a sphere, the projection of the spherical panoramic video to a planar format depends on a one-to-one pixel correspondence relationship between the panorama and each screen. Specifically, take the cubic screen layout as an example, as demonstrated in Figure 3, we first find the center of the sphere of the panorama and the center of the cube, and let them coincide. Next, we, respectively, denote the 6 faces of the cube as $X+$, $X-$, $Y+$, $Y-$, $Z+$, and $Z-$, according to their intersecting coordinate axes. Then, take face $Y+$ as an example, for each point (ρ, θ, φ) on the sphere; we calculate the intersection of the line passing through it and the origin with the cube through equation (1) under the constraint of equation (2).

$$\begin{cases} x = \dfrac{l}{2} \bullet \cot\theta, \\ y = \dfrac{l}{2}, \\ z = \dfrac{l \bullet \cot\varphi}{2\sin\theta}, \end{cases} \tag{1}$$

$$\left\{ \rho, \theta, \varphi | \rho = d, \theta \in \left(\cot^{-1}\left(\frac{w}{l}\right), \tan^{-1}\left(\frac{w}{l}\right) + \frac{\pi}{2} \right), \varphi \in \left(\cot^{-1}\left(\frac{h}{l}\right), \tan^{-1}\left(\frac{h}{l}\right) + \frac{\pi}{2} \right) \right\}, \tag{2}$$

where l, w, and h are the length, width, and height of the cube, respectively, and d is the radius of the sphere. And the resulting (x, y, z) is the coordinate of the corresponding point on the cube of the point (ρ, θ, φ) on the sphere. The projection of the remaining faces is calculated similarly, and finally, we get the projection of the spherical panoramic video.

In our implementation, the above projection process is realized in UE4 through nDisplay. Specifically, we create a sphere model, and the panoramic video is applied to the inner surface of the sphere as materials. Then, the panoramic video is rendered and displayed on multiple screens. The detailed steps are described as follows.

Firstly, create a media player and select creating "media texture" asset which connects to the media player. Then, select media texture and create material. Next, set the material in a detail panel, and establish a blueprint to perform UV coordinate transformation (see Figure 4(a)). The following step is to drag a sphere into the current level and set its size and position. After dragging the material into the sphere material and creating a blueprint (see Figure 4(b)) in the level blueprint, run it and we can get the panoramic video display in UE4.

Note that the size and position of the sphere matter. Smaller sizes lead to fisheye distortion, while larger sizes lead to an unclear background. Therefore, in our system, we empirically set the size of the sphere to 1000 times the original size and set the center of the sphere to the origin.

The panoramic video is then displayed on multiple screens using nDisplay. Figure 5 shows how nDisplay works with multiple HD screens in the network. The host node PC is considered the conductor of nDisplay. It is used for centralized processing of the following: managing data in a synchronous manner and dispatching data to other cluster node PCs in the nDisplay cluster network, such as data inputs, camera tracking data, and custom cluster events, ensuring that the same data and the same inputs are received at the same time. The nDisplay Launcher and nDisplay Listener are used to control instances of UE4 across different computers on a network, each connected to one or many displays [15].

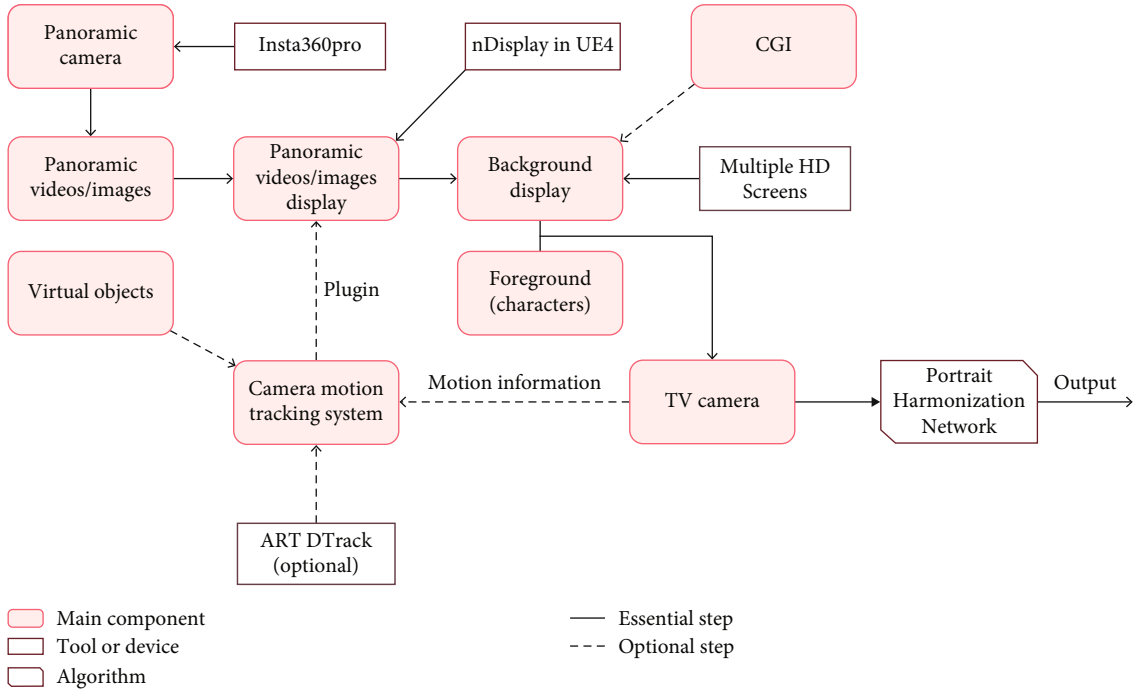


Figure 2: The framework of our virtual studio system.

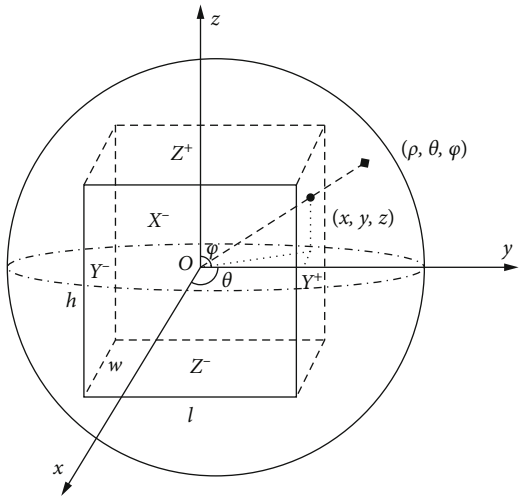


Figure 3: One-to-one pixel correspondence relationship on the panorama and multiple HD screens.

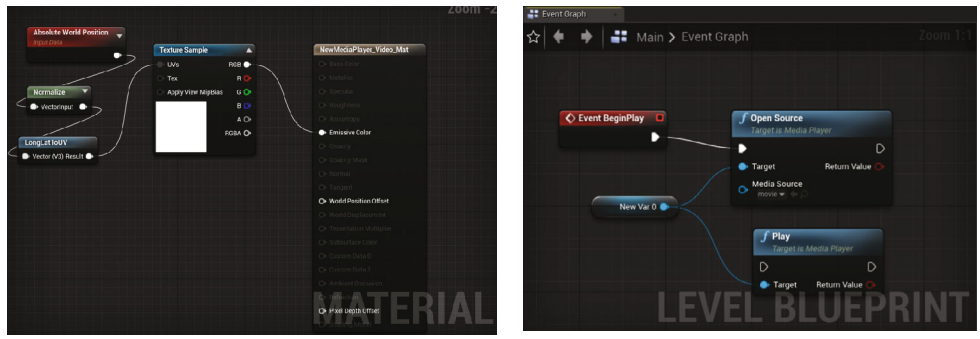
The nDisplay toolset consists of a plugin, a set of configuration files, and applications developed for UE4. The configuration files describe the topology of the display system and overall centralized location for project setting. It is worth noting that the configuration file needs to be modified to be in accordance with the size, resolution, node structure, virtual camera position, and other parameters of the multiple screens and related computers. In general, there is no need to modify the configuration file after confirming the correct operation unless the network topology changes. These are the exceptions: when the physical arrangement of the screens is changed or the computers which we are rendering to need to be changed [15].

With nDisplay Listener on every PC started and the projector turned on, add the appropriate configuration file to "config files" in nDisplay Launcher and run the corresponding EXE file; the content of UE4 can then be played on multiple HD screens. Therefore, when we are in the virtual studio, it is just like we are in the real world.

3.2. Tracking with the ART System. To track objects in the virtual studio system in real time, in our implementation, the ART tracking system, which is an infrared (IR) optical tracking system, is adopted. Tracking means the process of determining the position of a moving object in space. In order to track, these objects need to be equipped with individual marking points or rigid body. The spatial coordinates (X, Y, Z) of the rigid body are called 3DOF, and the measurement process of position and direction parameters is called 6DOF tracking [16].

Individual markers can only meet the 3DOF coordinate requirements, but if 6DOF coordinates are needed, the rigid body is required. The ART system mainly consists of four TRACKPACK/E cameras, four camera cables for power supply and sync, one external sync input cable, the atc-301604008 controllers of ART, several marking points, and DTrack2 software [16].

The principle of infrared optical tracking is shown as follows. The infrared transmitter cameras installed above HD screens send out synchronized IR flashes, and the marking points covered by the retroreflective material could then reflect the infrared light towards the lens. Intelligent tracking cameras create a greyscale image after detecting the reflected IR radiation. Therefore, the cameras can calculate 2D coordinates of the marking points based on pattern recognition. Then, the 2D data are sent to the controller via Ethernet. To get 6DOF data, DTrack2 calculates the position coordinates



(a) (b)

Figure 4: Blueprints in Unreal Engine 4: (a) material blueprint; (b) level blueprint.

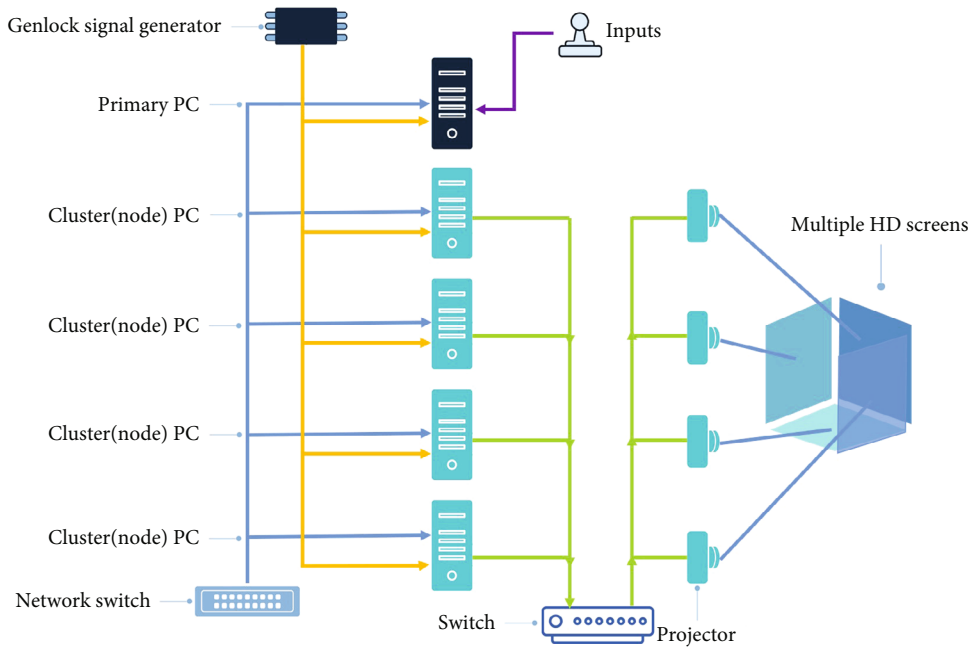


Figure 5: Network setup for multiple HD screens.

of the tracking target and determines the rigid body shape by the position distribution of the marking points. And the principle of optical tracking is shown in Figure 6 [16].

In optical tracking systems, it is necessary that the position of the object to be tracked is limited in the tracking range of the cameras, and the target should not be occluded by other objects because it will lead to problems such as inaccurate tracking.

Before tracking in the virtual studio, DTrack2 should be started. All necessary calculations (3DOF and 6DOF data) are done by the controller. The data and control commands are interchanged via a TCP/IP connection between the controller and the DTrack2 frontend software on the remote PC [16].

In advance, the orientation of each camera should be adjusted and a room calibration should be carried out. After starting the software, select the desired output data in the output settings panel. In UE4, just download the Dtrack-plugin previously and create a connection to Dtrack service source; then, real-time tracking can be achieved by adding components "Live Link Controller" to the target you want to track. As the marker moves, the tracked object on HD screen panels moves with it.

In the virtual studio, characters can better interact with the background through DTrack2. If we add a virtual object onto the panorama in nDisplay and add a live link to this object, the character can control the object using Flystick, which is similar to gamepad used in DTrack2. For example, if the character is performing in the virtual studio (such as football match commentary), he or she can use Flystick to pull up and zoom in/out on the scoreboard.

3.3. Capturing Video with Both the Foreground and Background on Multiple HD Screens. When the panoramic video is played on multiple HD screens and forms the background, the characters can perform in the studio. Now we focus on the shooting part. We should make sure that the

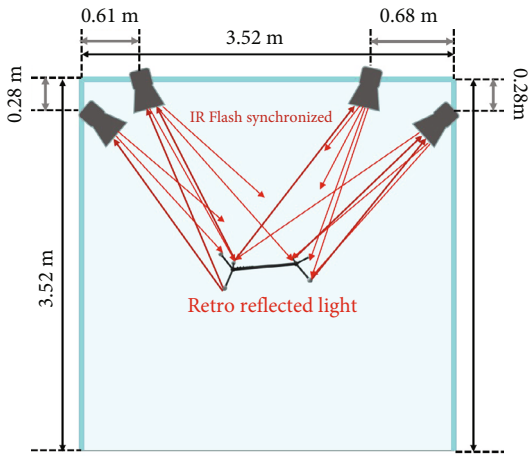


Figure 6: Principle of optical tracking.

camera center is located at the center of the sphere, so that there is no geometric distortion in the output video. To achieve this, we track the camera with the ART system and move it until the two centers coincide.

With the appropriate shooting method, the output of the TV camera combines both the foreground and background. However, due to various factors, the foreground and the background look inharmonious. We will discuss the solution in the next section.

4. Postprocessing Using Portrait Harmonization

4.1. Problem Formulation. In the virtual studio, the TV camera captures both characters on set and background displayed on the HD screens. For the sake of simplicity, we ignore the influence of the imaging process. Let us consider only one frame. Given the foreground image I_f and the background image I_b, this process can be simply regarded as the superposition of two images:

$$I = \alpha \circ I_f + (1 - \alpha) \circ I_b, \tag{3}$$

where α is the alpha values of the respective pixels in the foreground region, "$\circ$" denotes the Hadamard product, and I is the image captured by the TV camera.

However, due to the difference between the shooting conditions of the studio and the scene where the background is collected, such as lighting and exposure, the appearances of the foreground and background are inevitably incompatible. Intuitively, this problem can be solved by applying image harmonization on captured image I. As described in Section 2.2, many image harmonization methods based on deep learning are presented and can achieve impressive results. But to our knowledge, all these methods require a foreground mask as an auxiliary input, making them inappropriate for our virtual studio.

4.2. Architecture of the Proposed Method. Our goal is to make the appearances of the foreground region and the background region consistent, without giving a mask. Intuitively, we can generate a foreground mask and utilize any effective image harmonization network. In our case, the foreground is usually human. Therefore, we propose a novel deep neural network that consists of a portrait matting module and an image harmonization module. Figure 7 demonstrates the architecture of the proposed mask-free portrait harmonization network.

For the portrait matting module, we borrow the network from the state-of-the-art work MODNet [17]. This network consists of 3 interdependent branches, including low-resolution semantic estimation branch S to predict coarse semantic mask s_p, high-resolution detail prediction branch D to calculate the boundary detail matte d_p while considering the dependency between semantics and original images, and semantic-detail fusion branch F to combine semantics and details and to get α_p. Our loss function consists of 3 parts:

$$\mathscr{L}_m = \lambda_s \mathscr{L}_s + \lambda_d \mathscr{L}_d + \lambda_\alpha \mathscr{L}_\alpha, \tag{4}$$

where $\mathscr{L}_s$ is the $L2$ loss of s_p and thumbnail of the ground truth matte $G(\alpha_g)$, i.e., $\mathscr{L}_s = 1/2\|s_p - G(\alpha_g)\|_2$; $\mathscr{L}_d$ is the $L1$ loss of d_p and the ground truth mask, i.e., $\mathscr{L}_d = \|d_p - \alpha_g\|_1$; $\mathscr{L}_\alpha$ is the $L1$ loss of α_p and the ground truth mask plus the compositional loss $\mathscr{L}_c$ from DIM [18], i.e., $\mathscr{L}_\alpha = \|\alpha_p - \alpha_g\|_1 + \mathscr{L}_c$; $\lambda_s, \lambda_d, \lambda_\alpha$ are hyperparameters balancing the three losses. We set $\lambda_s = \lambda_\alpha = 1$ and $\lambda_d = 10$. It is worth noting that we do not use the trimap-dependent binary mask proposed in the original architecture of MODNet, because our training does not require the image trimap. Details of training will be mentioned in Section 4.3.

Finally, we get the mask α of the characters in the input image.

For the image harmonization module, the network from the state-of-the-art work RainNet [1] is used. This network takes a simple U-Net [13, 19] alike network without any feature normalization layers as basic architecture. The baseline also adds three attention blocks in the decoder part. Besides, RainNet has designed a delicate RAIN module that can be applied in any layers of the basic network.

As the foreground mask α is obtained, it is then concatenated with the input image along channel dimensions and sent into the RAIN module. Let $I_f^i \in \mathbb{R}^{H^i \times W^i \times C^i}$ be the activations, where H^i, W^i, C^i denote the height, width, and the number of channels of the input feature map, respectively. Because we want to match the foreground with the background, referring to the method of RainNet [1], the new activation value $I_f^{i\prime}$ at site (h, w, c) in the foreground region is computed by

$$I'_{f(h,w,c)} = \gamma_c^i \frac{I^i_{f(h,w,c)} - \mu_c^i}{\sigma_c^i} + \beta_c^i, \tag{5}$$

where μ_c^i and σ_c^i are the channel-wise mean and variance of the foreground feature and γ_c^i and β_c^i are the mean and standard deviation of the activations of the background in channel c of layer i. γ_c^i and β_c^i represent the statistical style only from

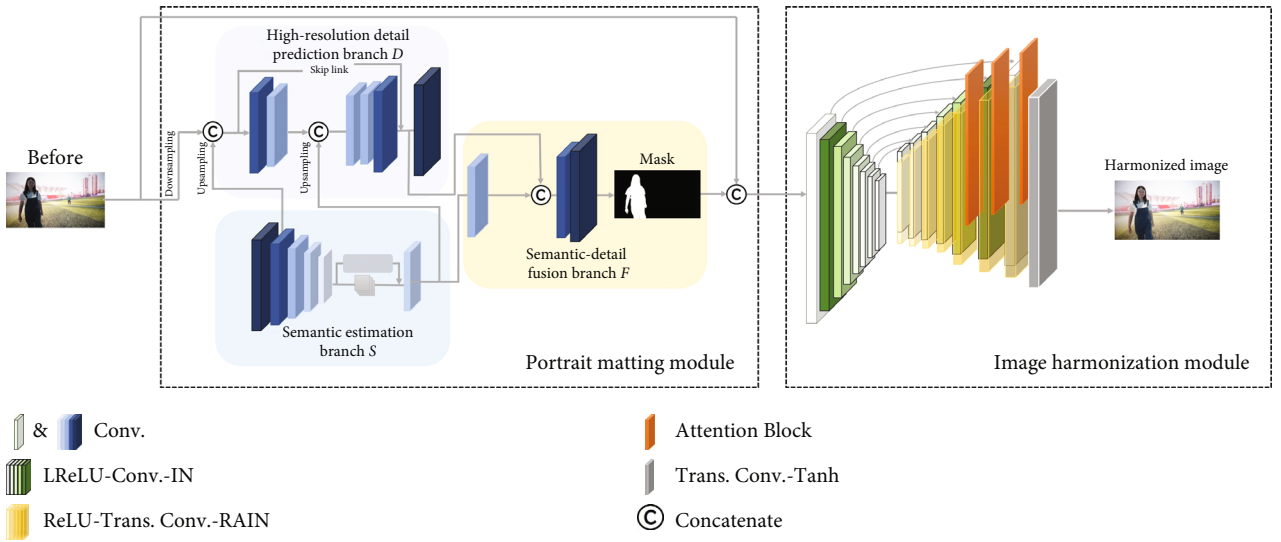


Figure 7: Architecture of the proposed portrait harmonization network. It consists of 2 modules: the portrait matting module and the image harmonization module. When a composite image is input, it can be directly harmonized into a consistent image. The bottom legend: Conv. = Convolution; Trans. = Transposed.

(a) (b)

Figure 8: Implementation of our virtual studio system with CAVE: (a) workstation and multiple screens; (b) shooting environment in CAVE.

(a) (b)

Figure 9: Portrait harmonization example: (a) original image; (b) harmonized image with our proposed method.

background features, and we normalize the foreground features to get the harmonized results accordingly.

As for the loss function of this module, we adopt the adversarial loss used in [1]. Specifically, we define the harmonization model as G and the harmonized image by $I' = G(I, \alpha)$. We optimize the results by calculating the $L1$ loss of I' and the ground truth I_g, i.e., $\mathscr{L}_{\text{rec}}(G, I_g, I, \alpha) = \|G(I, \alpha) - I_g\|_1$. And the additional adversarial loss is calculated following the training strategy in [14]. For the training of our network, we follow the setting of hyperparameters in [1]. As for the full objective,

$$\begin{aligned}\mathscr{L}\left(D, D_v, I_g, I', \alpha\right) &= \lambda_1 \mathscr{L}_{\text{adv}}\left(D, I_g, I'\right) + \lambda_2 \mathscr{L}_v\left(D_v, I_g, I', \alpha\right),\\ \mathscr{L}\left(G, I_g, I, \alpha\right) &= \lambda_1 \mathscr{L}_{\text{adv}}(G, I, \alpha) + \lambda_2 \mathscr{L}_v(G, I, \alpha) + \lambda_3 \mathscr{L}_{\text{rec}}\left(G, I_g, I, \alpha\right),\end{aligned} \tag{6}$$

Table 1: Quantitative comparisons (PSNR/SSIM) of different methods over the PPM-100 dataset. Four images we show in Figure 10 were selected as examples to show individual results. The average values of PSNR/SSIM over the whole PPM-100 dataset are shown in the table as well.

Example	MODNet+RainNet		RainNet with GT alpha matte		Ours	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
(a)	19.32	0.822	27.68	0.865	26.62	0.869
(b)	19.37	0.848	20.95	0.861	27.00	0.894
(c)	20.51	0.792	25.96	0.852	29.58	0.863
(d)	17.74	0.823	20.45	0.863	27.22	0.904
Avg. over the whole PPM-100 dataset	24.59	0.861	27.03	0.880	28.46	0.886

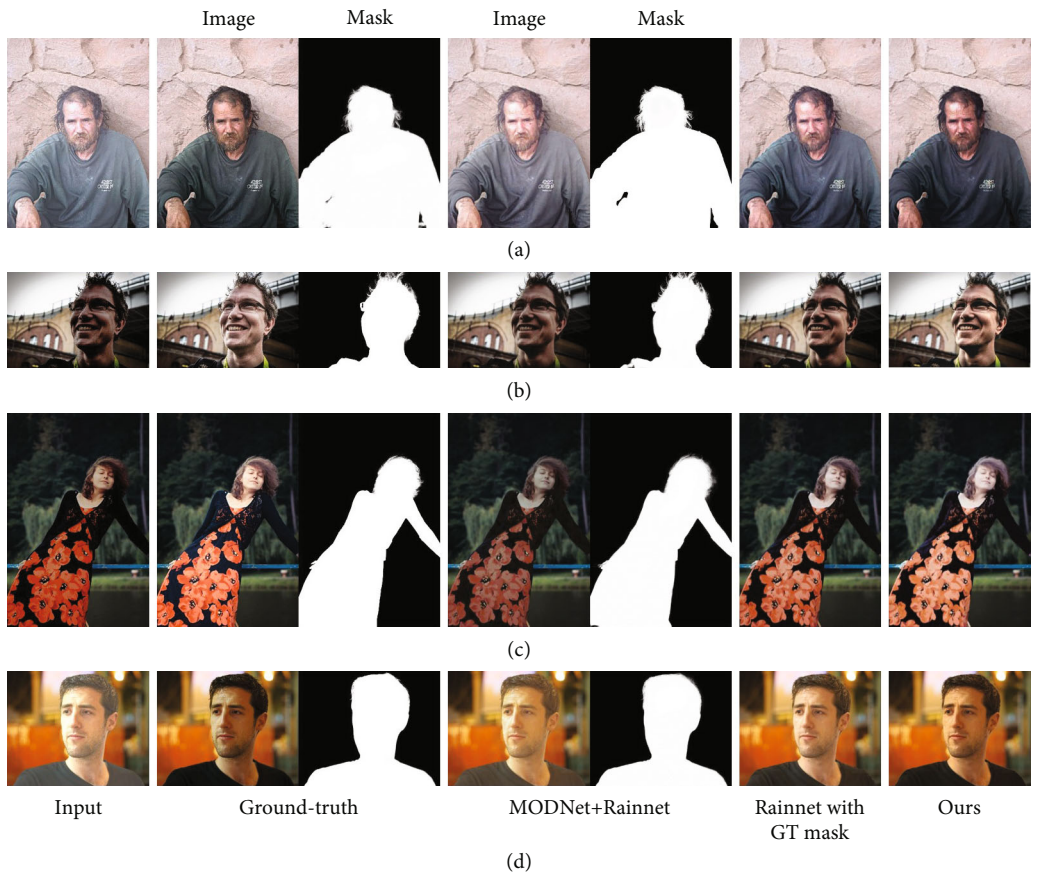


Figure 10: Visualization of comparisons on four randomly picked examples. From left to right: input images, ground truth images and their foreground masks, and harmonized results of MODNet+RainNet, RainNet with ground truth mask, and our methods.

where $\mathscr{L}_{\text{adv}}(D, I_g, I')$ and $\mathscr{L}_{\text{adv}}(G, I, \alpha)$ are adversarial losses and $\mathscr{L}_v(D_v, I_g, I', \alpha)$ and $\mathscr{L}_v(G, I, \alpha)$ are domain verification losses; we all follow the training strategy in [14] and set $\lambda_1 = \lambda_2 = 1$ and $\lambda_3 = 100$.

In our design, the foreground mask is generated and used implicitly. Therefore, better results can be obtained through global optimization, which will be verified in Section 5.

4.3. Implementation and Training Details. Since both the portrait matting module and the image harmonization module are differentiable, the proposed portrait harmonization network can be learned through end-to-end training with backpropagation.

Common datasets used for image harmonization (such as iHarmony4 [14]) include more than portrait images. And most of them do not contain enough good portrait masks. Therefore, we used a new dataset based on the Baidu people segmentation dataset [20] for end-to-end learning. This new dataset is composed of more than 5000 portrait images, corresponding masks, and composite images. This dataset guarantees sample diversity well, for example, (1)

different scenes, such as full-body images and half-body images; (2) diverse background, such as blurred background and dim background; and (3) various postures, such as holding objects and wearing hats. But the Baidu people segmentation dataset only includes natural images and the corresponding mask and lacks images with inharmonious foreground and background. Therefore, we preprocessed this dataset to adapt the portrait harmonization task. To get inharmonious images, we extracted the foreground from the original portrait image using the mask and randomly adjusted the brightness, color temperature, and other parameters of the foreground within a reasonable range and finally superposed the modified foreground over the original background.

For both modules, we first loaded the pretrained models provided by the authors of RainNet and MODNet and then trained it on our dataset with input images resized to 256×256. We used the Adam optimizer for the image harmonization module with a learning rate of 0.00001 and betas set to $(0.5,0.999)$ and the SGD optimizer for portrait matting module with a learning rate of 0.00001 and momentum set to 0.9. Our model was optimized for 75 epochs on an Nvidia GV100GL GPU, with the batch size set to 16.

5. Experimental Results

5.1. Demonstration of the Proposed System with CAVE. Our virtual studio system was implemented and demonstrated using the Cave Automatic Virtual Environment (CAVE) system [21]. Besides, the screens in CAVE can synchronously play video in real time, so characters can see the variation directly, which is beyond the reach of traditional green screen studio. This CAVE consists of four projection hard screens. Although the display effect is not as good as advanced LED walls, it is enough to show our virtual studio system. It is worth noting that our system is applicable to any multiple-screen display device, and we chose CAVE because we only have CAVE.

Our CAVE system consists of four screens—the left, right, front, and floor screens. The four projection surfaces form a cube, in which three adopt the way of the rear projection and the floor takes front projection. The screen of CAVE is 3.52 m long, 3.52 m wide, and 2.20 m high. Meanwhile, the resolution of computer screens is 1920×1200, which can properly suit CAVE.

The workstation contains five computers with 32.0 G RAM and Intel (R) Xeon (R) CPU at 2.50 GHz, one major node that controls the whole system, and four computers connected to the four screens of CAVE, respectively, as shown in Figure 8(a).

In order to display the panoramic video properly in CAVE, we created an nDisplay project in UE4, where we created a sphere and pasted the panoramic video onto the sphere as depicted in Section 3.1. After rendering and adjusting following the steps in Sections 3.2-3.3, the panoramic video was successfully played on CAVE, as shown in Figure 8(a).

When the panoramic video is playing, the characters can step into CAVE and perform. In the meantime, the TV camera captures video containing both the foreground and background. The camera we chose for recording is SONY ILCE-7RM3. Figure 8(b) shows the actual shooting environment in CAVE, and Figure 9(a) gives an example.

It can be seen from Figure 9(a) that the appearances of the character and the background are inharmonious. We can find that characters and the background fuse well, greatly reducing the feelings of separation compared with the green screen studio. But due to the limitation of projection brightness, the illumination of on-site characters does not match the background environment, as well as other features such as color temperature. Although this can be alleviated by appropriate lighting, it is costly. In our system, the inharmonious video frames were then postprocessed by the proposed portrait harmonization network. The foreground is obviously more compatible with the background, and the brightness and color temperature of the foreground and background are more consistent, making the image look more realistic, as shown in Figure 9(b). Our postprocessing with portrait harmonization can achieve a good result without additional hardware. Therefore, our system is very practical.

5.2. Portrait Harmonization Results. In this section, we further carried out experiments to verify our proposed portrait harmonization method and compared it with the state-of-the-art image harmonization method RainNet [1]. Since RainNet was not specifically trained for portrait harmonization, the pretrained model provided by the authors was carefully fine-tuned using our dataset and the suggested training configurations by the authors for fair comparisons.

For testing, we used the PPM-100 dataset, which is provided in MODNet paper [17] with many advantages as a validation set, such as fine annotation, natural background, rich diversity, and high resolution. So we argue that PPM-100 is a more comprehensive benchmark to verify the ability to deal with different scenes and different characters. The PPM-100 dataset was processed in the same way as the training set in advance.

Since RainNet requires a foreground mask, we carried out two different combinations:

MODNet+RainNet. We used MODNet with the pretrained model given by the authors to generate the foreground masks and provided them and corresponding input images to RainNet.

RainNet with GT mask. We directly used the ground truth foreground masks and corresponding input images as inputs of RainNet.

To quantitatively validate our postprocessing method, we followed the evaluation protocols from [1, 14]. We evaluated both methods on the PPM-100 dataset by measuring PSNR and SSIM for the harmonized images. Table 1 gives the results. Our average PSNR (28.46 dB) and SSIM (0.89) are much higher than other two methods, with PSNR increased by 15.74% and 5.29%, respectively, and SSIM increased by 2.90% and 0.68%, respectively.

We also visually compared the harmonized results of different algorithms on four randomly picked images, as shown in Figure 10. It can be seen from Figure 10 that our method performs well without foreground masks, proving the superiority of our method.

6. Conclusion

In this paper, we have presented a new multiple HD screen-based virtual studio system. It can take panoramic videos as the background and employ multiple HD screens to reconstruct an immersive environment. The system also includes methods such as panorama capturing, motion tracking, and shooting, providing a complete set of solutions for virtual production. We have further presented an end-to-end portrait harmonization network that can ensure that the appearances of the foreground region and the background are consistent, making the final video realistic. Experimental result on real composite images shows that the proposed network outperforms the previous state-of-the-art method.

Our current solution exploits panoramic video as the background, which lacks motion parallax and correct-in-all-directions disparity cues. Therefore, when using a panoramic video background, the camera position must be fixed. There is also an emerging trend on recording scenes with the dynamic light field or radiance field. In the future, we will try to introduce these 6DOF formats into our system to better promote the promotion of virtual production technology in the film and television production industry.

Data Availability

The portrait harmonization data supporting these findings of this study are from previously reported studies and datasets, which have been cited. The processed data are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work is supported by the National Key R&D Program of China (Grant No. SQ2020YFF0426386), the National Natural Science Foundation of China (Grant Nos. 62001432 and 61971383), the Major Science and Technology Project of Beijing (Grant No. z201100001820024), and the Fundamental Research Funds for the Central Universities (Grant Nos. CUC19ZD006 and CUC21GZ007).

References

[1] J. Ling, H. Xue, L. Song, R. Xie, and X. Gu, "Region-aware adaptive instance normalization for image harmonization," in *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9357–9366, Nashville, TN, USA, 2021.

[2] "Virtual studios|Brainstorm Index," 2021, https://www.brainstorm3d.com/solutions/virtual-studios.

[3] "CJP Broadcast Index," 2021, https://www.avinteractive.com/markets/media/sunderland-university-completes-led-ready-4k-uhd-virtual-studio-26-08-2021/.

[4] "Absen Virtual Studio LED Solutions Index," 2021, https://www.absen.com/virtual-studio/.

[5] "Insider," 2020, https://www.insider.com/green-screen-virtual-sets-mandalorian-2020-4.

[6] S. Xue, A. Agarwala, J. Dorsey, and H. Rushmeier, "Understanding and improving the realism of image composites," *ACM Transactions on Graphics*, vol. 31, no. 4, pp. 1–10, 2012.

[7] M. W. Tao, M. K. Johnson, and S. Paris, "Error-tolerant image compositing," *International Journal of Computer Vision*, vol. 103, no. 2, pp. 178–189, 2013.

[8] Y.-H. Tsai, X. Shen, Z. Lin, K. Sunkavalli, and M.-H. Yang, "Sky is not the limit," *ACM Transactions on Graphics*, vol. 35, no. 4, pp. 1–11, 2016.

[9] S. Yang, J. Wang, S. Arif, M. Jia, and S. Zhong, "SAL-net: self-supervised attribute learning for object recognition and segmentation," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 2891303, 13 pages, 2021.

[10] M. Yan, Z. Li, X. Yu, and C. Jin, "An end-to-end deep learning network for 3D object detection from RGB-D data based on hough voting," *IEEE Access*, vol. 8, pp. 138810–138822, 2020.

[11] Y.-H. Tsai, X. Shen, Z. Li, K. Sunkavalli, X. Lu, and M.-H. Yang, "Deep image harmonization," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2799–2807, Honolulu, HI, USA, 2017.

[12] X. Cun and C.-M. Pun, "Improving the harmony of the composite image by spatial-separated attention module," *IEEE Transactions on Image Processing*, vol. 29, pp. 4759–4771, 2020.

[13] O. Ronneberger, P. Fischer, and T. Brox, "Unet: convolutional networks for biomedical image segmentation," in *Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. MICCAI 2015*, N. Navab, J. Hornegger, W. Wells, and A. Frangi, Eds., vol. 9351 of Lecture Notes in Computer Science, pp. 234–241, Springer, Cham, 2015.

[14] W. Cong, J. Zhang, L. Niu, Z. Ling, W. Li, and L. Zhang, "DoveNet: deep image harmonization via domain verification," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8391–8400, Seattle, WA, USA, 2020.

[15] "Unreal Engine," 2021, https://www.unrealengine.com/.

[16] "Advanced Realtime Tracking," 2021, https://ar-tracking.com/en/product-program/dtrack.

[17] Z. Ke, K. Li, Y. Zhou et al., "Is a green screen really necessary for real-time portrait matting?," 2020, https://arxiv.org/abs/2011.11961.

[18] N. Xu, B. Price, S. Cohen, and T. Huang, "Deep image matting," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 311–320, Honolulu, HI, USA, 2017.

[19] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5967–5976, Honolulu, HI, USA, 2017.

[20] "Baidu people segmentation dataset," 2021, http://www.cbsr.ia.ac.cn/users/ynyu/dataset/.

[21] H. Creagh, "Cave automatic virtual environment," in *Proceedings: Electrical Insulation Conference and Electrical Manufacturing and Coil Winding Technology Conference (Cat. No.03CH37480)*, pp. 499–504, Indianapolis, IN, USA, 2003.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 5968047, 12 pages
https://doi.org/10.1155/2022/5968047

Research Article

A Renovated CNN-Based Model Enhances KGC Task Performance

Fang Miao,[1] **Xueting Wang**,[2] **Feng Feng**,[3] **Cong Jin**,[1] **and Libiao Jin**[1]

[1]*School of Information and Communication Engineering, Communication University of China, Beijing 10024, China*
[2]*China United Network Communications Corporation Qingdao Branch, Qingdao, China*
[3]*Department of Electrical Engineering and Computer Science, University of Missouri, Columbia, MO 65211, USA*

Correspondence should be addressed to Fang Miao; miaof03@cuc.edu.cn

Received 15 October 2021; Accepted 24 February 2022; Published 16 March 2022

Academic Editor: Bo Rong

Copyright © 2022 Fang Miao 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.

Knowledge graph (KG) contains a large number of real-world knowledge and has become an invaluable aid to assist the application of artificial intelligence. Knowledge graph completion (KGC) is the task to complete the missing triple in KG database. Our goal in this study is to enhance the performance of KGC tasks based on CNN model. To do this, we first investigated the effect of adding multiple filters of different shapes into the pioneer model. The obscure improvement leads us to seek other approaches. Our second proposed model, termed DP-ConvKB, which is a deep convolution-neural-network-based model, outperforms state-of-the-art models on several metrics. Our study provides supporting evidence that, by cooperating deep pyramid network structure into models, it can significantly improve the KGC performances.

1. Introduction

Since the development of the Semantic Web in the 1980s, knowledge graph (KG), as a concept of knowledge base, serves as a surrogate of the real-world information, which focuses on describing the relationships between interlinked entities. Based on the graph-structured data model of integrating data, KG uses a triple (head entity, relation, and tail entity) to store interlinked descriptions of entities. Thanks to its structure-based and machine-readable attributes, the application of KG proliferates in various domains, such as search, analytics, and recommendation. In 2012, search engine giant Google launched their KG [1], and by May 2020, Google KG had grown to 500 billion facts on 5 billion entities [2]. However, information incompleteness is a problem that always exists in all KGs, including massive KGs. For example, in Freebase and DBpedia, more than 66% of the person entities are missing a birthplace [3, 4]. These missing facts not only affect the KG structure itself but also would sometimes result in inferring misleading information. Therefore, knowledge graph completion (KGC) researchers aim to identify missing links under the supervision of the existing KG. The specific work scope of KGC includes finding and mining missing entities and relationships [5], link prediction [6], and inferring new facts [7].

The key problem of knowledge graph completion is how to represent and model the combination of entities and relations. Data representation and representation learning are known to play a pivotal role in the development of KGC [8]. Embedding method is a popular data representation technique, which generally converts the entities and relations into low-dimensional vectors or matrixes. For example, in TransE [9], the candidate triple (h, r, t) is used for representing a valid fact, and then, the relation r corresponds to a translation between the embeddings of the head entity h and the tail entity t, that is, $h + r \approx t$. Several succeeding transition-based models are proposed with the basic idea of TransE, such as TransH [10], TransD [11], TransR [12], and TransG [13]. Among them, TransH models a relation as a hyperplane and TransR divides the workspace into entity space and relation space during representing the relations and entities. In addition, a trilinear dot product has also been applied to compute the score for each triple; DistMult [14] and ComplEx [15] are two typical such linear models.

KGC models can be generally divided into embedding-based models, linear models, and neural network models. Neural networks (NNs) have been widely used in machine learning tasks such as pattern recognition and perception science [16] and also gained more attention in the field of KG recently. By cooperating with convolution algorithm, convolutional neural networks (CNNs) are a specialized type of NNs that have been heavily used in computer vision since its birth. Such popularity of CNN-based applications in computer vision is because the neural-network-based structure of CNNs can match the composition of images natively. Since Kim proposed TextCNN in 2013 [17], CNNs have also begun to receive numerous attentions in the field of natural language processing (NLP), such as sentence classification [18], sentence modeling [17], and search query retrieval [19]. Most of these models adopt the convolution layer similar to TextCNN to extract features in the embedding. Inspired by computer vision, Dettmers et al. proposed the first CNN-based model for the KGC task—ConvE [20]. Following ConvE, Nguyen et al. proposed the model ConvKB [21], where the triple (h, r, t) is represented as k-dimensional vectors V_h, V_r, V_t, and the input matrix combines these vectors into a $k \times 3$ matrix. In the convolution layer, different filters with the same shape of 1×3 are designed to explore the global features among the same dimensional units of the embedding triple (v_h, v_r, v_t).

Although the model ConvKB has overcome the limitation of ConvE and obtains better link prediction results than the existing models on several benchmark datasets, there are still several unsolved problems. For example, limited by the convolution kernel of fixed size, long-distance interaction between different positions in embedding vectors cannot be extracted, but only shallow information from the single convolutional layer. Therefore, multiple methods have been proposed to solve this problem, such as CapsE [22] and ConvR [23]. In this paper, we applied a deep pyramid convolutional network structure with the original ConvKB model and designed a new model named deep pyramid ConvKB (DP-ConvKB).

We summarize the main contribution of our work as follows.

To improve KGC task performance, we first attempted it with a tentative model by introducing multiple complex filters to the ConvKB, hereafter referred to as multifilter ConvKB (MF-ConvKB). Compared to other existing CNN-based models, including ConvKB, experiment results on two benchmark datasets, FB15k-237 and WN18RR, showed that MF-ConvKB could only bring about mild improvements on some specific metrics. However, it motivated us to seek for other possible model structures which might potentially help to overcome the limit of improvement. We found that by incorporating a deep pyramid network structure into ConvKB, the new-designed model DP-ConvKB could significantly improve the KGC performances on several metrics.

2. Related Work

2.1. Embedding-Based Model. Embedding-based TransE was the first model using transseries methods for modeling multirelational data [9]. Inspired by the Word2Vec Skip-gram model, the TransE maps the relationship in the triple to the translations in latent feature space, denoted as $h + r \approx t$. However, the TransE model can only work well on the instances with one-to-one relation due to its lower parameter complexity and fails to deal with complex relations, such as one-to-more, more-to-one, and more-to-more. To enhance the scope ability and the efficiency of the score function of the previous models, TransH [10] is proposed to model the complex relations as a hyperplane together with a translation operation on it; however, this model still lacks of adaptation to scenarios with multirelations. TransD [11] and TransR/CTransR [12] represent entities and relations in separate spaces and project each entity with a relation-specific matrix. Unlike TransR, STransE [24] and TranSparse [25] tie the head and tail entities together with their own projection matrices. In addition, PTransE [26] and PTransR [27] take the relation path information into consideration when representing the triple (h, r, t). Recently, aiming at expanding the embedding space, embedding-based models like RotatE [28] continue to play a big role in the KGC task field.

2.2. Linear Model. RESCAL [29] is a typical bilinear model obtaining the underlying semantics information of the representation of the entity, but this model is prone to overfitting because of the large amounts of parameters. DistMult [14] can be considered as a special case of RESCAL, in which DistMult represents each relation as a diagonal matrix rather than a full matrix to avoid overfitting. ComplEx [15] and SimplE [30] can be viewed as direct extensions of the DistMult, in which ComplEx applies the complex domain to handle a variety of binary relations, and in SimplE, the subject and object embeddings for the same entity are learned dependently.

2.3. Neural Network Model. ConvE [20] and ConvKB [21] both are CNN-based models, and ConvE is the first model with CNNs applied for the KGC task. In ConvE, various filters with the fixed shape are operated over the input matrix—reshaping and concatenating of the head entity and relation embeddings. The design of the input is straightforward and one sided, which ignores the global information among the same dimensional position of the representation of the whole triple. ConvKB was proposed to deal with the problem from ConvE, replacing the reshaping operation with a convolution layer over the embedding triple (h, r, t). CapsE [22] applies the capsule network for the KGC task by adding a capsule network layer instead of the convolution layer at the base of the ConvKB. SACN [31] and R-GCN [32] are graph convolutional network- (GCN-) based models, and DOLORES [33] introduces long short-term memory (LSTM) to KGC tasks. In addition, KBGAT [34], GAATs [35], ConvR [23], and CoPER-ConvE [36] are effective methods which are proposed in recent years. Collectively, incorporating NNs has surfaced to become the mainstream way to solve the problem of the incomplete knowledge graph. Table 1 illustrates the score function and the optimization methods of each related work.

Table 1: The score function and the optimization methods of each related work. In these models, the entities and relations are represented by vectors h, r, t. And in the CNN-based model, g denotes a nonlinear function, $*$ denotes a convolution operator, $\cdot$ denotes a dot product, and Ω denotes a set of filters.

Category	Model	Score function	Opt.
Embedding-based model	TransE	$\|h + r - t\|_{1/2}$	SGD
	TransH	$\|h_\perp + r - t_\perp\|_{1/2}$; $h_\perp = h - w_r^\top t w_r$; $t_\perp = t - w_r^\top t w_r$	SGD
	TransD	$\|h_\perp + r - t_\perp\|_{1/2}$; $h_\perp = M_r h$, $t_\perp = M_r t$; $M_r \in \mathbb{R}^{k\times d}$	AdaDelta
	TransR	$\|h_\perp + r - t_\perp\|_{1/2}$; $h_\perp = M_r^1 h$; $t_\perp = M_r^2 t$; $M_r^1 = w_r w_h^\top + I$; $M_r^2 = w_r w_t^\top + I$	SGD
	STransE	$\|W_{r,1} h + r - W_{r,2} t\|_{1/2}$; $W_{r,1}, W_{r,1} \in \mathbb{R}^{k\times k}$	SGD
	TranSparse	$\|h_\perp + r - t_\perp\|_{1/2}$; $h_\perp = M_r^1(\theta_r^1) h$; $t_\perp = M_r^2(\theta_r^2) t$	SGD
Linear model	RESCAL	$h_r^\top M_r t$; $M_r \in \mathbb{R}^{d\times d}$	SGD
	DistMult	$h_r^\top M_r t$; $M_r = \text{diag}(r)$; $r \in \mathbb{R}^d$	AdaDelta
	ComplEx	$\text{Re}\left(h^\top \text{ diag}(r)\bar{t}\right)$ Re (.): the real part of a complex value $\bar{t}$: the conjugate of t	AdaDelta
	SimplE	$\frac{1}{2}\left(h \odot r \cdot t + t \odot r^{-1} \cdot h\right)$ $\odot$ represents Hadamard multiplication $\cdot$represents dot product	AdaDelta
Neural network model	ConvE	$g\left(\text{vec}\left(g\left(\text{concat}\left[\bar{h}; \bar{r}\right] * \Omega\right)\right) W\right) \bullet t$	Adam
	ConvKB	$\text{concat}(g([h; r; t] * \Omega)) \bullet w$	Adam
	CapsE	$\|\text{capsnet}(g([h; r; t] * \Omega))\|$	Adam

Deep CNNs are commonly used to extract depth features from images, and a number of powerful models have emerged, such as LeNet [37], AlexNet [38], ResNet [39], and Google Inception Net [40]. Inspired by ResNet, Johnson and Zhang proposed a low-complexity word-level deep convolutional neural network architecture for text categorization—DPCNN [41]. In this paper, we proposed a novel neural network model, namely, DP-ConvKB, which takes the advantage of deep convolution structure, to improve the performance of KGC.

3. The Construction of Proposed Models

As we mentioned above, we attempted two different approaches to improve the KGC task performances by ConvKB. In the first model, MF-ConvKB, we incorporated multiple filters of different shapes into ConvKB. In the second model, DP-ConvKB, we utilized a deep pyramid convolutional network to improve the KGC task performance. In this section, we present the procedures for constructing these two models.

3.1. MF-ConvKB. An incomplete KG ζ collects the facts in the form of (h, r, t), with $h, t \in \mathscr{E}, r \in \mathscr{R}$, where $\mathscr{E}$ and $\mathscr{R}$ denote the sets of entities and relations, respectively. Each triple is represented as a unique embedding group (v_h, v_r, v_t) collecting with K-dimensional vectors, and all these vectors are concatenated into a matrix $A \in \mathbb{R}^{k\times 3}$ and $A_i \in \mathbb{R}^{1\times 3}$ represents the i-th row of A. For example, a filter $\omega \in \mathbb{R}^{h\times 3}$ is applied to the window consisting from A_i to A_{i+h-1} to generate a feature c_i, which is defined as follows:

$$c_i = g(\omega \bullet A_{i:i+h-1} + b), \tag{1}$$

where $b \in \mathbb{R}$ is a bias term and g is a nonlinear function such as rectified linear unit (ReLU), and we use $c = [c_1, c_2, \cdots, c_{k-h+1}]$ to form the feature map of this filter.

Different from the ConvKB that only uses filters of a single shape, our model involves multiple shapes of filters to generate the transitional features from the pretrained embeddings. As shown in Figure 1, multiple filters are designed to capture features from long or short distances. We used $N = [N_1, N_2, N_3]$ to denote the number of filters of three different shapes, which are 1×3, 2×3, and 3×3, so the original model ConvKB can be viewed as a special case of MF-ConvKB with $N_2 = N_3 = 0$. After convolution, all $\sum_{i=1}^{3} N_i$ feature maps are concatenated into a vector V. Then, the score for the triple (h, r, t) is calculated by a dot product between V and w, where $w \in \mathbb{R}^{\sum_{i=1}^{3} N_i \times 1}$ is a weight vector, and the score function f is defined as in

$$V_i = \text{concat}(g([v_h; v_r; v_t] * \Omega_i)), \tag{2}$$

$$f(h, r, t) = \text{concat}(V_1, V_2, V_3) \cdot w, \tag{3}$$

where Ω_i denotes the set of filters of the i-th shape, V_i is the feature map generated from the set of filters of the i-th

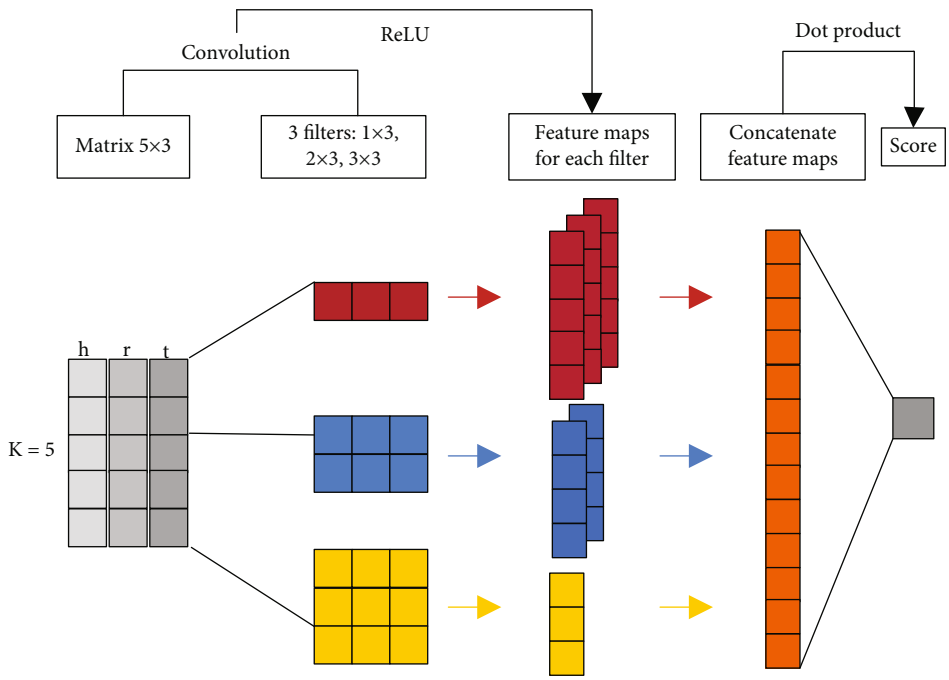


Figure 1: Illustration of the MF-ConvKB model framework.

shape, $*$ denotes a convolution operator, and $\cdot$ denotes a dot product.

3.2. DP-ConvKB. Compared to the tentative MF-ConvKB model, we implemented a completely different approach to extract features in DP-ConvKB. Instead of adding a variety of filters, we took advantage of the deep residual technique [39] and built another module of deep residual learning in addition to the pioneer ConvKB model. In this way, more features in a long distance can be taken into consideration. DP-ConvKB takes its name from referring its structure to deep pyramid (DP) CNNs [41]. As shown in Figure 2, the design of DP-ConvKB consists of two modules in general.

3.2.1. ConvKB Module. The left module is from the original ConvKB, where TransE model is used to initialize the input entities and relations. In order to keep the maximum potential of the transitional features in the transition-based model, we only applied the filter $w \in \mathbb{R}^{1\times 3}$ to exploit the global relationships among the same dimensional entries of the embedding triple. The feature map V_{ConvKB} is defined as

$$V_{\text{ConvKB}}(h, r, t) = \text{concat}(g([v_h; v_r; v_t]) * \Omega), \quad (4)$$

in which Ω denotes the set of filters and $*$ denotes a convolution operator. The output of the final feature map from this module is fed into the right module as the initial layer of region embedding.

3.2.2. Deep Residual Learning Module. In the deep residual learning module, the initial region embedding is followed by two convolutional layers, an activation layer, and a shortcut connection. This module with shortcut connection can be represented as $f(x) + x$, where f denotes the skipped layers of feedforward neural networks and x is the shortcut connection as proposed in ResNet [39]. A similar structure has also been applied in DPCNN for text categorization [41]. This design of the shortcut connection can effectively avoid the vanishing gradient problem when training the deep NNs. ReLU was chosen as the activation layer, which corresponds to the design of the feature map.

In this paper, we focused more on the global information among the same dimension between long distances, but different from the semantic features among a long sentence; in the KGC task, the input matrix is only composed of the embeddings of entities and relations, without the semantic characters between different dimensions. In order to capture the features with a relatively long distance, we used the kernel size of 2 for convolution, as shown in Figure 2.

After the first shortcut connect, another circulation block is added, and it consists of a pooling layer, two convolution layers, and a shortcut connection, in which the convolution operation is defined as $g(x) * \Omega + b$, where $g(x)$ is for ReLU and Ω is the filter initialized by a normal distribution. At the beginning of this submodule, we performed max-pooling with size 2 and stride 2: the pooling layer chooses the maximum over 2 contiguous internal vectors and the 2-stride pooling window reduces the size of the input feature map by half. After this, output from each pooling layer is collected, arranged in the form of a "pyramid," and this is where DP takes its name.

For the shortcut connection $f(x) + x$, both $f(x)$ and x require the same dimensionality so that they can be summed up. To avoid the extra dimension matching operation, in the DP structure, we unified the convolutional layers with the same number of filters to obtain N_* feature maps for each convolution layer. In DP-ConvKB, we set $N_* = N$, where N is the number of filters used in ConvKB (the right module);

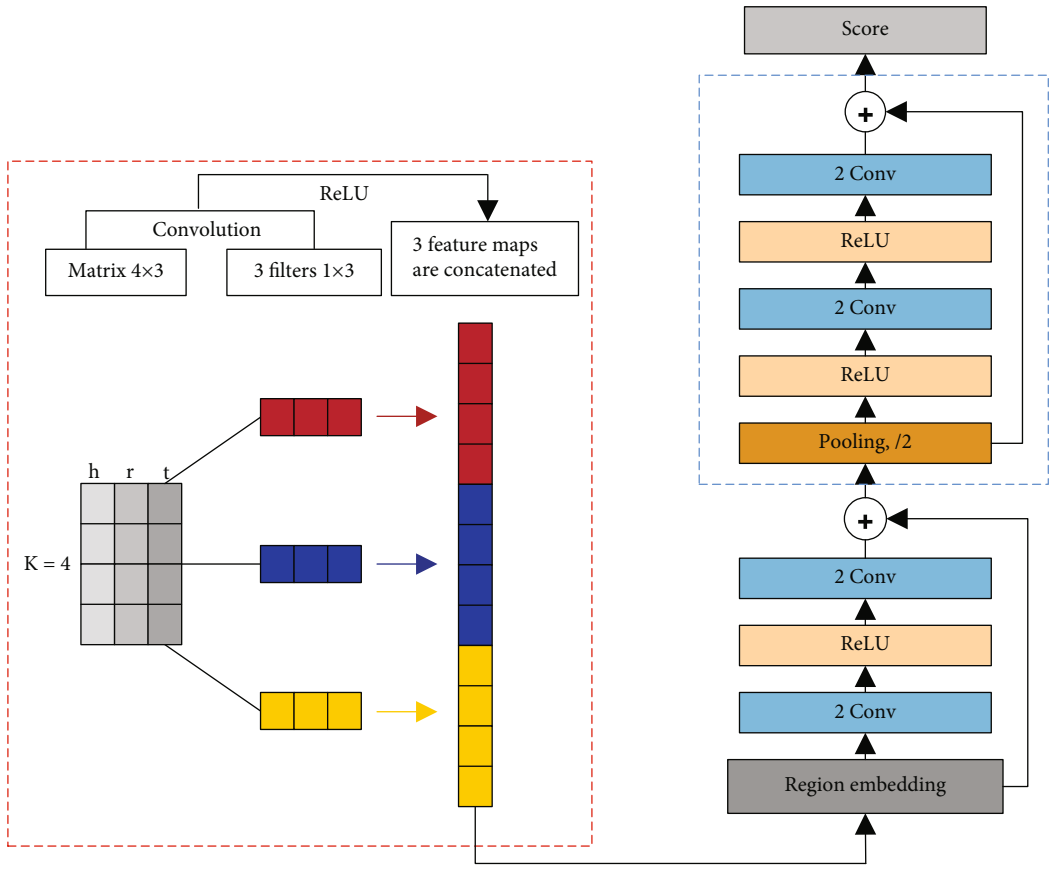


Figure 2: Illustration of the DP-ConvKB model framework.

Table 2: Statistics of the experimental datasets used in this study. #Entity is the number of entities, #Rel is the number of relation types, and #Train, #Valid, and #Test are the numbers of triples in the training, validation, and test sets, respectively.

Dataset	#Entity	#Rel	#Train	#Valid	#Test
WN18RR	40943	11	86835	3034	3134
FB15k-237	14541	237	272115	17535	20466

the value of region embedding is the same as V_{ConvKB}. After the circulation block, the score function of the model can be written as follows:

$$f_{\text{DP-ConvKB}} = V_C \cdot W, \tag{5}$$

where V_C is the output of the final convolution layer and W is a weight vector.

4. Experiment Setup

4.1. Dataset. We evaluated MF-ConvKB and DP-ConvKB models on two benchmark datasets: WN18RR and FB15k-237, which are the subsets of the datasets WN18 and FB15k, respectively. This is because WN18 and FB15k contain highly redundant relations, as mentioned in studies [42, 43]; these highly redundant relations could lead to inaccurate test results. One example of redundant relations is when a triple and its inverse triple exist in the test set and training set, respectively. This would make the original triple easily retrievable which leads to an overly good result and jeopardize the model's generalization. WN18RR and FB15k-237 are two datasets intentionally designed to remove triples with such inverse relation. Statistics of the two benchmark datasets are given in Table 2.

The relations from the two datasets can be generally classified into 4 categories: 1 – to – 1, 1 – to – M, M – to – 1, and M – to – M, with M for MANY. According to [9], 1 – to – 1 denotes one head can appear with one tail entity at most, and 1 – to – M denotes one head can appear with more than one tail. Similarly, M – to – 1 denotes more than one head can appear with the same tail, and M – to – M. is for the case that multiple heads map to multiple tails. We found that 0.9% of the relations on FB15k-237 is 1 – to – 1 type, and for 1 – to – M, M – to – 1, and M – to – M type, the portion is 6.3%, 20.5%, and 72.3%, respectively. Regarding the WIN18RR dataset, there are 11 relations, among which, *also_see*, *similar_to*, *verb_group*, and *derivationally_*

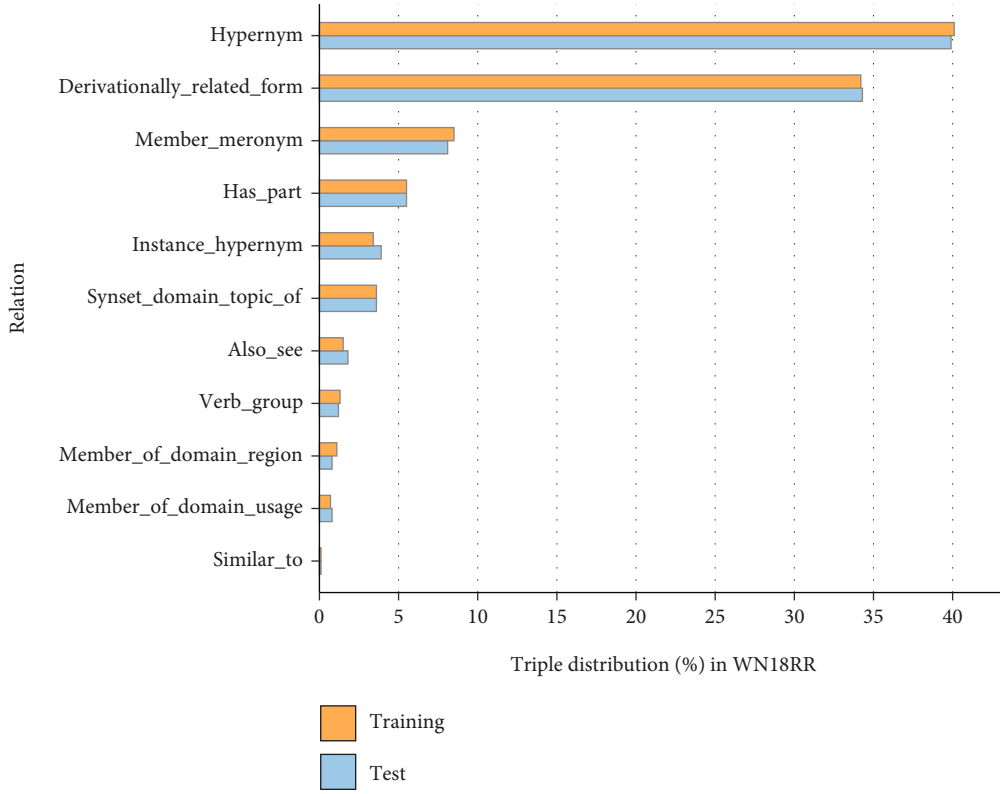


Figure 3: Distribution of both training and test set in WN18RR based on relations.

related_form can be classified into $M-\text{to}-M$ relation, and *member_meronym* and *hypernym* are $1-\text{to}-M$ and $M-\text{to}-1$ relation, respectively. Moreover, there are 1251 triples containing the relation *hypernym* and 1074 triples containing the relation *derivationally_related_form*, which in total accounts for about 75% of 3134 test triples in the test set. Figure 3 shows the percentage of each relation in both test and training sets.

4.2. Design of Loss Function. The loss function was minimized as follows:

$$L = \sum_{(h,r,t)\in\varsigma\cup\varsigma'} \log\left(1+\exp\left(l_{(h,r,t)}\cdot f_{\text{DP-ConvKB}}(h,r,t)\right)\right) + \frac{\lambda}{2}\|w\|_2^2,$$
$$\text{in which, } l_{(h,r,t)} = \begin{cases} 1 & \text{for}(h,r,t)\in\varsigma, \\ -1 & \text{for}(h,r,t)\in\varsigma', \end{cases} \tag{6}$$

where ς is the train set collection of valid triples, while ς' is a collection of invalid triples generated by corrupting a valid triple $(h,r,t)\in\varsigma$ by replacing the head entity or the tail entity with other entities in $\mathscr{E}$. According to Bernoulli trick [4], the new invalid triples (h',r,t) and (h,r,t') occur with the probability $\eta_h/(\eta_h+\eta_t)$ and $\eta_t/(\eta_h+\eta_t)$, respectively. For a certain relation r, η_h denotes the average number of head entities per tail entity, and η_t denotes the average number of tail entities per head entity. $f_{\text{DP-ConvKB}}$ is the score function as in (5), and $(\lambda/2)\|w\|_2^2$ is the $L2$ regularization on w.

4.3. Evaluation Protocol. The object of this study is to predict the missing entity in a triple, i.e., predicting the head entity with given (r,t) or predicting the tail entity with given (h,r). To evaluate our proposed models' performance, we focused on ranking the scores of candidate entities from the dataset. We employed three commonly used evaluation metrics in the previous study: mean rank (MR), mean reciprocal rank (MRR), and Hits@10. MR denotes the mean rank of all test triples as calculated in (7). It evaluates whether all of the ground-truth relevant items selected by the model are ranked higher or not.

$$\text{Mean Rank} = \frac{1}{N}\sum_{i=1}^{N}\text{rank}_i. \tag{7}$$

MRR is calculated by taking the mean of the reciprocal rank for each query of the test triples as in (8), and it only cares about the single highest-ranked relevant item.

$$\text{Mean Reciprocal Rank} = \frac{1}{N}\sum_{i=1}^{N}\frac{1}{\text{ran k}_i}. \tag{8}$$

Hits@k is the proportion of the rank that is lower than or equal to k, with k usually set to 10 in the link prediction task. Either lower MR, higher MRR, or higher Hits@k indicates a better result of the task. Note that, according to the work of

Table 3: Comparison results of link prediction task on WN18RR and FB15k-237 test sets. Hits@10 is reported in %.

Method	WN18RR			FB15k-237		
	MR	MRR	Hits@10	MR	MRR	Hits@10
TransE	3384	0.226	50.1	347	0.294	46.4
DistMult	5110	0.425	49.1	254	0.241	41.9
ComplEx	5261	0.444	50.7	339	0.247	42.8
ConvE	4187	0.433	51.5	244	0.325	50.1
ConvKB	1711	0.251	52.9	246	0.407	52.7
CapsE	719	0.415	56.0	303	0.523	59.3
RotatE	3340	0.476	57.1	177	0.338	53.3
MF-ConvKB	1273	0.261	56.1	207	0.307	48.2
DP-ConvKB	139	0.703	73.1	101	0.734	75.0

TransE [9], we had to remove the corrupt triples that already exist in the datasets when ranking the test triples.

4.4. Implementation Details. Prior to the model training, the entities and relation embeddings were first preprocessed by the embeddings produced from TransE. We then applied stochastic gradient descent (SGD) algorithm with 3000 epochs on WN18RR and FB15k-237 to train the TransE model parameters. We adopted grid search algorithm on the validation set, in order to optimize parameters, including the dimensionality of the word embedding, the margin hyperparameter, and the SGD learning rate. We fixed the batch size = 256 in our objective function. Our experiment showed that, for MF-ConvKB, the highest Hits@10 can be obtained when $k = 50$, $r = 5$, and $\eta = 5e^{-4}$ on WN18RR and $k = 100$, $r = 1$, and $\eta = 5e^{-4}$ on FB15k-237; for DP-ConvKB, these optimized parameters were $k = 100$, $r = 1$, and $\eta = 5e^{-4}$ for both datasets.

We used the Adam optimizer to train both MF-ConvKB and DP-ConvKB. For training MF-ConvKB, we set the initial learning rate of Adam [44] at $\eta \in \{1e^{-3}, 1e^{-4}, 5e^{-5}, 5e^{-6}\}$, and set the L_2-regularizer λ at 0.001 to avoid overfitting. After convolution, we chose ReLU as the activation function. We designed the filters with the shape of 1×3, 2×3, and 3×3. Each filter was initialized by a truncated normal distribution. The highest Hits@10 scores were obtained when using $N = [500,500,500]$, $k = 50$, and $\eta = 1e^{-4}$ on WN18RR and $N = [100,40,20]$, $k = 100$, and $\eta = 5e^{-6}$ on FB15k-237. For training DP-ConvKB, we set the initial learning rate of Adam at $\eta \in \{1e^{-5}, 1e^{-6}, 1e^{-7}, 5e^{-8}\}$ and set the L_2-regularizer λ at 0.001. To get the feature map for deep convolution, we initialized the original filter by a truncated normal distribution, and we fixed its shape at 1×3. In the deep NN part, convolutional filters were initialized by a normal distribution. The highest Hits@10 scores were obtained when using $N = 200$, $k = 100$, and $\eta = 5e^{-8}$ on WN18RR and $N = 100$, $k = 100$, and $\eta = 1e^{-7}$ on FB15k-237.

4.5. Main Results. Table 3 shows the link prediction comparison results between our two models and the state-of-the-art models.

From Table 3, it is clear that the proposed model DP-ConvKB significantly outperformed all other models listed, including the original ConvKB (baseline model, hereafter). Specifically, on dataset WN18RR, DP-ConvKB obtains significant improvements of $0.703 - 0.251 = 0.452$ in MRR and $73.1\% - 52.9\% = 20.2\%$ absolute improvement in Hits@10. On dataset FB15k-237, these improvements were 0.327 in MRR and 22.3% in Hits@10.

As shown in Table 3, we also noticed that the tentative model MF-ConvKB performed just slightly better than the baseline model on all metrics on WN18RR and even obtained a less score of MRR and Hits@10 on FB15k-237. However, it could achieve lower MR than the baseline model on both datasets. The shape of the convolutional filter is known to play a pivotal role in extracting features in fields like image processing [45], speech recognition [46], and sentence classification [47]. However, its effect in the link prediction task is not well understood. To investigate, we embedded convolutional filters with different shapes into MF-ConvKB and test model performances on link prediction tasks. Results are shown in Table 4: the first row with a single 1×3 filter represents the baseline model, and starting from the second row, we replaced the filter with different shapes and the final row was from MF-ConvKB. These results suggested that, in general, replacing the single filter in the baseline model with different shapes did not bring substantial improvements and combing multiple different filters can ameliorate performance on MR but failed on MRR and Hits@10 necessarily. Specifically, MF-ConvKB with a variety of filters can have better performances on WN18RR but not necessarily on FB15k-237. Taken together, these results indicate that combining filters of different shapes may not be the most efficient strategy for link prediction task performance improvement. Therefore, we sought other possible structures, e.g., deep NN in our proposed model DP-ConvKB, to improve link prediction task performance.

Figure 4 shows a result comparison between our proposed DP-ConvKB model and baseline model, tested on MRR and Hits@10 scores of different relations on dataset FB15k-237. Results of predicting head and tail entities are separately displayed. It is clear that DP-ConvKB improves baseline performances on all relations. Specifically, $M - 1$ achieves the highest scores, followed by $M - M$, $1 - 1$, and $1 - M$ ranking last. Moreover, DP-ConvKB shows more

Table 4: Link prediction results of different designed filters on WN18RR and FB15k-237 test sets.

Filter shape	WN18RR			FB15k-237		
	MR	MRR	Hits@10	MR	MRR	Hits@10
1×3 (baseline)	1711	0.251	52.9	246	0.407	52.7
2×3	1614	0.258	54.9	260	0.417	52.6
3×3	1547	0.251	54.9	277	0.424	52.6
$2 \times 3, 3 \times 3$	1480	0.256	55.0	296	0.400	50.8
$1 \times 3, 2 \times 3, 3 \times 3$ (MF-ConvKB)	1273	0.261	56.1	207	0.307	48.2

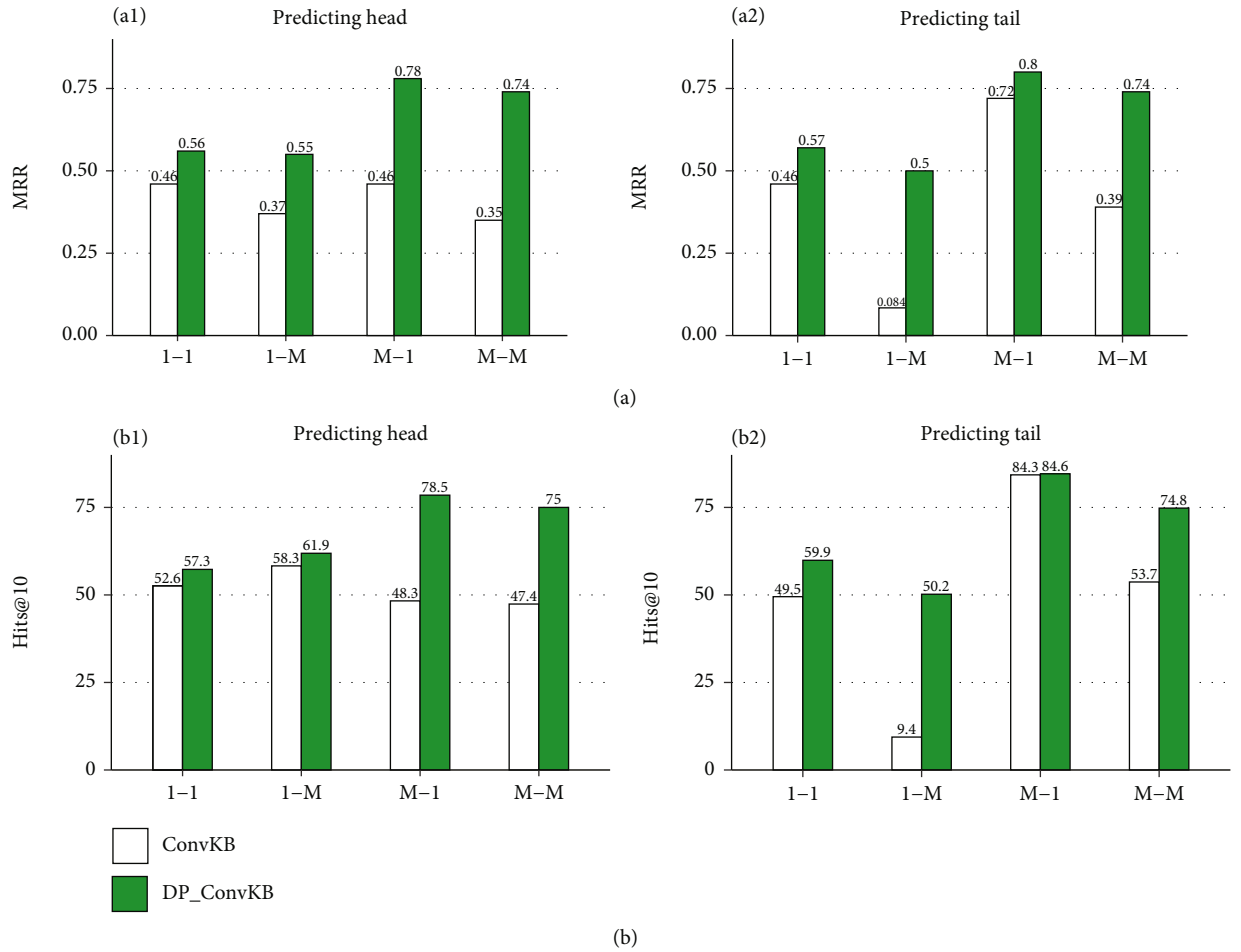


Figure 4: Performance of (a) MRR and (b) Hits@10 on FB15k-237 from DP-ConvKB and baseline model. A1 and B1 are the results of predicting head task, while A2 and B2 are the results of predicting tail task. Results are depicted in 4 categories of relations, which are $1 - 1$, $1 - M$, $M - 1$, and $M - M$.

robust performances when predicting head and tail on $1 - M$ and $M - 1$ relations, while the performances from the baseline model fluctuate in these cases (from 0.37 to 0.084 and 0.46 to 0.72 on MRR).

Performance comparisons for all relations on dataset WN18RR between DP-ConvKB and baseline model are shown in Figure 5. In general, our proposed model DP-ConvKB outperforms the baseline model in all considered relations on both Hits@10 and MRR metrics. Strikingly, when evaluated by MRR score, model performance is completely boosted by DP-ConvKB by more than 2 times. Particularly, performances on $M - M$ relations of *Verb_group* and *similar_to* reach the highest scores on MRR. On WN18RR, we also tested the model's performances in predicting heads and tails. Figure 6 depicts the performance of DP-ConvKB in predicting head and tail tasks with the metrics of MRR and Hits@10 on the dataset WN18RR. We can see that the scores of predicting head and tail on almost every relation category are very close. As a comparison, the baseline model has a relatively large gap in predicting head and tail tasks which can be found in Figure 4. Results shown in Figure 6 demonstrate the

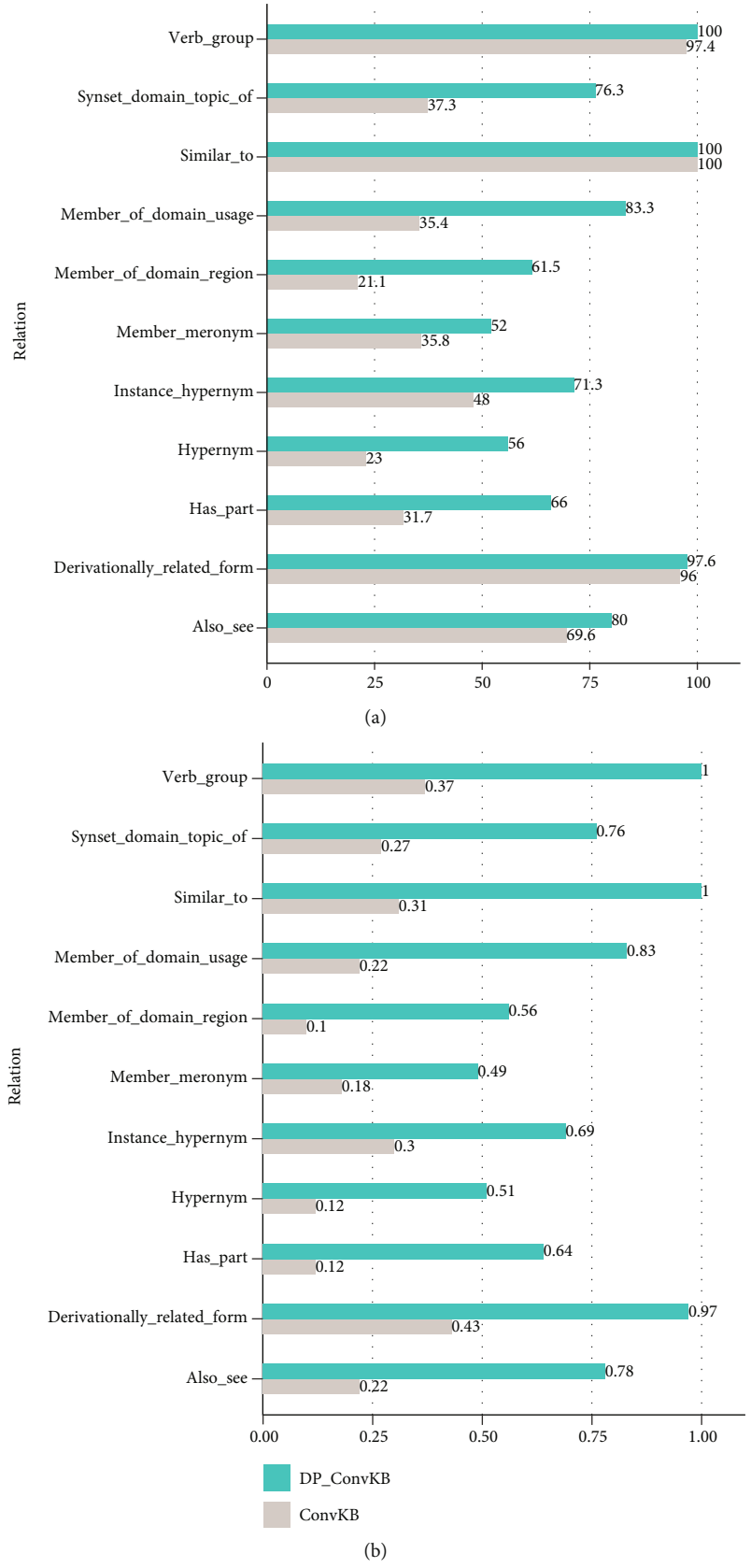


Figure 5: Performance of (a) Hits@10 and (b) MRR on WN18RR from DP-ConvKB and baseline model. There are 11 different relations in dataset WN18RR.

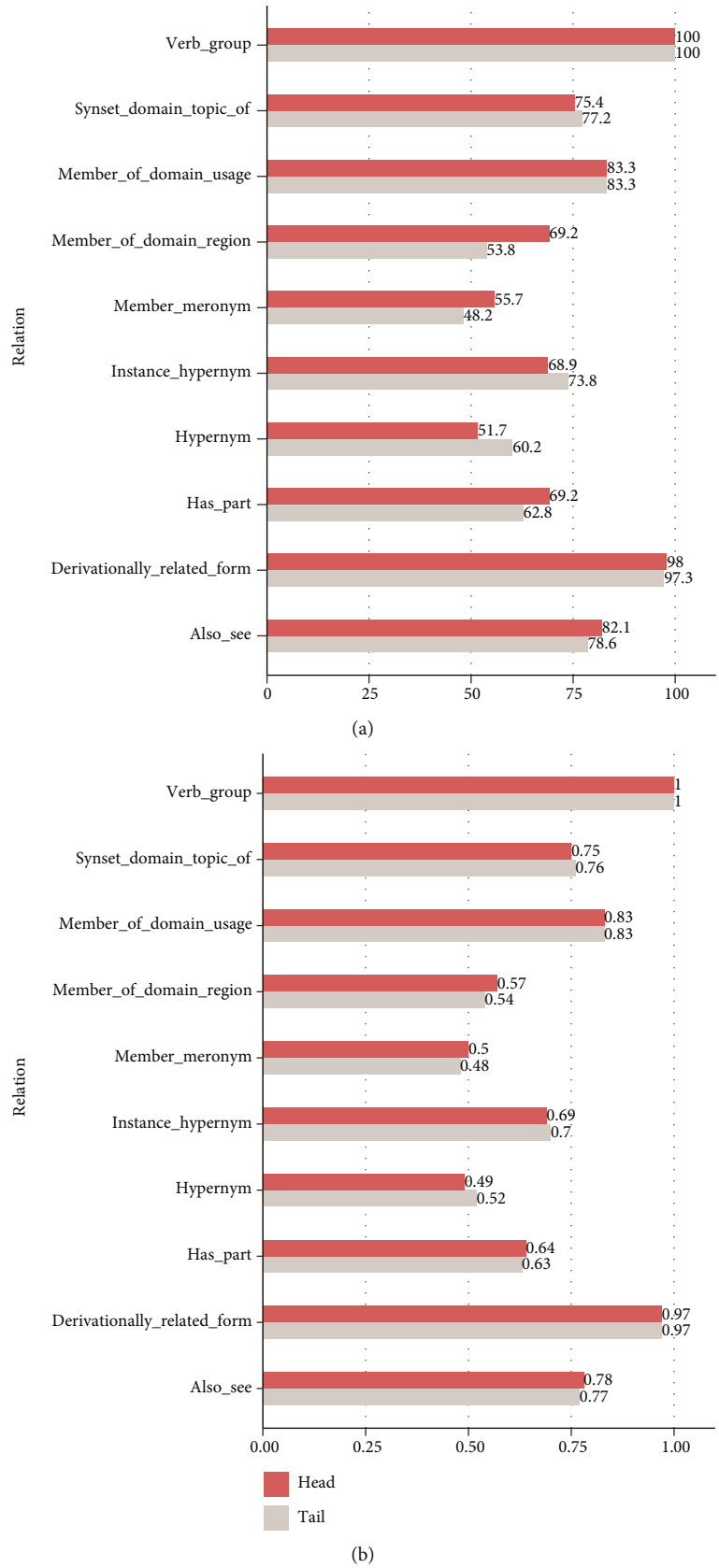


Figure 6: Performance of DP-ConvKB in predicting head and tail tasks on WN18RR. Evaluation metrics are (a) Hits@10 and (b) MRR.

ability of minimizing discrepancies in predicting head and tail when using DP-ConvKB.

Taken together, the overall experimental results show that the deep CNN-based model DP-ConvKB can effectively sum up the global features beyond distance and obtain the best performance for both the head prediction task and tail prediction task.

5. Conclusion

In this paper, we proposed a CNN-based model, DP-ConvKB, to improve the performance of the knowledge graph completion task. We first showed that simply adding a variety of filters into the pioneer model ConvKB might not be in the right direction to enhance its performance. We then designed a new model, DP-ConvKB, which cooperates a deep pyramid neural network into ConvKB; therefore, it is capable of exploring the deep features. Our results on datasets WN18RR and FB15k-237 show that DP-ConvKB outperforms the baseline model (ConvKB). DP-ConvKB obtains the best mean rank and the highest mean reciprocal rank and Hits@10. To sum up, our study demonstrates that, by implementing such deep convolutional network structure into models for KGC tasks, it can significantly improve the performances.

Although DP-ConvKB has achieved great improvement on the link prediction task, however, it involves more computational complexity and is difficult to find the optimized hyperparameters in the training process. In future work, we plan to prune the deep neural networks to minimize the training time and generalize our structure to other NLP tasks.

Data Availability

The data that support the findings of this study are available from the authors. Requests for access to these data should be made to Xueting Wang, xuetingcuc@163.com.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] A. Singhal, "Introducing the knowledge graph: things, not strings," 2012, http://googleblog.blogspot.com/2012/05/introducing-knowledge-graph-things-not.html.

[2] D. Sullivan, "A reintroduction to our knowledge graph and knowledge panels," 2020, https://blog.google/products/search/about-knowledge-graph-and-knowledge-panels/.

[3] X. Dong, E. Gabrilovich, G. Heitz et al., *Knowledge Vault: A Web-Scale Approach to Probabilistic Knowledge Fusion*, ACM, 2014.

[4] D. Krompaß, S. Baier, and V. Tresp, "Type-constrained representation learning in knowledge graphs," in *The Semantic Web-ISWC*, vol. 9366, pp. 640–655, Springer, Cham, 2015.

[5] G. He, J. Li, W. X. Zhao, P. Liu, and J. R. Wen, "Mining implicit entity preference from user-item interaction data for knowledge graph completion via adversarial learning," in *Proceedings of The Web Conference*, Taipei Taiwan, 2020.

[6] A. Rossi, D. Firmani, A. Matinata, P. Merialdo, and D. Barbosa, "Knowledge graph embedding for link prediction," *A Comparative Analysis*, vol. 15, no. 2, pp. 1–49, 2021.

[7] Z. Chen, Y. Wang, B. Zhao, J. Cheng, and Z. Duan, "Knowledge graph completion: a review," *IEEE Access*, vol. 8, pp. 192435–192456, 2020.

[8] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: a review and new perspectives," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[9] A. Bordes, N. Usunier, A. Garcia-Duran, J. Weston, and O. Yakhnenko, "Translating embeddings for modeling multi-relational data," *Advances in Neural Information Processing Systems*, vol. 26, pp. 2787–2795, 2013.

[10] J. Zhang, "Knowledge graph embedding by translating on hyperplanes," in *Proceedings of the AAAI Conference on Artificial Intelligence*, Québec City, Québec, Canada, 2014.

[11] G. Ji, S. He, L. Xu, K. Liu, and J. Zhao, "Knowledge graph embedding via dynamic mapping matrix," in *Proceedings of the 53rd annual meeting of the association for computational linguistics and the 7th international joint conference on natural language processing*, pp. 687–696, Beijing, China, 2015.

[12] Y. Lin, Z. Liu, M. Sun, Y. Liu, and X. Zhu, "Learning entity and relation embeddings for knowledge graph completion," *Twenty-ninth AAAI conference on artificial intelligence*, 2015.

[13] X. Han, M. Huang, H. Yu, and X. Zhu, "TransG: a generative mixture model for knowledge graph embedding," 2015, https://arxiv.org/abs/1509.05488.

[14] B. Yang, W. T. Yih, X. He, J. Gao, and L. Deng, "Embedding entities and relations for learning and inference in knowledge bases," 2014, https://arxiv.org/abs/1412.6575.

[15] T. Trouillon, J. Welbl, S. Riedel, R. Gaussier, and G. Bouchard, "Complex embeddings for simple link prediction," 2016, https://JMLR.org.

[16] M. Yan, Z. Li, X. Yu, and C. Jin, "An end-to-end deep learning network for 3D object detection from RGB-D data based on Hough voting," *IEEE Access*, vol. 8, pp. 138810–138822, 2020.

[17] N. Kalchbrenner, E. Grefenstette, and P. Blunsom, "A convolutional neural network for modelling sentences," vol. 1, 2014, https://arxiv.org/abs/1404.2188.

[18] Y. Kim, "Convolutional Neural Networks for Sentence Classification," in *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)* of *Association for Computational Linguistics*, pp. 1746–1751, Doha, Qatar, 2014.

[19] Y. Shen, X. He, J. Gao, L. Deng, and G. Mesnil, "Learning semantic representations using convolutional neural networks for web search," in *Proceedings of the 23rd International Conference on World Wide Web*, Seoul, Korea, 2014.

[20] T. Dettmers, P. Minervini, P. Stenetorp, and S. Riedel, "Convolutional 2D knowledge graph embeddings," in *32nd AAAI Conference on Artificial Intelligence (AAAI-18), February 2018*, New Orleans, LA, USA, 2017.

[21] Q. N. Dai, D. N. Tu, D. Q. Nguyen, and D. Phung, "A novel embedding model for knowledge base completion based on convolutional neural network," in *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, New Orleans, Louisiana, 2018.

[22] D. Q. Nguyen, T. Vu, T. D. Nguyen, D. Q. Nguyen, and D. Phung, "A capsule network-based embedding model for knowledge graph completion and search personalization," in *Proceedings of the 2019 Conference of the North*, Minneapolis, Minnesota, 2019.

[23] X. Jiang, Q. Wang, and B. Wang, "Adaptive convolution for multi-relational learning," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 978–987, Minneapolis, Minnesota, June 2019.

[24] D. Q. Nguyen, K. Sirts, L. Qu, and M. Johnson, "STransE: a novel embedding model of entities and relationships in knowledge bases," 2016, https://arxiv.org/abs/1606.08140.

[25] G. Ji, K. Liu, S. He, and J. Zhao, "Knowledge graph completion with adaptive sparse transfer matrix," in *Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence*, Phoenix, Arizona, 2016.

[26] Y. Lin, Z. Liu, H. Luan, M. Sun, S. Rao, and S. Liu, "Modeling relation paths for representation learning of knowledge bases," 2015, https://arxiv.org/abs/1506.00379.

[27] W. Huang, L. Ge, and J. Zhi, "Improved knowledge base completion by the path-augmented TransR model," in *International Conference on Knowledge Science, Engineering and Management*, Springer, Cham, 2017.

[28] Z. Sun, Z. H. Deng, J. Y. Nie, and J. Tang, "RotatE: knowledge graph embedding by relational rotation in complex space," 2019, https://arxiv.org/abs/1902.10197.

[29] M. Nickel, V. Tresp, and H. P. Kriegel, "A three-way model for collective learning on multi-relational data," *International Conference on International Conference on Machine Learning*, , Omnipress, Madison, WI, USA, 2011.

[30] S. Mehran Kazemi and D. Poole, "SimplE embedding for link prediction in knowledge graphs," 2018, https://arxiv.org/abs/1802.04868.

[31] C. Shang, Y. Tang, J. Huang, J. Bi, and B. Zhou, "End-to-end structure-aware convolutional networks for knowledge base completion," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, pp. 3060–3067, 2019.

[32] M. Schlichtkrull, T. N. Kipf, P. Bloem, R. V. Berg, and M. Welling, "Modeling relational data with graph convolutional networks," in *European Semantic Web Conference*, Springer, Cham, 2018.

[33] H. Wang, V. Kulkarni, and W. Y. Wang, "DOLORES: deep contextualized knowledge graph embeddings," 2018, https://arxiv.org/abs/1811.00147.

[34] D. Nathani, J. Chauhan, C. Sharma, and M. Kaul, "Learning attention-based embeddings for relation prediction in knowledge graphs," 2019, https://arxiv.org/abs/1906.01195.

[35] R. Wang, B. Li, S. Hu, W. Du, and M. Zhang, "Knowledge graph embedding via graph attenuated attention networks," *IEEE access*, vol. 99, pp. 1–1, 2019.

[36] G. Stoica, O. Stretcu, E. A. Platanios, T. Mitchell, and B. Póczos, "Contextual parameter generation for knowledge graph link prediction," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 3, pp. 3000–3008, 2020.

[37] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[38] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, 2012.

[39] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, Las Vegas, NV, USA, 2016.

[40] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, and A. Rabinovich, "Going deeper with convolutions," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, Boston, MA, 2014.

[41] R. Johnson and T. Zhang, "Deep pyramid convolutional neural networks for text categorization," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 562–570, Vancouver, Canada, 2017.

[42] K. Toutanova, D. Chen, P. Pantel, H. Poon, P. Choudhury, and M. Gamon, "Representing text for joint embedding of text and knowledge bases," in *Proceedings of the 2015 conference on empirical methods in natural language processing*, pp. 1499–1509, Lisbon, Portugal, 2015.

[43] K. Toutanova and D. Chen, "Observed versus latent features for knowledge base and text inference," in *Proceedings of the 3rd workshop on continuous vector space models and their compositionality*, pp. 57–66, Beijing, China, 2015.

[44] D. P. Kingma and J. Ba, "Adam: a method for stochastic optimization," 2014, https://arxiv.org/abs/1412.6980.

[45] S. Han, Z. Meng, Z. Li et al., "Optimizing Filter Size in Convolutional Neural Networks for Facial Action Unit Recognition," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5070–5078, Salt Lake City, 2018.

[46] V. Mitra, G. Sivaraman, H. Nam, C. Espy-Wilson, E. Saltzman, and M. Tiede, "Hybrid convolutional neural networks for articulatory and acoustic information based speech recognition," *Speech Communication*, vol. 89, pp. 103–112, 2017.

[47] E. Vargas-Ocampo, E. Roman-Rangel, and J. Hermosillo-Valadez, *Learning Word and Sentence Embeddings Using a Generative Convolutional Network*, Pattern Recognition, pp. 135–144, Springer International Publishing, Cham, 2018.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 4570755, 10 pages
https://doi.org/10.1155/2022/4570755

Research Article

Flexible Light Field Angular Superresolution via a Deep Coarse-to-Fine Framework

Qian Wang, Li Fang, Long Ye, Wei Zhong, Fei Hu, and Qin Zhang

Key Laboratory of Media Audio & Video (Communication University of China), Ministry of Education, Beijing 100024, China

Correspondence should be addressed to Li Fang; lifang8902@cuc.edu.cn

Received 10 October 2021; Accepted 25 January 2022; Published 10 March 2022

Academic Editor: Ivan Lee

Copyright © 2022 Qian 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.

Acquisition of densely-sampled light fields (LFs) is challenging. In this paper, we develop a coarse-to-fine light field angular superresolution that reconstructs densely-sampled LFs from sparsely-sampled ones. Unlike most of other methods, which are limited by the regularity of sampling patterns, our method can flexibly deal with different scale factors with one model. Specifically, a coarse restoration on epipolar plane images (EPIs) with arbitrary angular resolution is performed and then a refinement with 3D convolutional neural networks (CNNs) on stacked EPIs. The subaperture images in LFs are synthesized first horizontally, then vertically, forming a pseudo 4DCNN. In addition, our method can handle large baseline light field without using geometry information, which means it is not constrained by Lambertian assumption. Experimental results over various light field datasets including large baseline LFs demonstrate the significant superiority of our method when compared with state-of-the-art ones.

1. Introduction

The light field (LF) encodes the distribution of light into a high-dimensional function, contains rich scene visual information [1, 2] and has a wide range of applications in various fields, such as image refocusing [3], 3D scene reconstruction [4], depth inference [5], and virtual augmented reality [6]. In order to obtain high-quality views without ghosting effects, many studies have focused on dense sampling of LF [7].

Dense sampling of LF means great acquisition difficulties. Early light field cameras, such as multicamera arrays and light field racks etc. [8], are bulky and expensive in hardware. In recent years, the introduction of commercial and industrial light field cameras such as Lytro [9] and RayTrix [10] has brought light field imaging into a new era. Unfortunately, due to the limited resolution of the sensor, a trade-off must be made between spatial resolution and angular resolution.

One possible solution to this problem is view synthesis, which synthesizes novel views from a sparse set of input views. One type of existing previous work is to estimate the scene depth as auxiliary information [11, 12], but it relies heavily on the depth estimation, which tends to fail in occluded regions, as well as in glossy or specular ones. The other is based on sampling and consecutive reconstruction of the plenoptic function [13–16]. They do not use the depth information as an auxiliary mapping but suffer from either aliasing or blurring problem when the input LF is extremely undersampled.

Some learning-based methods have recently appeared [17–19], and they can be roughly classified into two categories: nondepth based and depth based. But most of them have average reconstruction quality under large parallax conditions. Besides, retraining is required for different scale factors, which increases the difficulty of actual acquisition.

In our previous work [20], we proposed a learning-based model for reconstructing densely-sampled LFs via angular superresolution, which is achieved by using an image superresolution network on epipolar plane images (EPIs). However, EPIs have very clear structure, which is very different from natural images. The performance is degraded by large-baseline sampling. In this paper, we provide a few distinguishable improvements and enable flexible and accurate reconstruction of a densely-sampled LF from very sparse

sampling. We inherit the coarse-to-fine framework in [20], that is, the proposed model consists of a coarse EPI angular superresolution module and an efficient EPI stack refinement module. Specifically, the coarse EPI angular superresolution module magnifies each EPI individually using a specially designed EPI superresolution network, where independent coefficients are used for row and column EPIs, respectively. We further refine the coarse results using 3DCNNs on stacked EPI based on photoconsistency between subaperture images. The overall frame explores pseudo 4DCNN, which is capable of making full use of LF data. In addition, we introduced perceptual loss [21] to better fit the network training. Experimental results demonstrate the superiority of our method in reconstruction with higher numerical quality and better visual effect.

The rest of this paper is organized as follows: Latest developments of view synthesis and LF reconstruction are introduced in Section 2. Our presented approach is described in Section 3. The performance evaluation is given in Section 4. Finally, Section 5 summarizes this paper.

2. Related Works

The problem of reconstructing a complete densely-sampled LF from a set of sparsely sampled images has been extensively studied. These algorithms can be divided into depth-dependent view synthesis that depends on depth information and depth-independent LF reconstruction that does not depend on depth information.

2.1. Depth-Dependent View Synthesis. Depth-dependent view synthesis approaches typically consist of two steps to synthesize the novel view of the scene [22], i.e., first estimating depth map at the novel view or the input view, and then using it to synthesize the novel [23], Kalantari et al. [17] proposed the first deep learning system for view synthesis with two sequential networks that perform depth estimation and color prediction successively. Srinivasan et al. [24] proposed to synthesize a 4D RGBD LF from a single 2D RGB image based on estimated 4D ray depth. Flynn et al. [12] mapped input views to a set of depth planes of the same perspective through homography transform and then fused them together through two parallel CNNs for learning weights to average the color of each plane. Zhou et al. [25] and Mildenhall and Ben [26] trained a network that inferred alpha and multiplane images. Although utilizing depth information makes it easier to handle inputs with large disparities, most methods cannot achieve acceptable performance for large-baseline sampling, Jin et al. [27] focus on the angular superresolution of light field images with a large-baseline and propose an end-to-end trainable method, by making full use of the intrinsic geometry information of light fields. However, unfortunately inaccurate depth estimation usually happens in challenging scenes that contain significant depth variations, complex lighting conditions, occlusions, non-Lambertian surfaces, etc.

2.2. Depth-Independent LF Reconstruction. Depth-independent LF reconstruction approaches can be considered as an angular dimension upsampling without any geometry information of the scene. Zhouchen and Heung-Yeung [28] proved that for a LF which disparity between neighboring views is less than one pixel, and novel views can be produced using linear interpolation. Some methods have investigated LF reconstruction with specific sampling patterns. Levin and Durand [29] exploited dimensionality gap priors to synthesize novel views from a set of images sampled in a circular pattern. Lixin et al. [14] sampled only the boundary viewpoints or diagonal viewpoints to recover the full LF using sparsity analysis in the Fourier domain. These methods are far from practical application due to the difficulty in capturing input views in specific mode.

Recently, deep learning has been applied in many fields, such as 3D object [30] detection and object recognition [31]. Some learning-based approaches were also proposed for depth-independent reconstruction. Yoon et al. [32] proposed a deep learning framework, in which two adjacent views are employed to generate the interview, while it can only generate novel views at 2x upsampling factor. Wu et al. [19] proposed a blur-restoration-deblur framework as learning-based angular detail restoration on 2D EPIs. Wu et al. [33] further discussed the trade-off either aliasing or blurring problem and designed a Laplacian pyramid EPI (LapEPI) structure that contains both low spatial scale EPI (for aliasing) and high-frequency residuals (for blurring) to solve the trade-off problem. However, the potential of the full LF data is under used in both works. Wang et al. [24] and their subsequent work [35] introduced an end-to-end learning-based pseudo 4DCNN framework using 3D LF volumes with a fixed interpolating rate. On the basis of the pseudo 4DCNN network, Ran et al. [20] added an EPI repair module that magnifies the EPI with arbitrary scale factor.

3. The Proposed Approach

3.1. 4D LF Representation. A 4D LF is usually denoted as $L(u, v, s, t)$, which uses the intersections of light rays with two parallel planes to record light rays, called the two-plane representation (see Figure 1). Each light ray travels from the spatial coordinates (u, v) on the focal plane then to the angular coordinates (s, t) on the camera plane. Therefore, a 4D LF is regarded as a 2D image array with $Cols \times Rows$ images sampled on a 2D angular grid, of which each image is of spatial resolution $W \times H$.

As shown in Figure 1, by fixing u in spatial domain and s in angular domain (or v and t), we can get an EPI map denoted as $E_{u_0,s_0}(v, t)$ (or $E_{v_0,t_0}(u, s)$), which is a 2D slice of the 4D LF. A 3D volume $V_{t_0}(u, v, s)$ (or $V_{s_0}(u, v, t)$) can be produced if we stack EPIs from a column (or a row) views by fixing $t = t_0$ (or $s = s_0$). In the EPI, relative motion between the camera and object points manifests as lines with depth depending slopes. Thus, EPIs can be regarded as an implicit representation of the scene geometry. EPI has a highly clear structure compared with conventional photo images.

The goal of LF angular superresolution is to recover a densely-sampled LF with a resolution of $W \times H \times Cols \times Ro$

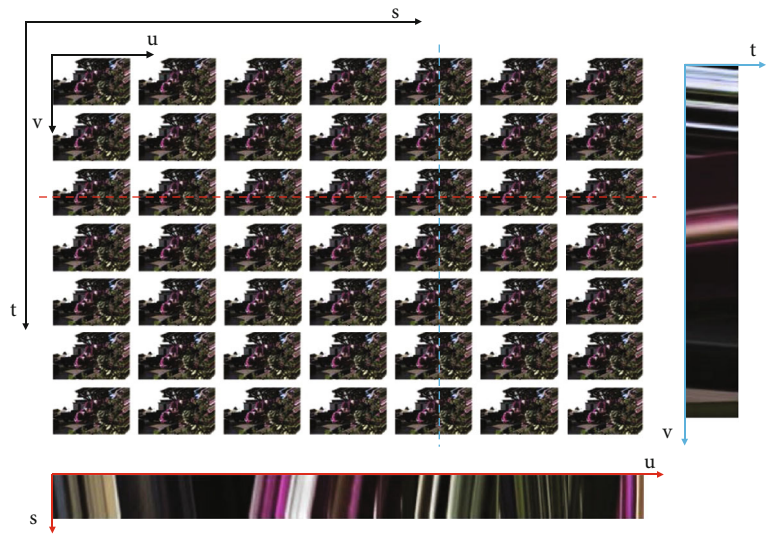


FIGURE 1: A 4D light field $L(u, v, s, t)$ visualization. The horizontal EPI is a 2D (u, s) slice $L(u, v_0, s, t_0)$ by positioning $v = v_0$ and $t = t_0$ (highlighted in red) and the vertical EPI (v, t) by positioning $u = u_0$ and $s = s_0$ (highlighted in blue).

ws from a sparse one with a resolution of $W \times H \times cols \times rows$. The presented framework will be comprehensively introduced in the next section.

3.2. Overview of the Proposed Method. In order to make full use of 4D LF data while circumventing high computational burden, we propose to first synthesize novel views in each row and then in each column, forming a pseudo 4DCNN. Specifically, as shown in Figure 2, for each row/column of views, our network consists of two parts: coarse EPI angular superresolution module based on 2D metalearning and EPI stack refinement module using 3DCNNs. First, we perform these two parts on the 3D row EPI volumes, after by converting the angular from row to column. Then, perform the same operation on the 3D column EPI volumes.

Given an input sparse LF $LF_{\text{in}}(u, v, s, t)$, with the size of $W \times H \times cols \times rows$. First we fix the angular coordinate $s = s_0, s_0 \in \{1, 2, \cdots, \text{cols}\}$ to get the 3D row EPI volumes with the size of $W \times H \times \text{rows}$. Then, we fix the spacial axis $u = u_0, u_0 \in \{1, 2, \cdots, W\}$ and perform the angular superresolution metalearning based network $F(\cdot)$ on each EPI $E_{u_0,s_0}(v, t)$ to get $E^*_{u_0,s_0}(v, t)$ with the size of $H \times \text{Rows}$:

$$E^*_{u_0,s_0}(v, t) = F\left(E_{u_0,s_0}(v, t), r\right), \tag{1}$$

where $r = (Rows - 1)/(rows - 1)$ is the magnification scale factor. The network $N_c(\cdot)$ is followed to recover the high-frequency details of $E^*_{u_0,s_0}(v, t)$ and obtain the LF_{inter} (u, v, s_0, t) with the size of $(W, H, \text{cols}, \text{Rows})$:

$$LF_{\text{inter}}(u, v, s_0, t) = N_c\left(E^*_{s_0}(u, v, t)\right). \tag{2}$$

The angular conversion is performed on $LF_{\text{inter}}(u, v, s_0, t)$, that is, the angular dimension is changed from t to s. First, by fixing the angular coordinate $t = t_0, t_0 \in \{1, 2, \cdots, \text{Rows}\}$, we get the 3D column EPI volumes with the size of $W \times H \times \text{cols}$; then, we extract column EPI by fixing the spacial axis $v = v_0, v_0 \in \{1, 2, \cdots, H\}$ and perform the angular superresolution network $F(\cdot)$ on to obtain superresolved $E^*_{v_0,t_0}(u, s)$, with the size of $W \times \text{Cols}$:

$$E^*_{v_0,t_0}(u, s) = F\left(E_{v_0,t_0}(u, s), r\right). \tag{3}$$

Finally, we use the network $N_r(\cdot)$ to recover the high-frequency details of $E^*_{v_0,t_0}(u, s)$ and obtain the $LF_{\text{out}}(u, v, s, t)$ with the size of $W \times H \times Cols \times Rows$:

$$LF_{\text{out}}(u, v, s, t) = N_r\left(E^*_{t_0}(u, v, s)\right). \tag{4}$$

We use a learnable architecture for both EPI angular superresolution module and EPI stack refinement module, so that the proposed framework can be trained in an end-to-end strategy.

3.3. Coarse EPI Angular Superresolution Module. As shown in Figure 3, for the angular superresolution of EPI, taking $E_{u_0,s_0}(v, t)$ as an example, the acquisition of $E^*_{u_0,s_0}(v, t)$ can be regarded as image superresolution on angular dimension t. We use a feature extraction module to learn the structure of the EPI and then upsample it to the desired resolution. In order to deal with arbitrary scale factor, we use the metalearning strategy in [36]. Since each pixel on the upsampled EPI is predicted via a local kernel. For different scale factors, both the number of the kernels and the weights of the kernels are different. We take advantage of the metalearning to predict the number of the kernels and the weights of the kernels for each scale factor.

Specifically, a feature learning module is settled to extract features from the low-resolution (LR) EPI; then, the upsample module dynamically predicts the weights of the upsampling filters by taking the magnification scale factor as input and using these weights to generate the high-

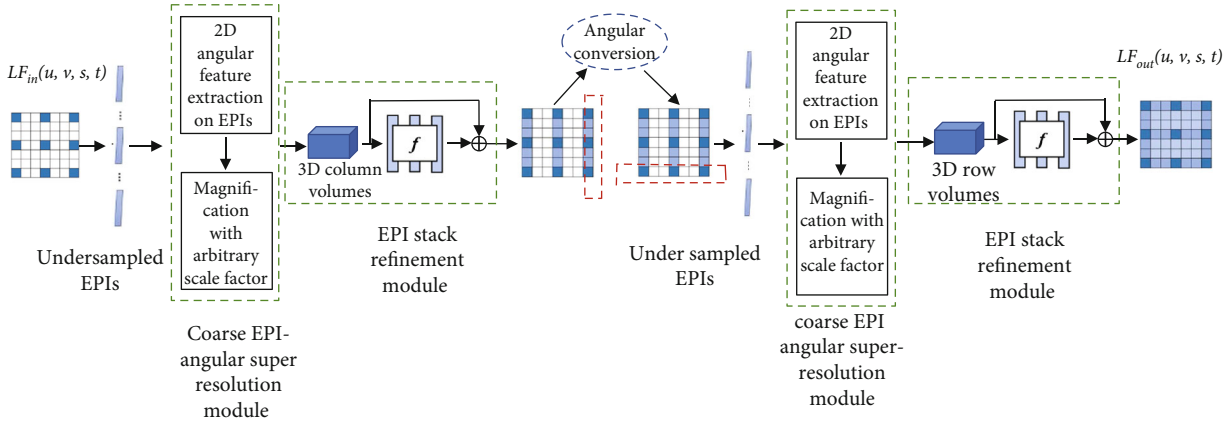


Figure 2: The flowchart of the proposed method for reconstructing a densely-sampled LF from an undersampled LF. Our proposed model consists of two phases, i.e., coarse EPI angular superresolution module and EPI stack refinement module.

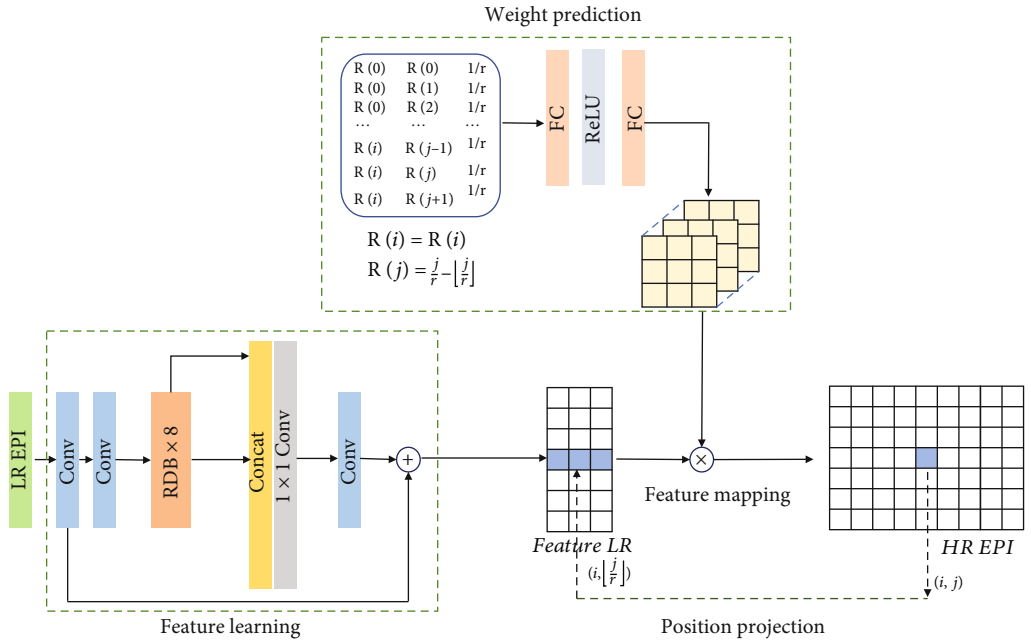


Figure 3: Coarse EPI angular superresolution module structure.

resolution (HR) EPI of arbitrary size. We modify the network to perform magnification only on the angular dimension in EPI.

For the feature learning module, we use the residual dense network (RDN) [37], which is composed of residual learning and dense connections. It consists of 3 layers of 2D CNN and 8 residual dense blocks (RDBs). Each RDB is composed of 8 layers of 2D CNN and ReLU with dense connections.

Taking $E_{u_0,s_0}(v,t)$ as an example, the resolution of the HR EPI is (H, Rows), and the feature is denoted as feature (H, rows). The upsample module can be regarded as the mapping function between EPI $E^*_{u_0,s_0}(v,t)$ with the size of $H \times \text{Rows}$ and feature(H, rows). This part is mainly composed of three steps, namely, position projection, weight prediction, and feature mapping. First, we determine the corresponding pixels across the spatial resolution through the position projection operation, and the specific implementation is as follows:

$$(H, \text{rows}) = T(H, \text{Rows}) = \left(H, \left\lfloor \frac{\text{Rows}}{r} \right\rfloor\right), \tag{5}$$

where T refers to the conversion function, r is the magnification scale factor, and $\lfloor\ \rfloor$ refers to the floor operation. For different magnification scale factors r, we use the weight prediction module to predict the corresponding prediction filter weight:

$$\text{Weight}(H, Rows) = \varphi(V_{H,\text{Rows}}; \theta), \tag{6}$$

$$V_{H,\text{Rows}} = \left(H, \frac{\text{Rows}}{r} - \left\lfloor \frac{\text{Rows}}{r} \right\rfloor\right). \tag{7}$$

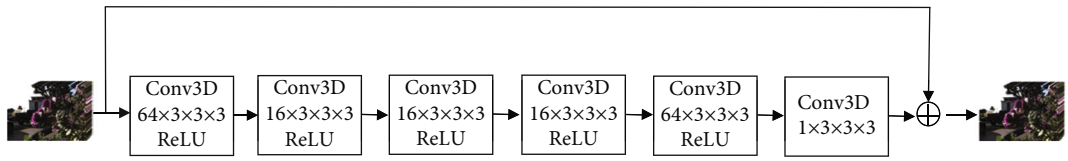


FIGURE 4: Structure of the network for recovering details of 3D volumes. The first 5 layers are followed by a rectified linear unit (ReLU). The final detail restored volume is the sum of the predicted residual and the input.

Among them, $\varphi(\cdot)$ represents the weight prediction network, θ is the network related parameters, and $V_{H,\text{Rows}}$ is the vector of the pixel on the output EPI. Finally, the feature mapping step maps feature(H, rows) with weight(H, Rows) to get the final HR EPI, and matrix multiplication was chosen as its function.

3.4. EPI Stack Refinement Module. The proposed networks $N_c(\cdot)$ and $N_r(\cdot)$ are shown in Figure 4. Both networks consist of an encoder and a decoder, where both of them comprise 3 convolution layers and are exactly symmetric. The first 3D convolutional layer comprises 64 channels with the kernel $3 \times 3 \times 3$, where each kernel operates on 3×3 spatial region across 3 adjacent EPIs. Similarly, the second layer comprises 16 channels with the kernel $3 \times 3 \times 3$. The last layer also comprises 16 channels with the kernel $3 \times 3 \times 3$. Each layer uses a stride of 1 followed by a rectified linear unit (ReLU), i.e., $\sigma(\text{x}) = \max(0, \text{x})$, excepting for the last one. The ultimate output of the network is the sum of the predicted residual and the input 3D EPI volume. To avoid border effects, we appropriately pad the input and feature maps before every convolution operation to maintain the input and output at the same size.

3.5. Perceptual Loss Function. As observed in prior work on image synthesis. Simply comparing the pixel colors of the synthesized image and the reference image could severely penalize perfectly realistic outputs. Instead, we adopt the perceptual loss. The basic idea is to match activations in a visual perception network that is applied to the synthesized image and separately to the reference image.

Let ϕ be a trained visual perception network (we use VGG-16 [38]). Layers in the network represent an image at increasing levels of abstraction: from edges and colors to objects and categories. Matching both lower-layer and higher-layer activations in the perception network guides the synthesis network to learn both fine-grained details and more global part arrangement. Here, g is the image synthesis network being trained and ¦È is the set of parameters of this network.

Let ϕ_1 be a collection of layers in the network ϕ, such that ϕ_1 denotes the input image. Each layer is a three dimensional tensor. For a training pair $(I, L) \in D$, our loss is

$$\iota_{I,L}(\theta) = \sum_l \|\phi_l(I) - \phi_l(g(L;\theta))\|_1. \tag{8}$$

Here, g is the image synthesis network being trained, and θ is the set of parameters of this network. For layers $\phi_l(l \geq 1)$, we use *conv*1_2, *conv*2_2, and *conv*3_2 in VGG-16 [21].

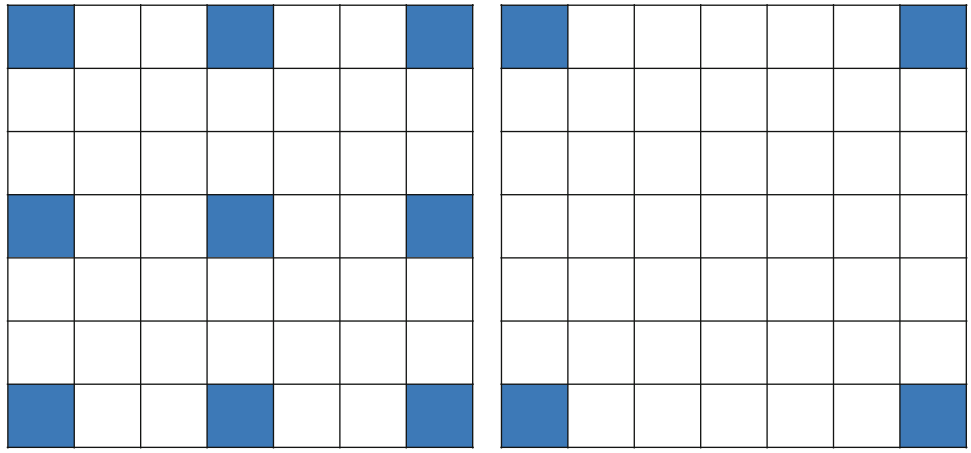
4. Experiments and Results

4.1. Datasets and Training Details. We took real-world LF images captured with a Lytro Illume camera provided by Stanford Lytro LF Archive [39] and Kalantari et al. [17] as well as synthetic LF images from the 4D light field benchmark [40, 41] to train and test the proposed framework. Specifically, 20 synthetic images and 100 real-world images were used for training. 70 LF images captured by Lytro Illum camera were used for real-world scenes test, including 30 test scenes provided by Kalantari et al. [17], 15 LF images from reflective [39] dataset, and 25 LF images from occlusions [39] dataset. 4 LF images from the HCI [41] dataset and 5 LF images from the old HCI [41] dataset were used for synthetic scenes test. We removed the border views and crop the original LF data to 7×7 views as ground truth and then downsampled randomly to 2×2 and 3×3 views as the input. For each LF data, small patches in the same position of each view were extracted to formulate the training LF data. The spatial patch size was 32×32 and the stride was 20.

Similar to other methods, we only processed the luminance Y channel in the YCbCr color space. The framework was implemented with PyTorch. The optimization of end-to-end training was ADAM optimizer with $\beta_1 = 0.9$ and $\beta_2 = 0.999$, and the batch size was set to 1. The learning rate was initially set to 10^{-4} and then decreased by a factor of 0.1 every 10 epochs until the validation loss converges. The filters of 3DCNNs were initialized from a zero-mean Gaussian distribution with standard deviation 0.01, and all the bias were initialized to zero.

4.2. Angular Superresolution Evaluation. We used the average value of peak signal-to-noise ratio (PSNR) and structural similarity index (SSIM) over all synthesized novel views in each scene to quantitatively measure the quality of reconstructed densely-sampled LFs. We designed two experiments to test the angular superresolution quality at different angular resolutions, respectively, reconstructing a 7×7 densely-sampled light field from 2×2 and 3×3 sparse views. Figure 5 demonstrates the sampling patterns.

Our experiments are compared with six state-of-the-art deep learning-based methods designed for densely-sampled light field angular superresolution, namely, Kalantari et al. [17], Wu et al. [18], Wu et al. [19], Wang et al. [35], Jin et al. [27], and Ran et al. [20]. Table 1 shows the properties of the above methods and our proposed method. For the



(a) (b)

Figure 5: Illustration of sampling patterns: (a) 3×3; (b) 2×2.

Table 1: Comparison of properties of different densely-sampled light field angular superresolution algorithms.

Methods	Based on deep learning	Based on geometry	Arbitrary scale factor	One model for various scale factor
Kalantari et al. [17]	√	√	√	−
Wu et al. [18]	√	−	−	−
Wu et al. [19]	√	√	√	−
Wang et al. [35]	√	−	−	−
Jin et al. [27]	√	√	√	−
Li et al. [20]	√	−	√	√
Ours	√	−	√	√

Table 2: Quantitative comparisons (PSNR/SSIM) of the proposed approach with the state-of-the-art ones under task 2×2 to 7×7 on synthetic scenes.

Methods	Kalantari et al. [17]	Wu et al. [18]	Wu et al. [19]	Li et al. [20]	Jin et al. [27]	Ours
HCI	32.85/0.909	26.64/0.744	31.84/0.898	33.14/0.910	34.60/0.937	33.92/0.916
HCI old	38.58/0.944	31.43/0.850	37.61/0.942	38.54/0.944	40.84/0.960	41.50/0.975
Average	36.03/0.928	29.30/0.803	35.05/0.922	36.14/0.928	38.07/0.949	38.13/0.949

compared algorithms, although some of them do support different scale factors [17, 19, 20, 27], they need to train different models for each scale factor separately, while our method is able to perform different scale factors with a single model.

For the task 2×2 to 7×7, we used the 9 synthetic LF images with angular resolution of 9×9, including 4 LF images from the HCI [41] dataset (bicycle bedroom, herbs, and dishes) and 5 LF images from the old HCI [41] dataset (Buddha Buddha2, StillLife, Papillon, and Mona). The central 7×7 views were extracted as ground truth, and 2×2 corner images were taken as input. We carried out comparison with the methods by Wu et al. [19], Kalantari et al. [17], Ran et al. [20], and Wanner and Goldluecke [23]. Table 2 shows the quantitative evaluation of the proposed approach on the synthetic dataset compared with the above methods.

On the two datasets, our proposed method provides an average angular superresolution advantage of 0.06 dB in terms of PSNR. On the old HCI dataset, our proposed method provides an average angular superresolution advantage of 0.66 dB in terms of PSNR and an advantage of 0.015 in terms of SSIM. On the HCI dataset, our method is inferior to Wanner and Goldluecke [23] since they used depth information to connect correspondence in views, but it is still better than other depth-dependent methods such as Kalantari et al. [17] and Wu et al. [19]. This is the cost for depth-free reconstruction. In general, our method has an absolute advantage in non-Lambertian scenes with smaller sampling baseline (the old HCI dataset), while it is also competitive on scenes with large sampling baseline (the HCI dataset). Therefore, we can say that our network has an advantage under task 2×2 to 7×7. We also provided visual comparisons of different methods, as shown in Figure 6. In contrast,

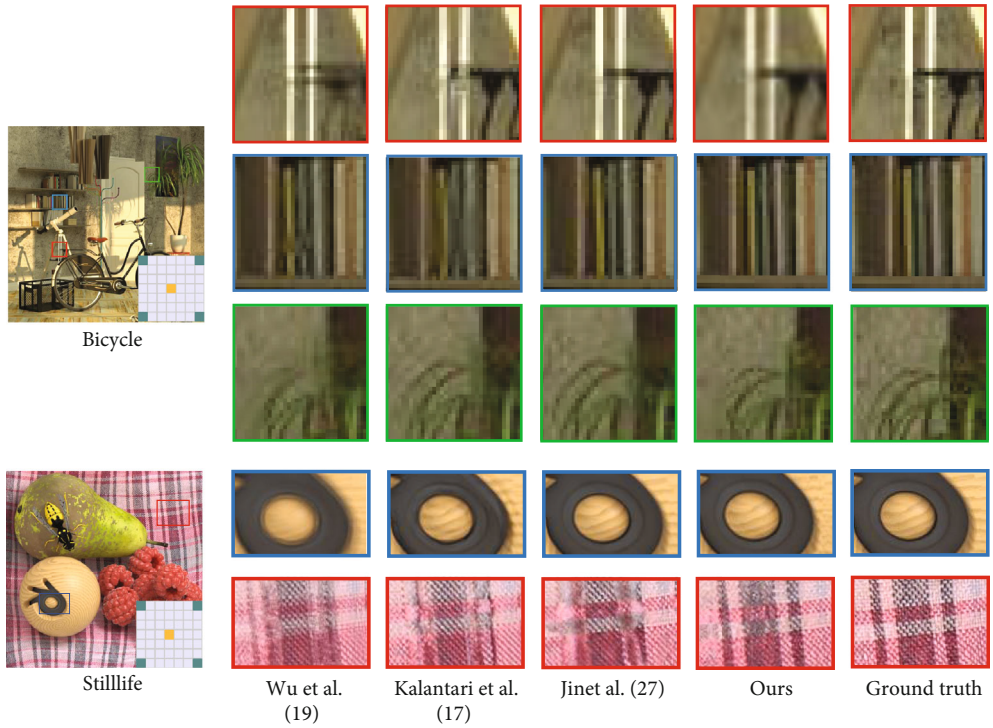


Figure 6: Visual comparison between Wu et al. [19], Kalantari et al. [17], Jin et al. [27], and our approach in synthetic scenes.

Table 3: Quantitative comparisons (PSNR/SSIM) of the proposed approach with the state-of-the-art ones under task 3×3 to 7×7 on real-world scenes.

Methods	Wu et al. [18]	Wang et al. [35]	Li et al. [20]	Ours
30scenes	41.02/0.988	43.82/*0.993*	43.83/0.993	*44.07*/0.988
Occlusions	39.80/0.981	41.23/0.983	41.89/0.984	*42.42/0.988*
Reflective	40.53/0.984	42.33/0.985	42.34/0.982	*42.36/0.986*
Average	40.48/0.985	42.58/*0.988*	42.82/0.987	*43.11/0.988*

our method produces high-quality images which are closer to the ground truth ones.

For the task 3×3 to 7×7, we used the real-world scene LF images with angular resolution of 7×7. We carried out comparison with the method by Wu et al. [18], Wang et al. [35], and Ran et al. [20].

As shown in Table 3, our proposed method performs better for all datasets than comparing methods: with 2.63 dB angular superresolution advantage over Wu et al. [18], 0.53 dB over Wang et al. [35], and 0.41 dB over Ran et al. [20] in terms of PSNR. Experimental results have further proven the advantages of our method.

4.3. Ablation Study. We conducted ablation experiments in terms of network structure. First, in order to verify the effectiveness of the two modules from our coarse-to-fine framework, we removed the EPI stack refinement module and tested only the coarse EPI angular superresolution module. Then, in order to maximize the reconstruction quality and find the optimal structure of the EPI stack refinement module, we evaluated three different structure solutions and selected the best one as the one used in this paper. The specific structure is shown in Figure 7. Scheme 1 is composed of 3 3D convolutional layers, the filter sizes are $5 \times 5 \times 3$, $1 \times 1 \times 3$, and $9 \times 9 \times 3$, and the numbers of output channels are 64, 32, and 1. Each convolutional layer is followed by a ReLU, excepting for the last layer. The filter bank learns the residuals, and the output of the last layer is added to the input as the final result. In general, scheme 1 is used as a residual block, and two residual blocks are connected to form a detail recovery module. The second scheme is an extension of scheme 1 and also refers to the idea of residual network. Scheme 2 is composed of 6 3D convolutional layers and the filter size is set to $3 \times 3 \times 3$. The third one is the method used in this paper. Based on the three schemes, the angular superresolution performance was tested. The specific experimental design was to combine the coarse EPI angular superresolution module with the three schemes.

The experiment was carried out on the HCI dataset, and the task is 2×2 to 7×7. Table 4 shows the experimental results. It can be seen that the angular superresolution quality of the EPI coarse angular superresolution network itself is on the high side. And the addition of the EPI stack refinement module of any scheme can increase the angular

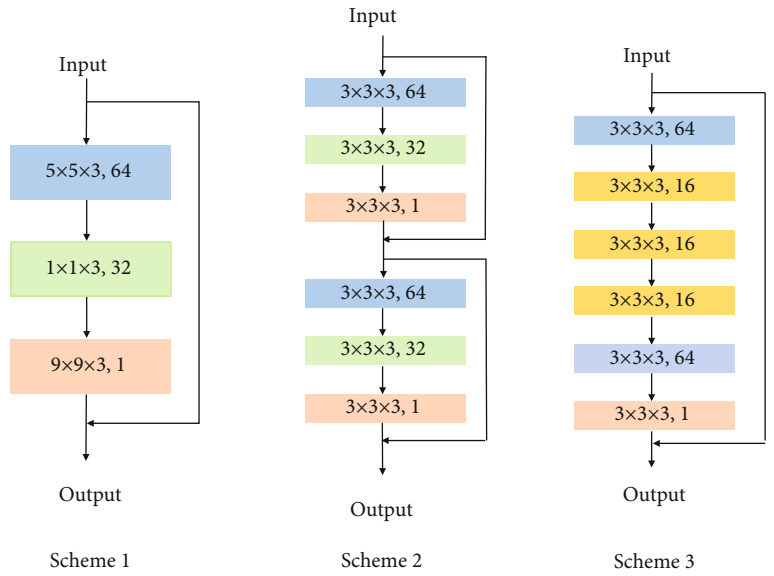


Figure 7: Three different schemes of EPI stack refinement module.

Table 4: Quantitative comparisons (PSNR/SSIM) on synthetic scenes with different EPI stack refinement module under task 2×2 to 7×7.

Methods	HCI
Only 2D EPI angular superresolution module	31.15/0.871
Coarse EPI angular superresolution module + solution 1	31.64/0.902
Coarse EPI angular superresolution module + solution 2	31.96/0.911
Coarse EPI angular superresolution module + solution 3 (ours)	33.92/0.916

superresolution performence on the HCI datasets. This shows that both the coarse EPI angular superresolution module and the EPI stack refinement module play a positive role in improving the quality of the reconstructed light field, and the design of the entire frame structure is reasonable and effective. Between these two modules, the main function of the coarse EPI angular superresolution module is to complete the magnification with any scale factor. The EPI stack refinement module plays an important role in the quality of the reconstructed view, which is responsible for the accurate generation of image textures and complex regions. For the EPI stack refinement module, among the three different structure solutions, scheme 3 has obvious advantages by 1.96 dB.

With the same training environment and training parameters, we used two different loss functions for training and observed the final training results. The first loss function is the edge enhancement loss function proposed in our previous work [20], and the second is the perceptual loss described in Section 3.5. The experiment was carried out on the HCI dataset and the old HCI dataset under the task 2×2 to 7×7. Table 5 shows the experimental results.

Table 5: Quantitative comparisons (PSNR/SSIM) on synthetic scenes with different loss functions under task 2×2 to 7×7.

Methods	Edge enhancement loss function	VGG-16 perception loss
HCI	33.31/0.910	33.92/0.916
HCI old	41.14/0.971	41.50/0.975
Average	37.66/0.944	38.13/0.949

It can be seen that training with perceptual loss has an advantage of 0.47 dB in terms of PSNR and 0.005 in terms of SSIM, compared with training with edge enhancement loss function. We believe that this is because the perceptual loss uses the features extracted by CNN as the object of the loss function, making the global structure and high-level semantics of the synthetic image and the ground truth closer. Compared with edge enhancement loss function, which pays more attention to texture information, perceptual loss performs better on angular superresolution tasks.

5. Conclusion

In this paper, an end-to-end coarse-to-fine framework is proposed to directly synthesize novel views of 4D densely-sampled LF from sparse input views. We combine magnification with arbitrary scale factor network for coarse EPI angular superresolution and 3DCNNs for EPI stack refinement, working in a coarse-to-fine manner and forming a pseudo 4DCNN which can make full use of 4D LF data while circumventing high computational burden. Our framework first reconstructs an intermediate LF by recovering EPI row volumes and then works on EPI column volumes to synthesize the final densely-sampled LF. In addition, metalearning is utilized to upsample EPIs in the coarse EPI angular superresolution module, which enables magnification with arbitrary scale factor with one model. We conducted ablation

experiments on the network structure and loss function, which proved the feasibility of our proposed network framework and the superiority of the perceived loss. Experimental results show that the proposed framework outperforms other state-of-the-art methods.

Data Availability

The light field image data supporting the findings of this study are from previously reported studies and datasets, which have been cited. The processed data are available from the corresponding author upon request.

Conflicts of Interest

The author(s) declare(s) that they have no conflicts of interest.

Acknowledgments

This work is supported by the National Natural Science Foundation of China (Grant Nos. 62001432 and 61971383), National Key R&D Program of China (Grant No. SQ2020YFF0426386), and the Fundamental Research Funds for the Central Universities (Grant Nos. CUC19ZD006 and CUC21GZ007).

References

[1] L. Marc and H. Pat, "Light field rendering," *ACM*, pp. 31–42, 1996.

[2] I. Ihrke, J. Restrepo, and L. Mignard-Debise, "Principles of light field imaging: briefly revisiting 25 years of research," *IEEE Signal Processing Magazine*, vol. 33, no. 5, pp. 59–69, 2016.

[3] J. Fiss, B. Curless, and R. Szeliski, "Refocusing plenoptic images using depth-adaptive splatting," in *IEEE International Conference on Computational Photography*, pp. 1–9, Santa Clara, CA, USA, 2014.

[4] C. Kim, H. Zimmer, Y. Pritch, A. Sorkine-Hornung, and M. H. Gross, "Scene reconstruction from high spatio-angular resolution light fields," *Association for Computing Machinery Transactions on Graphics*, vol. 32, no. 4, pp. 1–12, 2013.

[5] J. Chen, J. Hou, Y. Ni, and L. Chau, "Accurate light field depth estimation with superpixel regularization over partially occluded regions," *IEEE Transactions on Image Processing A Publication of the IEEE Signal Processing Society*, vol. 27, no. 10, pp. 4889–4900, 2018.

[6] H. Fu-Chung, C. Kevin, and W. Gordon, "The light field stereoscope," *ACM Transactions on Graphics*, vol. 34, no. 4, pp. 1–12, 2015.

[7] J. Chai, X. Tong, S. Chan, and S. Shum, "Plenoptic sampling," *SIGGRAPH conference*, pp. 307–318, 2000.

[8] Raytrix, "3d light field camera technology," http://www.raytrix.de/.

[9] T. Bishop and P. Favaro, "The light field camera: extended depth of field, aliasing, and superresolution," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 5, pp. 972–986, 2012.

[10] V. Boominathan, K. Mitra, and A. Veeraraghavan, *Improving Resolution and Depth-of-Field of Light Field Cameras Using a Hybrid Imaging System*, IEEE International Conference on Computational Photography, 2014.

[11] Penner and Eric, "Soft 3d reconstruction for view synthesis," vol. 36, Tech. Rep. 6, Association for Computing Machinery Transactions on Graphics, 2017.

[12] J. Flynn, I. Neulander, J. Philbin, and N. Snavely, *Deep Stereo: Learning to Predict New Views from the World's Imagery*, IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2016.

[13] K. Marwah, G. Wetzstein, Y. Bando, and R. Raskar, "Compressive light field photography using overcomplete dictionaries and optimized projections," vol. 32, Tech. Rep. 4, Association for Computing Machinery Transactions on Graphics, 2013.

[14] S. Lixin, H. Haitham, A. Davis, K. Dina, and D. Fredo, "Light field reconstruction using sparsity in the continuous Fourier domain," *ACM Transactions on Graphics*, vol. 34, no. 1, pp. 1–13, 2014.

[15] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Light field reconstruction using shearlet transform," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 1, pp. 133–147, 2018.

[16] S. Vagharshakyan, R. Bregovic, and A. Gotchev, "Accelerated shearlet-domain light field reconstruction," *IEEE Journal on Selected Topics in Signal Processing*, vol. 11, no. 7, pp. 1082–1091, 2017.

[17] N. K. Kalantari, T. C. Wang, and R. Ramamoorthi, "Learning-based view synthesis for light field cameras," *ACM Transactions on Graphics*, vol. 35, no. 6, pp. 1–193, 2016.

[18] G. Wu, M. Zhao, L. Wang, Q. Dai, T. Chai, and Y. Liu, *Light field reconstruction using deep convolutional network on epi*, IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2017.

[19] G. Wu, Y. Liu, Q. Dai, and T. Chai, "Learning sheared epi structure for light field reconstruction," *IEEE Transactions on Image Processing*, vol. 28, no. 7, pp. 3261–3273, 2019.

[20] R. Li, L. Fang, L. Ye, W. Zhong, and Q. Zhang, *Light field reconstruction with arbitrary angular resolution using a deep Coarse-To-Fine framework*, International Forum on Digital TV and Wireless Multimedia Communications, 2021.

[21] K. Simonyan and A. Zisserman, *Very deep convolutional networks for large scale image recognition*, International Conference on Learning Representati, 2014.

[22] G. Chaurasia, S. Duchene, O. Sorkine-Hornung, and G. Drettakis, "Depth synthesis and local warps for plausible image-based navigation," *Association for Computing Machinery Transactions on Graphics*, vol. 32, no. 3, pp. 1–12, 2013.

[23] S. Wanner and B. Goldluecke, "Variational light field analysis for disparity estimation and super-resolution," *IEEE Transac tions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 3, pp. 606–619, 2014.

[24] P. Srinivasan, T. Wang, A. Sreelal, R. Ramamoorthi, and R. Ng, *Learning to Synthesize a 4d Rgbd Light Field from a Single Image*, IEEE Computer Society, 2017.

[25] T. Zhou, R. Tucker, J. Flynn, G. Fyffe, and N. Snavely, "Stereo magnification," *ACM Transactions on Graphics*, vol. 37, no. 4, pp. 1–12, 2018.

[26] Mildenhall and Ben, "Local light field fusion," *Transactions on Graphics*, vol. 38, no. 4, pp. 1–14, 2019.

[27] J. Jin, J. Hou, H. Yuan, and S. Kwong, "Learning light field angular super-resolution via a geometry-aware network,"

Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, no. 7, pp. 11141–11148, 2020.

[28] L. Zhouchen and S. Heung-Yeung, "A geometric analysis of light field rendering," *International Journal of Computer Vision*, vol. 58, no. 2, pp. 121–138, 2004.

[29] A. Levin and F. Durand, "Linear view synthesis using a dimensionality gap light field prior," *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. 1831–1838, 2010.

[30] M. Yan, Z. Li, X. Yu, and C. Jin, "An End-to-End Deep Learning Network for 3D Object Detection From RGB-D Data Based on Hough Voting," *IEEE Access*, vol. 8, pp. 138810–138822, 2020.

[31] S. Yang, J. Wang, S. Arif, M. Jia, and S. Zhong, *SAL-Net: Self-supervised attribute learning for object recognition and segmentation*, Wireless Communications and Mobile Computing, 2021.

[32] Y. Yoon, H. G. Jeon, D. Yoo, J. Y. Lee, and I. S. Kweon, "Light-field image super-resolution using convolutional neural network," *IEEE Signal Processing Letters*, vol. 24, no. 6, pp. 848–852, 2017.

[33] G. Wu, Y. Liu, F. Lu, and T. Chai, "Lapepi-Net: a Laplacian pyramid epi structure for learning-based dense light field reconstruction," https://arxiv.org/abs/1902.06221, 2019.

[34] Y. Wang, L. Fei, Z. Wang, G. Hou, Z. Sun, and T. Tan, *End-to-End view synthesis for light field imaging with pseudo 4dcnn*, European Conference on Computer Vision, 2018.

[35] Y. Wang, F. Liu, K. Zhang, Z. Wang, Z. Sun, and T. Tan, "High-fidelity view synthesis for light field imaging with extended pseudo 4dcnn," *IEEE Transactions on Computational Imaging*, vol. 6, pp. 830–842, 2020.

[36] X. Hu, H. Mu, X. Zhang, Z. Wang, T. Tan, and J. Sun, *Meta-Sr: A Magnification-Arbitrary Network for Super-Resolution*, IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019.

[37] Y. Zhang, Y. Tian, and Y. Kong, *Residual Dense Network for Image Super-Resolution*, IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2018.

[38] Q. Chen and V. Koltun, *Photographic Image Synthesis with Cascaded Refinement Networks*, IEEE International Conference on Computer Vision, 2017.

[39] A. S. Raj, "Stanford lytro light field archive," http://lightfields.stanford.edu/.

[40] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke, *A Dataset and Evaluation Methodology for Depth Estimation on 4d Light Fields*, Asian Conference on Computer Vision Springer, 2016.

[41] S. Wanner, *Datasets and Benchmarks for Densely Sampled 4d Light Fields*, The Eurographics Association, 2013.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 8034435, 9 pages
https://doi.org/10.1155/2022/8034435

Research Article

MTAL: A Novel Chinese Herbal Medicine Classification Approach with Mutual Triplet Attention Learning

Wangli Hao,[1] **Meng Han**,[1] **Shancang Li**,[2] **and Fuzhong Li**[1]

[1]*School of Software, Shanxi Agricultural University, Shanxi, China*
[2]*Department of Computer Science at University of the West of England, Bristol BS16 1QY, UK*

Correspondence should be addressed to Fuzhong Li; lifuzhong@sxau.edu.cn

Received 22 September 2021; Revised 14 November 2021; Accepted 12 January 2022; Published 27 February 2022

Academic Editor: Ivan Lee

Copyright © 2022 Wangli Hao 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.

Chinese herbal medicine classification is a critical task in medication distribution and intelligent medicine, as well as a significant topic in computer vision. However, the majority of contemporary mainstream techniques are semiautomatic, with low efficiency and performance. To tackle this problem, a novel Chinese herbal medicine classification approach, Mutual Triplet Attention Learning (MTAL), is proposed. The motivation of our approach is to leverage a group of student networks to learn collaboratively and teach each other about cross-dimension dependencies throughout the training process, with the goal of quickly gaining strong feature representations and improving the outcomes. The results of the experiments show that MTAL outperforms other models in terms of accuracy and computation time. MTAL, in particular, improves accuracy by over 5.5 percent while reducing calculation time by over 50 percent.

1. Introduction

Two essential optimization goals for image classification, such as the Chinese herbal medicine (CHM) classification, are computational efficiency and classification accuracy. The CHM classification can be utilized for a variety of fields, including intelligent dispensing and pharmaceutical recommendations.

In order to perform Chinese herbal medicine classification and solve the high-dimensional nonlinear problem in the data of CHM, Luo et al. [1] developed a new approach that applying the linear discriminant analysis (LDA) and locally linear embedding algorithm (LLE) to conduct the CHM classification, whereas the dataset utilized in their paper is quite small that only includes six classes. To classify the leaves plant, Zhang et al. [2] applied a supervised local projection strategy and obtained promising results. Unger et al. [3] proposed a novel method that leverages the support vector machine (SVM) with morphometric and Fourier characteristics to perform the CHM classification based on two test datasets. These two datasets contain 17 and 26 classes separately, and each category has about 10 samples. Specifically, their method obtained the corresponding classification accuracy of 84% and 73.21% on two datasets, respectively. In the task of CHM classification, Luo et al. [4] compared two methods, PCA and SVM. And they found that SVM achieved better performance. Further, the SOM method has been applied to perform CHM classification [5]. Although promising performances have been achieved by these methods, the dataset they employed is quite small, and there are few samples in each category. However, these methods based on hand-crafted features with less robustness lead to poor classification performance. To achieve enhanced results, deep neural networks have been taken into consideration [6]. However, the model is large and has many parameters, which restricts their utilization in platforms or applications with fast execution or low memory demands, e.g., mobile phones.

Mutual learning is a practical method for achieving promising classification results, through tiny yet strong deep neural networks. The mutual learning begins with a group of students who synchronously learn to tackle the task together [7]. Concretely, each student is bound by two losses: a typical supervised learning loss, and a mimicry loss, which drives

each student's class probabilities to correspond with those of other students. Although mutual learning has obtained superior classification performance with less time consumption, it overlooks the interdependencies among spatial locations or channels.

Triplet attention is a simple yet effective attention strategy that may establish dependencies between weighted spatial or channels [8]. Triplet attention, in particular, utilizes a three-branch structure to calculate attention weights by capturing cross-dimension interaction. Triplet attention produces interdimensional relationships via the rotation operation and expresses spatial and interchannel information with low computational cost overhead when applied to an input tensor.

In order to further improve the Chinese herbal medicine classification performance in terms of accuracy and calculation time, this paper develops a novel Mutual Triplet Attention Learning (MTAL) approach by integrating the advantages of mutual learning and triplet attention. Specifically, MTAL allows two student networks to collaborate on parameter updates and learn interchannel and spatial dependencies from one another throughout the training process. These designs allow MTAL to achieve greater CHM classification efficiency and effectiveness by allowing the suggested model to gather more rich and robust features in a shorter amount of time.

In summary, our contribution can be listed as follows:

(i) To accomplish CHM classification, a novel Mutual Triplet Attention Learning (MTAL) technique is initially developed. MTAL's mutual learning component enables our model to get superior classification results while using significantly less computing time than previous models, in particular, with a 50% reduction in computing time and a 5.5% increase in accuracy. Furthermore, the MTAL's triplet attention unit allows our model to attain spatial and channel attention, which improves the CHM classification results by 5.5% over the model without the triplet attention component. Those advantages of MTAL will allow it to be applied in mobile devices to perform Chinese herbal medicine classification more efficiently and effectively

(ii) Several experiments have been designed and performed to verify the superiority of our model. Specifically, these experiments include the comparison of our MTAL model with different models, the evaluation of our del based on two identical basic student networks (two single ResNet18 or two single ResNet 50), and the evaluation of our MTAL model based on two distinct student works (one student network is ResNet18 and the other student network is ResNet50). Furthermore, MTAL has achieved promising CHM classification results with accuracy of 81.64%

The remainder of this paper is shown as follows. The materials and methods are described in Section 2. We list our experimental results in Section 3 and illustrate some discussions in Section 4. Finally, the conclusions are presented in Section 5.

2. Materials and Methods

2.1. Dataset. A CHM classification dataset (CHMC) with 100 classes (see some samples in Figure 1), proposed in our prior work [9], is utilized in the experiments. Specifically, each class in the dataset has a total of 100 samples. Among the dataset, 80% images are employed for training, and 20% images are utilized for testing. Hence, there are a total of 10000 images in the dataset, 2000 samples for testing and 8000 samples for training. Furthermore, the classes of medicinal materials in this dataset are relatively rich.

In particular, from the natural properties of medicinal materials, CHMC contains botanical, mineral, and animal medicines. Among them, botanical Chinese herbal medicine includes cockscomb, rice bud, hematoxylin, and cinnamon twig, respectively, animal medicine (sea cuttlebone, sea dragon, earth dragon, corrugated fruit, and scorpion), and mineral medicine (red stone fat, alum). Moreover, considering the medicinal part, CHMC contains roots (*Asarum*, ginseng), bark (cork, pomegranate peel), seeds (lotus seeds, wild jujube kernels, and orange cores), etc. In addition, the majority of the examples in CHMC have natural backgrounds, which can help with real-world applications.

2.2. The Proposed MTAL Model

2.2.1. MTAL Structure. The proposed MTAL model (see Figure 2) contains two student networks. They perform mutual learning by teaching each other interactively throughout the training stage, which will boost their performance. Furthermore, each student network integrates the triplet attention module, which can capture the cross-dimension interaction by calculating attention weights based on a three-branch structure.

2.2.2. Problem Formulation. The proposed MTAL model with two student networks is formulated as follows (as shown in Figure 2). Assume M samples $\mathcal{X} = \{x_i\}_{i=1}^{M}$ from C classes; the corresponding label set is denoted as $\mathcal{Y} = \{y_i\}_{i=1}^{M}$ with $y_i \in \{1, 2, \cdots, C\}$.

The predicted probability $p_1^c(x_i)$ of class c for sample x_i from $Network1$ is calculated as

$$p_1^c(x_i) = \frac{\exp\left(g_1^c(x_i)\right)}{\sum_{c=1}^{C} \boxtimes \exp\left(g_1^c(x_i)\right)}, \tag{1}$$

where the logit $g_1^c(x_i)$ is obtained from the "softmax" layer of $Network1$ for x_i.

The loss functions $\mathscr{L}_{\phi_1}$ and $\mathscr{L}_{\phi_2}$ for $Network1$ and $Network2$ can be formulated as follows:

$$\mathscr{L}_{\phi_1} = (1-\alpha) \times L_{l1} + \alpha \times D_{KL}(p_2 \| p_1)\ \mathscr{L}_{\phi_2} = (1-\alpha) \times L_{l2} + \alpha \times D_{KL}(p_1 \| p_2), \tag{2}$$

Figure 1: Samples of CHM dataset.

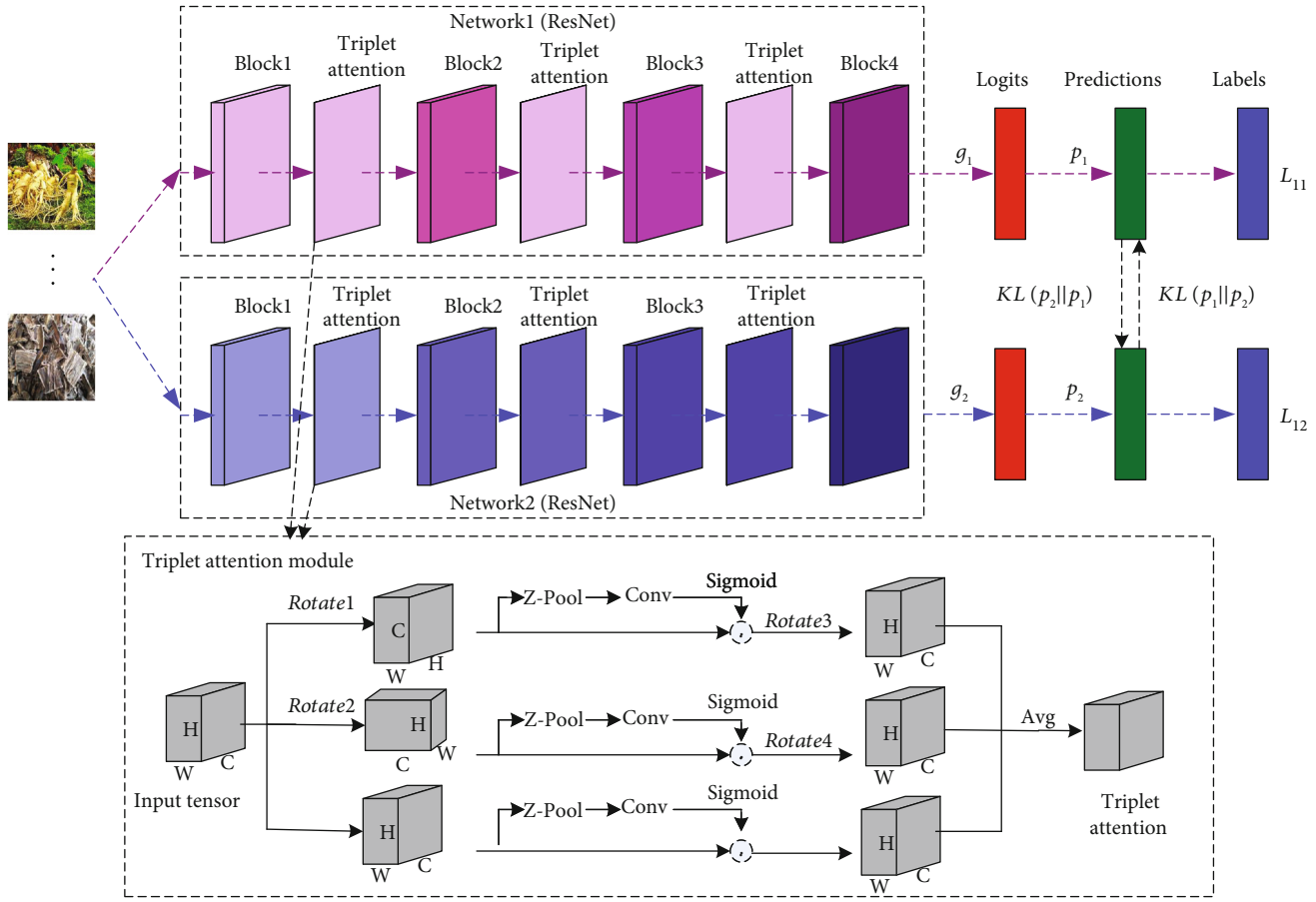


Figure 2: The framework of the MTAL model. The MTAL model contains two student networks. They perform mutual learning by teaching each other interactively throughout the training stage, which will boost their performance. Furthermore, each student network integrates the triplet attention module, which can capture the cross-dimension interaction by calculating attention weights based on a three-branch structure.

where L_{l1} and L_{l2} denote the conventional cross entropy losses in classification tasks and $D_{KL}(p_2||p_1)$ and $D_{KL}(p_1||p_2)$ indicate the Kullback-Leibler (KL) divergence losses. α indicates hyperparameters controlling the strengths of two loss terms. Among them, L_{l1} can be obtained by the following equation.

$$L_{l1} = -\sum_{i=1}^{M}\sum_{c=1}^{C} y_i^c \times \log\left(p_1^c(x_i)\right), \tag{3}$$

where y_i is the true label for x_i, y_i^c is an indicator, if $y_i = c$, $y_i^c = 1$, y_i=c, and $y_i^c = 0$, and L_{l1} denotes the cross entropy error between the correct labels and the predicted values, which can enforce the model to predict the correct results for the training samples.

To enhance the generalization capacity of *Netowork*1 on the testing samples, we employ another peer *Network*2 to offer training experience via its posterior probability p_2. In order to quantify the matching degree of the predictions p_1

and p_2, the KL divergence is utilized. $D_{KL}(p_2||p_1)$ indicates the KL distance from p_1 to p_2 and can be achieved through

$$D_{KL}(p_2||p_1) = \sum_{i=1}^{M} \boxtimes \sum_{c=1}^{C} \boxtimes p_2^c(x_i)^c \log \frac{p_2^c(x_i)}{p_1^c(x_i)}, \quad (4)$$

where $p_2^c(x_i)$ has a similar meaning to that of $p_1^c(x_i)$ of Equation (1).

L_{l2} and $D_{KL}(p_1||p_2)$ can be obtained by the similar ways with those of L_{l1} and $D_{KL}(p_2||p_1)$.

Considering the triplet attention module (TAM) in our MTAL method, they are integrated after each block of the corresponding ResNet. The purpose of TAM is to capture the cross-dimensional dependencies via rotating the input feature maps followed by residual transformation. Specifically, triplet attention contains three branches. The top branch is used to achieve the dependencies between the channel and the spatial dimensions (C and W). The middle branch is utilized to obtain the dependencies between the channel and the spatial dimensions (C and H). The bottom branch is employed to compute attention weights across the spatial dimensions (W and H). Finally, the simple average of the three-branch weights is utilized as the final weight.

*Rotate*1 and *Rotate*2 (see Figure 2) in TAM indicate the operations that rotate the input tensor 90° anticlockwise along the W and H axes, respectively. *Rotate*3 and *Rotate*4 represent the operations that rotate the input tensor 90° clockwise along the W and H axis, respectively.

The Z-pool layer in the TAM is employed for reducing the first dimensionality of the input tensor to two via concatenating the max pooled and average pooled features across the corresponding dimension. The Z-pool operation can be formulated as the following equation:

$$Z - \text{pool}(\mathcal{F}) = [\text{MaxPool}_{1d}(\mathcal{F}), \text{AvgPool}_{1d}(\mathcal{F})], \quad (5)$$

where $1d$ indicates the max and average pooling operations over the first dimension of the corresponding tensor.

The Conv layer in the TAM denotes the standard two-dimensional convolutional layers with kernel size k (k is an empirical parameter, specified 7 in our paper). The Sigmoid in the TAM indicates the sigmoid activation function, which has the following definition:

$$\text{sigmoid}(x) = \frac{1}{1 + \exp(-x)}. \quad (6)$$

2.2.3. Model Training. For a fair comparison, the models in the experiment uniformly adopt SGD to learn the parameters, with a batch size 32. The learning rate starts with 0.01 and then decreases to 1/10 per 80 epochs. The momentum is 0.9. α is 0.8. We stop our training at 200 epochs.

2.2.4. Evaluation Criteria. To compare the performance of different models, some evaluation criteria including accuracy, parameters, FLOPs [8], and loss are adopted. The accuracy reflects the performance of the model. Parameters denote the number parameters of the model; the smaller, the better. FLOPs indicate the floating point operations needed to be performed per second; the smaller, the better. Parameters and FLOPs illustrate the efficiency of the model. Note that all experiments are executed on TITANX GPUs.

3. Results

In this section, we will describe our experiments in detail. The proposed model's efficiency is investigated through five separate experiments, including the evaluation of the performance of MTAL, evaluation of the effectiveness of mutual learning, evaluation of the effectiveness of triplet attention, evaluation of the MTAL model based on the identical student networks, and evaluation of the MTAL model based on two distinct student networks.

3.1. Evaluation of the Performance of the MTAL Model. To explore the performances of the proposed models, we compare various popular deep neural networks. The comparison models include MobileNet [10], ResNet [11], and Xception71 [12], with results presented in Table 1. Table 1 illustrates that the MTAL model obtains the best results, outperforming MobileNetV2, MobileNetV3, ResNet18, ResNet50, and Xception71 by 15.39%, 11.76%, 10.25%, 6.16%, and 4%, which validates the superiority of the MTAL model in CHM classification task.

3.2. Evaluation of the Effectiveness of Mutual Learning. To evaluate the effectiveness of mutual learning, the single ResNets without mutual learning and the mutual learning models are utilized to perform comparisons. These models are simply represented as SigRes and MulRes, respectively. Specifically, SigRes$i$ indicates the ResNet model with ith layers, and MulRes$i$(j) denotes the ResNet$i$ model has received the knowledge from the other ResNet$j$ model during training. Comparison results are shown in Tables 2–4.

Tables 2 and 3 show the comparison of models with or without mutual learning based on two identical student networks. Table 2 shows the comparison of ResNet18, and Table 3 denotes the comparison of ResNet50, respectively. From these two tables, we can see that the models with mutual leaning obviously outperform those models without, with an accuracy increase of about 4.81% 5.13%. Those results validate the effectiveness of mutual learning.

Table 4 illustrates the comparison of models with or without mutual learning based on two distinct student networks, ResNet18 and ResNet50. From this table, we can see that the models with mutual leaning obviously outperform those models without, with an accuracy increase of about 4.94% and 6.41%. Furthermore, the small student network with mutual learning (MulRes18(50)) has achieved better results than the large network without mutual learning (SigRes50), with accuracy increase of 2.47% and parameter decrease of 53.98%, which further verifies the effectiveness and efficiency of mutual learning.

3.3. Evaluation of the Effectiveness of Triplet Attention. To evaluate the effectiveness of triplet attention, the ResNet models with and without triplet attention modules are utilized to perform comparisons. These models are simply

Table 1: Comparison of various models based on 100 types of CHM.

Models	MobileNetV2	MobileNetV3	ResNet18	ResNet50	Xception71	MTAL
Accuracy	70.75	73.05	74.05	76.90	78.5	81.64

Table 2: Comparisons of different models with or without mutual learning based on ResNet18.

Model	Parameters	FLOPs	Accuracy
SigRes18	11.69 M	1.82 G	74.05
MulRes18(18)-1	11.69 M	1.82 G	77.63
MulRes18(18)-2	11.69 M	1.82G	77.85

Table 3: Comparisons of different models with or without mutual learning based on ResNet50.

Model	Parameters	FLOPs	Accuracy
SigRes50	25.56 M	4.12 G	76.90
MulRes50(50)-1	25.56 M	4.12 G	80.40
MulRes50(50)-2	25.56 M	4.12 G	80.60

Table 4: Comparisons of different models with or without mutual learning based on ResNet18 and ResNet50.

Model	Parameters	FLOPs	Accuracy
SigRes18	11.69 M	1.82 G	74.05
SigRes50	25.56 M	4.12 G	76.90
MulRes18(50)	11.69 M	1.82 G	78.80
MulRes50(18)	25.56 M	4.12 G	80.70

represented as SigRes and SigAttRes, respectively. Specifically, SigRes*i*/SigAttRes*i* indicates the ResNet model with *i* th layers; comparison results are shown in Tables 5 and 6.

Tables 5 and 6 show the comparison of models with or without triplet attention based on ResNet18 and ResNet50, respectively. From these two tables, we can see that the models with triplet attention are obviously superior to those models without, with an accuracy increase of about 1.42% and 2.01%, which validates the effectiveness of triplet attention.

3.4. Evaluation of the MTAL Model Based on Two Identical Student Networks. In this section, we verify the performance of the proposed MTAL based on two identical student networks. Specifically, MTAL utilizes the ResNet backbones. Furthermore, the backbone network with different layers is employed to validate the generalization of the proposed approach, including ResNet18 and ResNet50 separately here.

The single convolutional neural network ResNet [11], single ResNet with triplet attention [8], and mutual learning model [7] based on the same ResNet backbone, respectively, are adopted to compare with the proposed MTAL with two identical basic student networks. These four models are simply named as SigRes, SigAttRes, MulRes, and MTALRes, respectively. Comparison results are shown in Figure 3.

Table 5: Comparisons of different models with or without triplet attention based on ResNet18.

Model	Parameters	FLOPs	Accuracy
SigRes18	11.69 M	1.82 G	74.05
SigAttRes18	11.69 M	1.83 G	75.10

Table 6: Comparisons of different models with or without triplet attention based on ResNet50.

Model	Parameters	FLOPs	Accuracy
SigRes50	25.56 M	4.12 G	76.90
SigAttRes50	25.56 M	4.17 G	78.45

Figure 3 illustrates the accuracy and loss of different comparison models based on two identical student networks, under different training epochs. Figures 3(a) and 3(b) show the comparison of accuracy of different models, and Figures 3(c) and 3(d) illustrate the comparison of loss of corresponding models in Figures 3(a) and 3(b). Specifically, in Figure 3, MulRes18(18)/MTALRes18(18) indicates the one ResNet18/ResNet18+triplet attention model learning experience from another model with the same architecture. For MulRes18(18)-*i*/MTALRes18(18)-*i*, *i* denotes the index of two mutual learning models. MulRes50(50)-*i* /MTALRes50(50)-*i* has a similar meaning to that of MulRes18(18)-*i*/MTALRes18(18)-*i*.

It is concluded that the results of SigAttRes18/SigAttRes50 exceed those of SigRes18/SigRes50 by 1.1%/2.01%, which validates the effectiveness of the triplet attention. MulRes18(18)/MulRes50(50) outperforms the corresponding Single ResNet models by 2.7%-5.1%, verifying the superiority of mutual learning. Moreover, both MTALRes18(18) and MTALRes50(50) obtain the best results in their corresponding counterparts at both performance and efficiency. Specifically, MTALRes18(18)/MTALRes50(50) surpasses SigRes18/SigRes50, SigAttRes18/SigAttRes50, and MulRes18(18)/MulRes50(50) by about 6.2%/6.1%, 4.7%/4.0%, and 1.3%/1.5% on accuracy and about 10%/19%, 78%/79%, and 80%/83% in terms of loss, respectively. The reason is that MTAL leverages the benefits of both mutual learning and triplet attention, which allows the network to learn cross-dimension dependencies from the other one.

Tables 7 and 8 summarize the comparisons of different models based on the corresponding evaluation criteria. The proposed MTAL models outperform SigRes, SigAttRes, and MulRes in terms of accuracy, with almost negligible parameters and FLOP increase (0.02% and 1%). In addition, MTALRes18(18) can acquire 2.3% better accuracy than

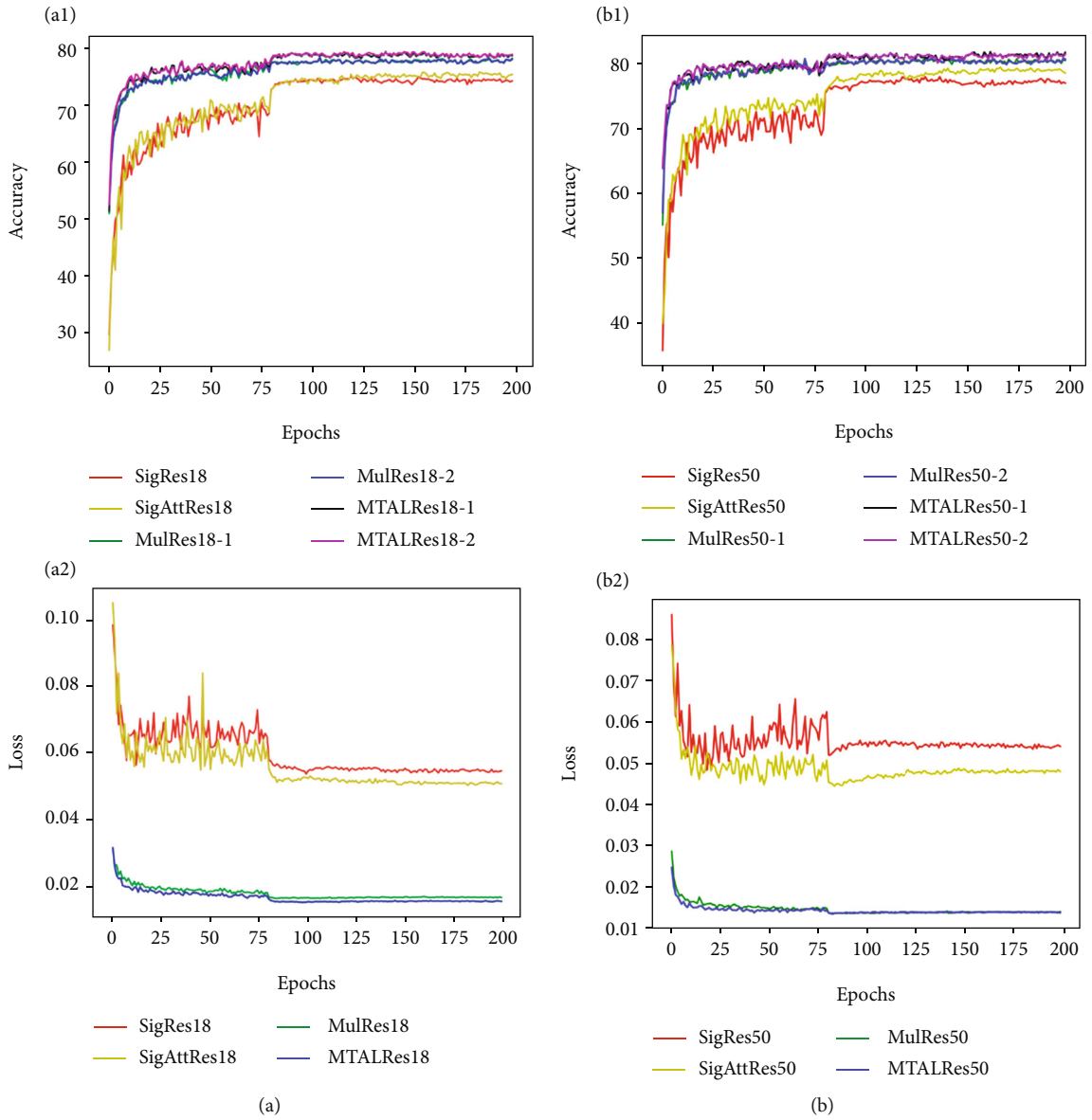


Figure 3: Comparison results of different models in terms of accuracy and loss based on the same basic network. (a) Comparison of accuracy of different models, including single ResNet18 model (SigRes18), single ResNet18+triplet attention (SigAttRes18), two single ResNet18 with knowledge learned from the other ResNet18 (MulRes18-1 and MulRes18-2), and two SigAttRes18 with knowledge learned from the other SigAttRes50 (MTALRes18-1 and MTALRes18-2). (b) Comparison of accuracy of different models, including single ResNet50 model (SigRes50), single ResNet50+triplet attention (SigAttRes50), two single ResNet50 with knowledge learned from the other ResNet50 (MulRes50-1 and MulRes50-2), and two SigAttRes50 with knowledge learned from the other SigAttRes50 (MTALRes50-1 and MTALRes50-2). (c) The comparison of losses of corresponding models of (a). (d) The comparison of losses of corresponding models of (b).

Table 7: Comparisons of different models for the classification of CHM based on the same basic student network ResNet18.

Model	Parameters	FLOPs	Accuracy
SigRes18	11.69 M	1.82 G	74.05
SigAttRes18	11.69 M	1.83 G	75.10
MulRes18(18)-1	11.69 M	1.82 G	77.63
MulRes18(18)-2	11.69 M	1.82 G	77.85
MTALRes18(18)-1 (ours)	11.69 M	1.83 G	78.50
MTALRes18(18)-2(ours)	11.69 M	1.83 G	78.70

Table 8: Comparisons of different models for the classification of CHM based on the same basic student network ResNet50.

Model	Parameters	FLOPs	Accuracy
SigRes50	25.56 M	4.12 G	76.90
SigAttRes50	25.56 M	4.17 G	78.45
MulRes50(50)-1	25.56 M	4.12 G	80.40
MulRes50(50)-2	25.56 M	4.12 G	80.60
MTALRes50(50)-1(ours)	25.56 M	4.17 G	81.64
MTALRes50(50)-2(ours)	25.56 M	4.17 G	81.35

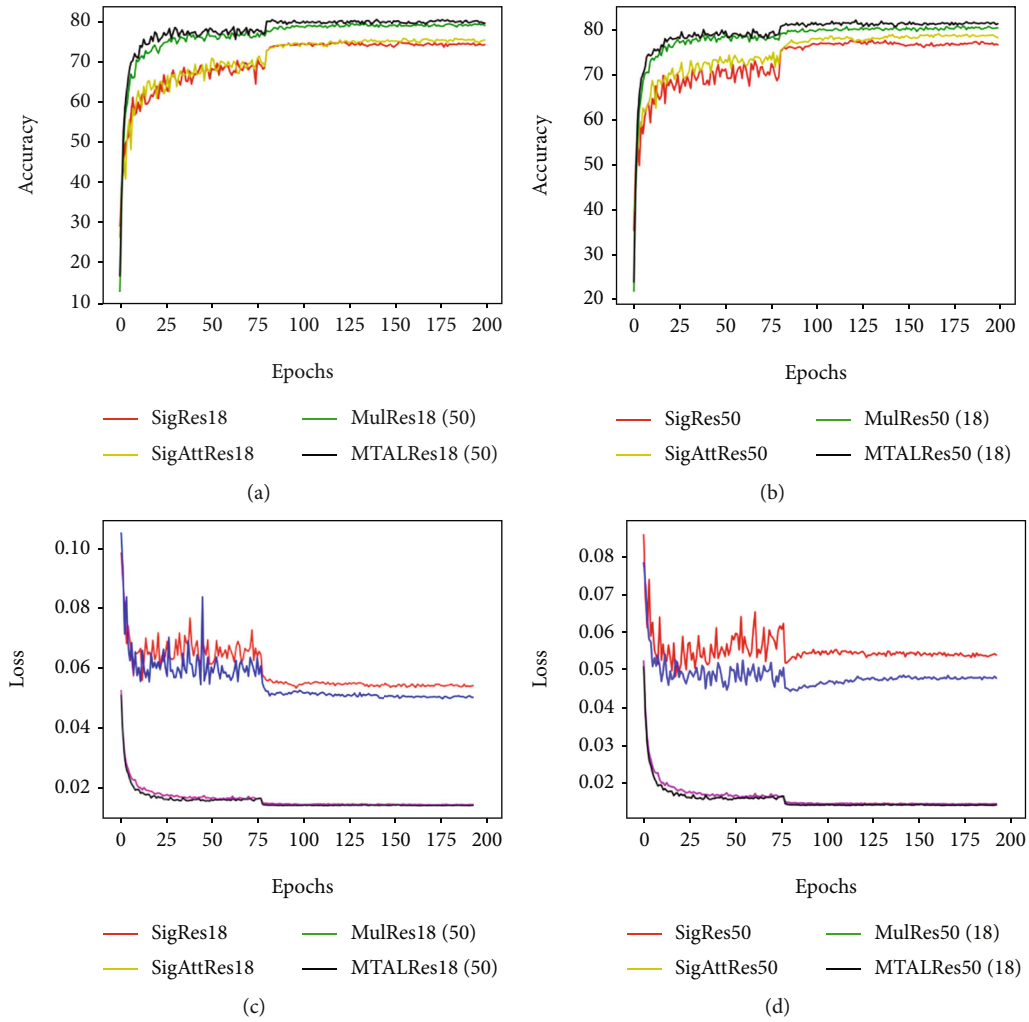


Figure 4: Comparison results of different models in terms of accuracy and loss based on the different basic student networks. (a) Comparison of accuracy of different models, including single ResNet18 model (SigRes18), single ResNet18+triplet attention (SigAttRes18), single ResNet18 with knowledge learned from ResNet50 (MulRes18(50)), SigAttRes18 with knowledge learned from SigAttRes50 (MTALRes18(50)). (b) Comparison of accuracy of different models, including single ResNet50 model (SigRes50), single ResNet50+triplet attention(SigAttRes50), single ResNet50 with knowledge learned from ResNet18 (MulRes50(18)), and SigAttRes50 with knowledge learned from SigAttRes18 (MTALRes50(18)). (c) The comparison of loss of the corresponding models of (a). (d) The comparison of loss of the corresponding models of (b).

Table 9: Comparisons of different models for the classification of CHM based on the basic network ResNet18.

Model	Parameters	FLOPs	Accuracy
SigRes18	11.69 M	1.82 G	74.05
SigAttRes18	11.69 M	1.83 G	75.10
MulRes18(50)	11.69 M	1.82 G	78.80
MTALRes18(50)(ours)	11.69 M	1.83 G	79.35

SigRes50 and achieve comparable accuracy with SigAttRes50, but its parameters and FLOPs are significantly lower than those of SigRes50 and SigAttRes50 (over 2 times less). Moreover, although MTALRes50(50) needs comparable parameters and FLOPs with other models with the same backbone, it achieves better results.

3.5. Evaluation of the Model Based on Two Distinct Student Networks. In this section, we validate the performance of the proposed MTAL based on two distinct student networks. Specifically, the two student networks are ResNet18 and ResNet50 separately.

The single convolutional neural network ResNet [11], single ResNet with triplet attention [8], and mutual learning model [7] based on different ResNet backbones, respectively, are adopted to compare with the proposed MTAL with different basic student neural networks. These four models are simply named as SigRes, SigAttRes, MulRes, and MTALRes, respectively. MulRes $i(j)$ denotes the model ResNet$i$ with knowledge learned from the model ResNet$j$. MTALRes$i(j)$ has a similar meaning to that of MulRes $i(j)$. Comparison results are presented in Figure 4.

Figure 4 shows the accuracy and loss of different comparison models based on two distinct student networks,

Table 10: Comparisons of different models for the classification of CHM based on the basic network ResNet50.

Model	Parameters	FLOPs	Accuracy
SigRes50	25.56 M	4.12 G	76.90
SigAttRes50	25.56 M	4.17 G	78.45
MulRes50(18)	25.56 M	4.12 G	80.70
MTALRes50(18)(ours)	25.56 M	4.17 G	81.35

under different training epochs. Figures 4(a) and 4(b) show the comparison of accuracy of different models, and Figure 4(c) and 4(d) illustrate the comparison of loss of corresponding models in Figures 4(a) and 4(b).

From Figure 4, we can see that SigAttRes18/SigAttRes50 exceeds SigRes18/SigRes50 by 1.42%/2.01%, which validates the effectiveness of the triplet attention. MulRes18(50)/MulRes50(18) outperforms the corresponding single ResNet (SigRes18/SigRes50) by 6.41%/4.94%, verifying the superiority of mutual learning (even the smaller student can further boost the larger one student). Moreover, both MTALRes18(50) and MTALRes50(18) obtain the best results in their corresponding counterparts at both performance and efficiency. Specifically, MTALRes18(50)/MTALRes50(18) surpasses SigRes18/SigRes50, SigAttRes18/SigAttRes50, and MulRes18(50)/MulRes50(18) by 7.16%/5.79%, 5.65%/3.70%, and 0.69%/0.80% on accuracy and about 9.1%/9.7%, 77.5%/79.2%, and 80.2%/80.7% in terms of loss, respectively, which validates the effectiveness of the proposed model.

Tables 9 and 10 summarize the comparisons of different models based on the corresponding evaluation criteria. The proposed MTAL models outperform SigRes, SigAtt, and MulRes in terms of accuracy, with almost negligible parameters and FLOPs increase (0.02% and 1%). Additionally, MTALRes18(50) can acquire 3.18% and 1.15% better accuracy than those of SigRes50 and SigAttRes50, but its parameters and FLOPs are significantly lower than SigRes50 and SigAttRes50 (over 2 times less). Moreover, although MTALRes50(18) needs comparable parameters and FLOPs with other models with the same backbone (SigAttRes50), it achieves better results.

4. Discussion

The proposed MTAL leverages both mutual learning and triplet attention modules, which may gain cross-dimension knowledge of the spatial dimensions and channel dimension from them and the other student network in an interactive manner. This interactive learning and the cross-dimension dependency capturing capacity allows our model MTAL to achieve promising performance at both accuracy and efficiency. In order to validate the superiority of MTAL, several experiments are designed and conducted, including evaluation of the performance of the MTAL model, evaluation of the MTAL model based on two identical student networks, and evaluation of the MTAL model based on two distinct student networks.

When compared to other popular models, such as MobileNetV2, MobileNetV3, ResNet18, ResNet50, and Xception71, MTAL achieves state-of-the-art performance, demonstrating MTAL's efficacy.

The MTAL model obtains better performance and efficiency when comparing with their corresponding counterparts, including SigRes, SigAttRes, and MulRes, respectively, which validates the generalization performance of our model.

Through triplet attention-enhanced mutual learning, the small student network of our model can even obtain better performance than those of large student networks. For example, MTALRes18(50) achieves 3.18% and 1.15% better accuracy than those of SigRes50 and SigAttRes50, saving even two times complexity and time cost. On the other hand, when the scale of the network is the same with other networks, our model obtains better performance than other models.

In conclusion, our model outperforms the competition in terms of accuracy and efficiency in the CHM classification task.

5. Conclusions

This paper has developed a novel MTAL approach for CHM classification, which combines mutual learning with the triplet attention module for transferring the cross-dimension dependencies from one network to another one. With the help of deep mutual learning, an ensemble of basic student neural networks of our model can update parameters collaboratively and gain information from each other during the whole training process. As a benefit from the triplet attention module, our model can collect interdimensional information via the rotation operation and accomplish interchannel and spatial dependencies with nearly no increase in computational overhead. Leveraging the mutual learning and triplet attention module, our MTAL model has achieved excellent classification performance of Chinese herbal medicines with higher efficiency and effectiveness. When compared to other models, the experimental findings show that MTAL can greatly improve CHM classification performance with minimally compromised training settings and FLOPs. The MTAL model, in particular, delivers 3.18% greater accuracy and 80.7% lower loss than other models.

Further work would explore more efficient mutual learning-based methodologies and a more promising attention-based feature extraction approach to boost the effectiveness and the efficiency of the CHM classification.

Data Availability

The datasets used to support the findings of this study can be obtained by sending email to the corresponding author.

Conflicts of Interest

The authors declare no conflicts of interest.

Acknowledgments

This work was supported by the Shanxi Province Higher Education Innovation Project of China (2020L0154), Shanxi Province Educational Science "14th Five-Year Plan" 2021 General Planning Project + Research on Intelligent Agricultural Talent Cultivation Model Driven by "Industry, Learning and Research" of Agricultural and Forestry Colleges (GH-21006), Shanxi Agricultural University 2021 "Neural Network" Course Ideological and Political Project (KCSZ202133), Shanxi Agricultural University Doctoral Research Startup Project (2021BQ88), Intelligent Information Processing, Shanxi Provincial Key Laboratory Open Project Fund (CICIP2021005), and Shanxi Agricultural University Academic Recovery Research Project (2020xshf38). We thank Jian zhou for helpful conversations, Dr. Li for assistance with the samples, and Zhiwu Dong for producing several of the figures.

References

[1] J. Wang, R. K. W. Wong, and T. C. M. Lee, "Locally linear embedding with additive noise," *Pattern Recognition Letters*, vol. 123, pp. 47–52, 2019.

[2] S. Zhang, Y. Lei, T. Dong, and X. P. Zhang, "Label propagation based supervised locality projection analysis for plant leaf classification," *Pattern Recognition*, vol. 46, no. 7, pp. 1891–1897, 2013.

[3] J. Unger, D. Merhof, and S. Renner, "Computer vision applied to herbarium specimens of german trees: testing the future utility of the millions of herbarium specimen images for automated identification," *BMC Evolutionary Biology*, vol. 16, no. 1, p. 248, 2016.

[4] D. Luo, W. Jia, Y. Chen, and G. H. Hamid, "Classification of chinese herbal medicines based on svm," in *2014 International Conference on Information Science, Electronics and Electrical Engineering*, vol. 1, pp. 453–456, Sapporo, Japan, April 2014.

[5] M. Wang, L. Li, C. Yu et al., "Classification of mixtures of chinese herbal medicines based on a self-organizing map (som)," *Molecular Informatics*, vol. 35, no. 3-4, pp. 109–115, 2016.

[6] S. Liu, W. Chen, and X. Dong, "Automatic classification of chinese herbal based on deep learning method," in *2018 14th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD)*, Huangshan, China, July 2018.

[7] T. H. T. L. H. Zhang and Y. Xiang, "Deep mutual learning," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4320–4328, Salt Lake City, UTAH USA, 2018.

[8] D. Misra, T. Nalamada, A. U. Arasanipalai, and Q. Hou, "Rotate to Attend: Convolutional Triplet Attention Module," in *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, Waikoloa, HI, USA, January 2021.

[9] W. Hao, M. Han, H. Yang, F. Hao, and F. Li, "A novel chinese herbal medicine classification approach based on efficientnet," *Systems Science and Control Engineering*, vol. 9, no. 1, pp. 304–313, 2021.

[10] A. Howard, M. Sandler, G. Chu et al., "Searching for mobilenetv3," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 1314–1324, Seoul, Korea (South), October 2019.

[11] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, Las Vegas, NV, USA, June 2016.

[12] F. Chollet, "Xception: deep learning with depthwise separable convolutions," in *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 8354150, 16 pages
https://doi.org/10.1155/2022/8354150

Research Article

ALBLP: Adaptive Load-Balancing Architecture Based on Link-State Prediction in Software-Defined Networking

Junyan Chen,[1,2] **Yong Wang**,[2] **Xuefeng Huang**,[2] **Xiaolan Xie**,[2] **Hongmei Zhang**,[1] **and Xiaoye Lu**[2]

[1]*School of Information and Communication, Guilin University of Electronic Technology, Guilin 541004, China*
[2]*School of Computer Science and Information Security, Guilin University of Electronic Technology, Guilin 541004, China*

Correspondence should be addressed to Yong Wang; ywang@guet.edu.cn

Received 11 October 2021; Accepted 20 January 2022; Published 12 February 2022

Academic Editor: Ming Yan

Copyright © 2022 Junyan Chen 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-balancing optimization in software-defined networking (SDN) has been researched for a long time. Researchers have proposed many solutions to the load-balancing problem but have rarely considered the impact of transmission delay between controllers and switches under high-load network conditions. In this paper, we propose an adaptive load-balancing architecture based on link-state prediction (ALBLP) in SDN that can solve the influence of transmission delay between controllers and switches on network load balancing. ALBLP constructs the prediction model of the network link status, adopts the long-term and short-term memory neural network (LSTM) algorithm to predict the network link-state value, and then uses the predicted value as the Dijkstra weight to calculate the optimal path between network hosts. The proposed architecture can adaptively optimize network load balancing and avoid the empty window period, in which the switch flow table does not exist by actively issuing the flow table. Under the network architecture, we collect the data set of the network link-state by simulating the GÉANT network, and we verify the effectiveness of the proposed algorithm. The experiment results show that the ALBLP proposed in this paper can optimize load balancing in SDN and solve the problem of transmission delay between controllers and switches. It has a maximum load-balancing improvement of 23.7% and 11.7% in comparison with the traditional Open Shortest Path First (OSPF) algorithm and the reinforcement learning method based on Q-Learning.

1. Introduction

In recent years, we have seen the rapid development of the Internet and an increasing growth in the prominence of its scale. The popularity of social media, high-definition movies, and online games has caused network traffic to grow rapidly. In this vast traffic data environment, traditional networks are confronted with even more severe challenges. Software-defined networking (SDN) is a new network-architecture revolution when compared to a traditional computer network. In SDN, due to the decoupling of the data plane and the control plane and due to a centralized control strategy, the artificial intelligence method can be directly deployed in the control plane and in the application plane [1].

The existing widely used open shortest path first (OSPF) or equal-cost multi-path (ECMP) routing methods in the current SDN structure calculate the shortest path or multiple optimal forwarding paths based on static local link information [2–4]. Because of lacking the intelligence to learn from previous experiences, these methods can lead to the repetition of wrong decisions when similar traffic patterns happen. The lousy path decision results in network congestion, which leads to further performance deterioration and cannot meet the global load-balancing requirements. Meanwhile, making routing decisions requires obtaining network link status information [5]. However, there are delays in the process of the controller periodically acquiring network link information. These include (i) the process delay of the switch processing the data packet; (ii) the link delay of the switch transmitting the data packet; and (iii) the command delay of the secure channel between the controller and the switch. The existence of these delays causes a difference

between the network link status information acquired by the controller and the current. After these delays, the current link-state value may have changed, causing the issued flow table not to use network resources effectively. Furthermore, it may increase network congestion under high-load conditions. Hence, it is necessary to obtain real-time network link status information to make forwarding decisions more accurate and improve the performance of network load balancing.

In this paper, we propose an adaptive load-balancing architecture based on dynamic link-state prediction (ALBLP) in SDN that can solve the influence on network load balancing of transmission delay between controllers and switches. ALBLP predicts the network link-state value through an improved long-term and short-term memory neural network (LSTM) model and then uses the predicted value as the Dijkstra weight to calculate the optimal path between network hosts and proactively issue the flow table. The data of the network status used for decision-making in the prediction model includes the transmission delay between the controllers and switches, thus reducing the effect of transmission delay.

The main contributions of this paper are as follows:

(1) Effective simulation experiments identify the influence of transmission delay between controllers and switches on network load balancing under high-load network conditions

(2) A prediction model of network link status is constructed to optimize network load balancing under high-load network conditions

(3) An adaptive load-balancing architecture based on link-state prediction in SDN is proposed that can solve the influence of transmission delay between controllers and switches on network load balancing

(4) A prototype system for the application of the proposed architecture is designed to verify its effectiveness

The rest of this paper is organized as follows. In Section 2, we discuss related research. Section 3 introduces the network architecture and optimization model. Section 4 describes the SDN load-balancing algorithm based on link-state prediction. Section 5 describes our evaluation and experimental results in detail. And finally, Section 6 concludes the paper and offers future directions for further investigation.

2. Related Research

In network management, accurate and timely network traffic data is essential for many downstream tasks. The characteristics of network traffic are that it is self-similar, multiscalar, long-term dependent, and highly nonlinear. These statistical characteristics determine the traffic predictability [6]. The traditional Poisson or Gaussian distributions cannot build the predictive model. The authors in [7] used Taylor's equation to predict network traffic changes, but its prediction effect was not effective, and its generalization ability was weak. With today's vigorous development of artificial intelligence technology, researchers have used machine learning algorithms to construct data prediction models in other fields. The authors in [8, 9] propose a position prediction model based on a weighted Markov model. The authors in [10] use an improved LSTM method to achieve better results in traffic flow prediction. The authors in [11] provided ultimate user experience to next generation mobile users by utilizing big data network analytics to predict users' service usage patterns and trajectories. The authors in [12] used the regression function of Convolutional Neural Network (CNN) to realize a polar code decoding structure.

Other researchers have used machine-learning methods in SDN. Azzouni and Pujolle [13] and Novaes et al. [14] verified the feasibility of LSTM in network traffic prediction. However, they only predicted traffic forwarding data between hosts. In an actual network, the number of hosts is often far greater than the number of switches, and so predicting traffic data between hosts plays a limited role in optimizing the data forwarding of switches. The work in [15] established LSTM models to predict the load of the SDN controller–having a significant effect on the load balance of the controller. The authors in [16] use the cache method based on LSTM to improve the cache hit rate, which can effectively improve the throughput of the data center. However, the network load-balancing problem of the data plane in SDN has not been solved.

In recent years, several studies have been carried out on network load balancing in the data plane. The authors in [17] propose a fuzzy comprehensive evaluation mechanism (FSEM), which dynamically selects paths to transmit network traffic according to changes in network status. The authors in [18] use a switch migration-based decision-making (SMDM) scheme to improve migration efficiency. The authors in [19] propose a traffic-scheduling method based on the ant colony algorithm. The authors in [20] capitalize on the merit of a fast global search of GA and efficient tracking of an optimal solution of ACO to enhance the performance of SDN. The authors in [21] propose a dynamic, multipath Dijkstra forwarding algorithm to optimize routing policy. The optimization effects and robustness of traditional algorithms for network load balancing are not ideal due to the limited perception of the network. Traditional algorithms cannot gather the network resource information from multiple dimensions such as time and space. Therefore, this often leads to local network congestion after distributing the flow table.

Methods based on deep learning (DL) or reinforcement learning (RL) can optimize network load balancing in multiple dimensions in the form of experience. The authors in [22, 23] propose a routing strategy that utilizes CNN to choose the paths combinations according to the network traffic trace in an online fashion. This strategy can avoid congested paths and balance the network traffic. However, it does not consider the influence of time dimension in a network. The authors in [24] propose the Reinforcement Learning and Software Defined Networking Intelligent Routing (RSIR) algorithm based on Q-learning, which optimizes the network in three dimensions: throughput,

delay, and packet loss rate. However, its Q-learning method requires significant time to train the Q-table between each host pair after collecting link information. It has no learning ability for historical network information and has high requirements for the controller's performance. The authors in [25] propose routing based on deep reinforcement learning (DRL-R). The authors in [26] propose a multi-controller routing planning algorithm based on a deep Q-learning network (DQN). The authors in [27] adopt the Deep Deterministic Policy Gradient (DDPG) algorithm to find the optimal scheduling scheme for flows. They achieved strong improvement in network load balancing, throughput, and QoS optimization. The authors in [28] proposed a real-time distributed learning method, which uses a multiagent deep reinforcement learning (MADRL) model to realize the independent routing decision of each edge node. This method overcomes the weakness of conventional turn-based DRL methods and increases delivered packet ratios and effective throughput. But the RL algorithm must constantly obtain network link status data as its environmental state, and so the transmission delay between the controllers and the switches may cause a difference between the collected data and the actual data.

There is still a lack of relevant research that considers the impact of transmission delay between controllers and switches on network load balancing. This delay cannot be ignored in the real network because the network load-balancing algorithm depends on the real-time network state. In our paper, we propose an adaptive load-balancing architecture based on link-state prediction in SDN that can solve this problem.

3. Network Architecture and Optimization Model

This section describes the overall architecture of the design and builds a model for the load-balancing optimization objectives.

3.1. Network Architecture. According to the application scenario of SDN, we designed a network architecture as shown in Figure 1. We used the OpenFlow 1.3 protocol to simulate the SDN network. The description of each part of the specific network architecture is as follows:

3.1.1. Data Plane. The data plane consists of hosts and SDN switches, and it uses the standard OpenFlow protocol to connect the control plane through the SDN switch.

3.1.2. Control Plane. The control plane comprises SDN controllers, and it is an intermediate network component that controls the data plane and transmits data to the application plane.

The communication between the controller and the switch includes the following:

(i) Establishing the connection between the controller and the switch

(ii) The controller requesting information from the switch

(iii) The switch sending packet_in messages to the controller without matching flow entries

(iv) The controller issuing and updating the flow entries to the data plane

In this paper, the control plane is divided into the following modules:

(i) The network topology discovery module: the controller periodically sends a connection message to the switch to detect the network topology through this module.

(ii) The flow monitoring module: this module regularly obtains the network state information and provides a fundamental basis for decisions of the load-balancing algorithm.

(iii) The flow entry distribution module: we adopt the method of actively issuing flow entries to periodically update flow entries based on the network status and the load-balancing algorithm in this module. Flow entries are actively delivered by the controller periodically generating a data packet forwarding strategy according to the acquired network status and sending it to all switches. When we compare it with passively sending flow entries, we see that active delivery will not generate a considerable window period when the flow entry does not exist. Figure 2 shows the difference between issuing a flow table actively or passively.

(iv) The message processing module: this module monitors and handles the packet_in message uploaded by the switch.

3.1.3. Application Plane. The application plane customizes different applications according to needs in various scenarios. In this article, the application plane uses the network load-balancing optimization algorithm based on LSTM to solve the problem of the transmission delay between the controller and the switch in obtaining network link status. Therefore, this plane consists of the following modules:

(i) The storage module: this module stores the network link information as the basis of the LSTM training module.

(ii) The LSTM training module: it uses the historical network link information as training data to train the LSTM model in this module. The trained model is used as the model basis of the load-balancing optimization algorithm module.

(iii) The load-balancing optimization algorithm module: this module takes the received residual bandwidth as the input of the trained LSTM model to obtain the predicted residual bandwidth. It uses the predicted value as the Dijkstra weight to calculate the optimal path between network hosts and then sends the updated flow entries to the data plane.

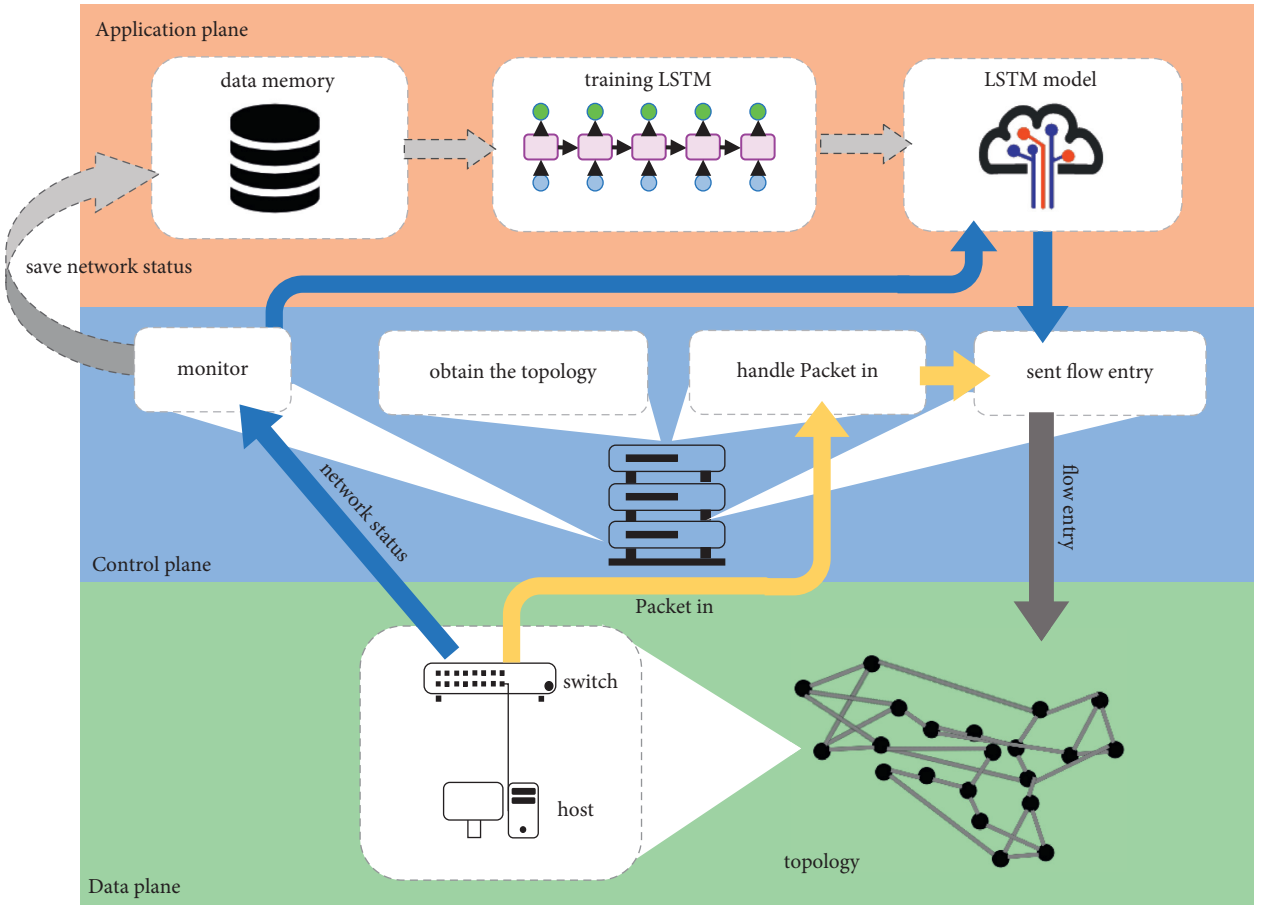


Figure 1: ALBLP network architecture. The architecture consists of the data plane, control plane, and application plane.

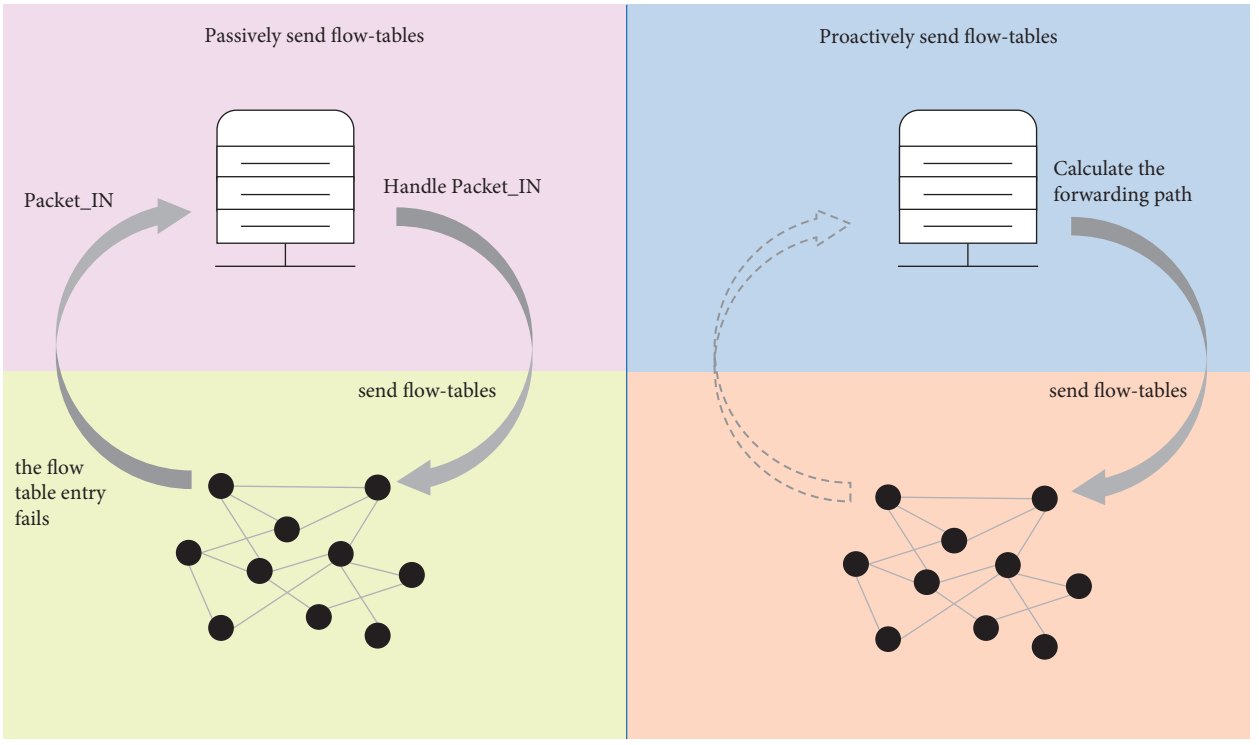


Figure 2: The difference between active and passive send-flow tables.

3.2. Load-Balancing Optimization Model. In SDN, equation (1) represents the graph structure of the data plane. We use N to denote the number of forwarding nodes; V is the set of all forwarding nodes v, that is, $V = [v_1, v_2, \ldots, v_N]$; E is the set of all links e. A link e can be defined by connecting two forwarding nodes, that is, $\forall e_{v_i v_j} \in E$, such that $v_i, v_j \in V$ and $v_i \neq v_j$. W represents the weight matrix of the link forwarding, that is, $W = [w_{ij}]_{N\times N}$, where w_{ij} represents the weight of the link between the i and j nodes. If the link exists, w_{ij} is a positive real number; otherwise, w_{ij} is 0, as shown in equation (2).

$$\text{Graph} = (V, E, W), \tag{1}$$

$$w_{ij} = \begin{cases} w_{ij} \in R^+, & e_{v_i v_j} \in E, \\ 0, & \text{else.} \end{cases} \tag{2}$$

We use B to denote the link bandwidth matrix for the entire data plane, that is, $B = [b_{ij}]_{N\times N}$, where b_{ij} represents the bandwidth of the link between the i and j nodes. If the link exists, b_{ij} is a positive real number; otherwise, b_{ij} is -1, as shown in the following equation:

$$b_{ij} = \begin{cases} b_{ij} \in R^+, & e_{v_i v_j} \in E, \\ -1, & \text{else.} \end{cases} \tag{3}$$

We use U_t to describe the used bandwidth matrix in the t time period, that is, $U_t = [u_{ij}^t]_{N\times N}$, where u_{ij}^t represents the used bandwidth between the i and j nodes in the t time period. If the link exists, u_{ij}^t is a positive real number; otherwise, u_{ij}^t is -1, as shown in the following equation:

$$u_{ij}^t = \begin{cases} u_{ij}^t \in R^+, & e_{v_i v_j} \in E, \\ -1, & \text{else.} \end{cases} \tag{4}$$

We use R_t to denote the link remaining bandwidth matrix of the data plane in the t time period, that is, $R_t = [r_{ij}^t]_{N\times N}$, where r_{ij}^t represents the remaining bandwidth between the i and j nodes in the t time period. If the link exists, r_{ij}^t is the value of b_{ij} minus u_{ij}^t; otherwise, r_{ij}^t is -1, as shown in equation (5). R_t is averaged and normalized to obtain R_t' as shown in equation (6), and the link weight can be expressed by equation (7).

$$r_{ij}^t = \begin{cases} r_{ij}^t = b_{ij} - u_{ij}^t, & r_{ij}^t \in R^+, e_{v_i v_j} \in E, \\ -1, & \text{else,} \end{cases} \tag{5}$$

$$R_t' = \left[r_{ij}^{'t} = \begin{cases} \dfrac{r_{ij}^t - \text{MIN}(R_t)}{\text{MAX}(R_t) - \text{MIN}(R_t)} & e_{v_i v_j} \in E \\ -1 & \text{else} \end{cases} \right]_{N\times N}, \tag{6}$$

$$W = \left[w_{ij} = \begin{cases} r_{ij}^{'t} & e_{v_i v_j} \in E \\ 0 & \text{else} \end{cases} \right]_{N\times N}. \tag{7}$$

Evaluating the performance of a network is essential. In this paper, we pay attention to the degree of load balancing in the network. We use the ratio of the used bandwidth to the link bandwidth to describe link utilization, as shown in equation (8), where $(\rho_t)_{ij}$ represents the link utilization rate between the i and j nodes.

$$\rho_t = \left[(\rho_t)_{ij} = \frac{u_{ij}^t}{b_{ij}} \right]_{N\times N}. \tag{8}$$

We use equation (9) to evaluate the network load-balancing factor β.

$$\beta = \frac{1}{N-1} \sum_{i=1,j=1}^{N} \left[(\rho_t)_{ij} - \text{avg}(\rho_t) \right]^2 \times b_{ij}. \tag{9}$$

In SDN, the load-balancing factor describes the relative degree of network link utilization. The forwarding of the same length of a data packet may cause a difference in network throughput due to the link bottleneck in the forwarding path [29]. Therefore, we describe the network throughput as Nt-shown in equation (10), where H_i is the amount of traffic received by host i in the data plane, and L_H is the number of hosts. The network throughput is a measure of the carrying capacity of the data plane:

$$Nt = \sum_{i=1}^{L_H} H_i. \tag{10}$$

In this paper, the optimization goal of load balancing is defined as equation (11). Equation (12) represents the limiting condition for obtaining the optimization goal:

$$\min(\beta), \tag{11}$$

$$s.\text{t} \begin{cases} \beta \geq 0, \\ b_{ij} > u_{ij}^t. \end{cases} \tag{12}$$

4. Adaptive Load-Balancing Algorithm Based on Link-State Prediction

This section describes the principle of the improved LSTM training model and the adaptive load-balancing algorithm based on link-state prediction in SDN.

4.1. Improved LSTM Training Model

4.1.1. LSTM Training Target. In this paper, the specific prediction task of the LSTM model is to predict the remaining bandwidth value between the switches in several consecutive time intervals in the future based on the remaining bandwidth observation values in several historical time intervals. The remaining bandwidth between the switches in the entire network within time interval t can be expressed as R_t. The historical time intervals of input and output are represented by W_{in} and W_{out} respectively. We use $X_t = (R_{t-W_{\text{in}}}, R_{t-W_{\text{in}}+1}, \ldots, R_t)$ to represent the historical interval observation value, and we use $Y_{t+1} = (R_{t+1}, R_{t+2}, \ldots, R_{t+W_{\text{out}}+1})$ to define the prediction interval value. The target of the model training is the mapping denoted by $f(\cdot)$ in

$$X \xrightarrow{f} (\cdot)\, Y. \tag{13}$$

4.1.2. LSTM Model. The model in this paper is composed of one LSTM layer, one Dropout layer, and one Full-connection layer. We use sliding windows to set the data. The two windows are separated by one time slice, and the input is the historical observation value X_t. The loss function uses mean squared error (MSE), and the optimizer uses the Adam optimizer. We use $x^t = X_t$ to represent the LSTM input. In LSTM, the number of memory cells is set to 5, and each memory cell contains four units-Input Gate, Forget Gate, Cell, and Output Gate-for calculation.

The Input Gate is shown in equation (14). a_l^t is calculated in equation (15), where s_c^{t-1} is the cell at the previous moment. The Forget Gate is shown in equation (16), where a_ϕ^t is calculated in equation (17). The input of the cell is x_i^t, q_l^t, and q_ϕ^t, which are, respectively, the input feature, the output of the Input Gate, and the output of the Forget Gate. The cell is calculated in equation (18). The Output Gate is shown in equation (19), where a_w^t is as shown in equation (20). The final output of the memory cell is shown in equation (21).

$$q_l^t = f\left(a_l^t\right), \tag{14}$$

$$a_l^t = \sum_{i=1}^{I} \omega_{il} x_i^t + \sum_{c=1}^{C} \omega_{cl} s_c^{t-1}, \tag{15}$$

$$q_\phi^t = f\left(a_\phi^t\right), \tag{16}$$

$$a_\phi^t = \sum_{i=1}^{I} \omega_{i\phi} x_i^t + \sum_{c=1}^{C} \omega_{c\phi} s_c^{t-1}, \tag{17}$$

$$s_c^t = q_\phi^t s_c^{t-1} + q_\iota^t g\left(\sum_{i=1}^{I} \omega_{ic} x_i^t\right), \tag{18}$$

$$q_w^t = f\left(a_w^t\right), \tag{19}$$

$$a_w^t = \sum_{i=1}^{I} \omega_{iw} x_i^t + \sum_{c=1}^{C} \omega_{cw} s_c^{t-1}, \tag{20}$$

$$q_c^t = q_w^t h\left(s_c^t\right). \tag{21}$$

4.1.3. Training Algorithm. According to the LSTM training objectives and the LSTM model, we draw the LSTM training algorithm as shown in Algorithm 1.

4.2. Adaptive Load-Balancing Optimization Algorithm. We obtain the LSTM training model as shown in the previous section. In this current section, we describe the ALBLP algorithm that optimizes the load balancing in SDN based on link-state prediction.

In SDN, the controller's decision depends on the link information obtained from the data plane. If the controller cannot receive the link information of the data plane in time, it will not be able to calculate accurately and to decide how to optimize the path. We use Delay_{cs} to represent the transmission delay between the controllers and switches in obtaining network link-state information. We calculate the time from the controller sending a link status request to the controller receiving the reply to the request as the value of Delay_{cs}. We use the forwarding state of the network to describe the impact of the transmission delay between the controllers and the switches, that is, the load-balancing factor proposed in equation (9).

According to the model obtained by LSTM, the historical matrix sequence X_t in the W_{in} interval can be input to obtain the prediction matrix sequence Y_{t+1}. However, the matrix sequence Y_{t+1} cannot be directly used as the basis for making a decision, because Delay_{cs} needs to be considered in the input time interval when placing the trained model in the network. It is necessary to select which of the prediction matrix sequences to use as the basis for eliminating the influence of Delay_{cs} as much as possible. We set the prediction time interval as T_f and the length of Delay_{cs} as T_d. The input $X_{k\times T_f}$ and output $Y_{k\times T_f+1}$ of the model in the network are expressed by equations (22) and (23), where k is the number of cycles, and the prediction interval T_f is calculated in equation (24).

$$X_{k\times T_f} = \left(R_{k\times T_f - W_{\text{in}}}, \ldots, R_{k\times T_f - W_{\text{in}}+T_d}, \ldots, R_{k\times T_f}\right), \tag{22}$$

$$Y_{k\times T_f+1} = \left(R_{k\times T_f+1}, \ldots, R_{k\times T_f+T_d}, \ldots, R_{k\times T_f+W_{\text{out}}+1}\right), \tag{23}$$

$$T_f = W_{\text{out}} - \text{Delay}_{\text{cs}}. \tag{24}$$

Figure 3 shows the input and output of the model in the network. The data used for deciding in the network is the Delay_{cs} time slice shifted backwards in the prediction matrix sequence, which is the Usable values part in Figure 3. We then use each matrix sequence in the Usable values part as the weight of the Dijkstra algorithm according to the time advancement. Finally, the flow entries are issued according to the forwarding decision output by the Dijkstra algorithm. We draw the ALBLP algorithm shown in Algorithm 2. Our ALBLP is an application-layer algorithm, and, as shown in Figure 1, the trained LSTM model will also be a part of the application plane ALBLP and will be called by ALBLP. As shown in Algorithm 2, this algorithm will periodically obtain the link information from the data plane through the controller as input, standardize the input data, and then call the LSTM model to predict the link information. The predicted link information needs to subtract part of the data that has been invalid once the time delay exists, affected by the time delay. Then, it needs to destandardize the remaining data and input the Dijkstra algorithm to obtain the forwarding path of the switch of the entire data plane. Finally, the flow entry obtained according to the forwarding path is sent to the controller and then sent to the data plane switch through the controller.

Input:
Link residual bandwidth matrix sequence X_t;
The input time series length W_{in} and the output time series length W_{out};
Training batch size *batch_size and* training times *epochs*;
Output:
model parameters after training;
(01) **for** $i = 1$ to $(n - W_{\text{in}} - W_{\text{out}} + 1)$ **do**
(02) $\mathcal{X}_i = (R_{i-W_{\text{in}}}, R_{i-W_{\text{in}}+1}, \ldots, R_i)$
(03) $\mathcal{Y}_{i+1} = (R_{i+1}, R_{i+2}, \ldots, R_{i+W_{\text{out}}+1})$;
(04) *Data_train add* $(\mathcal{X}_i, \mathcal{Y}_i)$;
(05) **end for**
(06) Initialize model *net*;
(07) **repeat**
(08) Take out *Data_train* from *Data_batch*, the size is *batch_size*;
(09) $Out = net(Data_batch)$;
(10) $Loss = MSEloss(Out, \mathcal{Y}_i)$;
(11) Back propagation to update model parameters;
(12) $epochs = epochs - 1$;
(13) **until** $epochs = 0$

Algorithm 1: LSTM training algorithm.

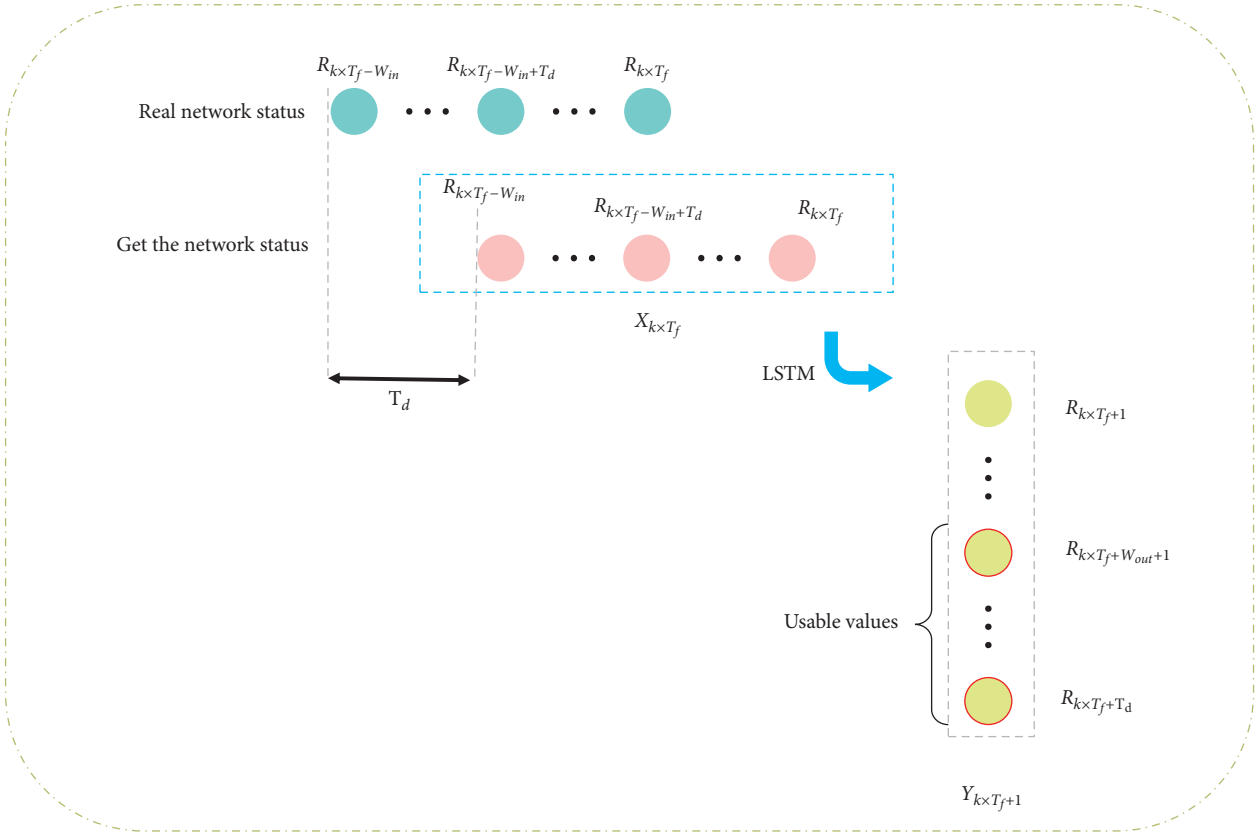


Figure 3: Input and output of the LSTM model in the experimental network. The data used for deciding on the network is the Delay_{cs} time slice shifted backwards in the prediction matrix sequence.

Input:
The network link-state value obtained periodically;
Output:
Flow entries used to control the SDN switch;
(01) Initialize the List of network link sequences $List_in = []$;
(02) Initialize the Data prediction flag $flag_dp$ = True;
(03) Obtain the network link-state value and assign it to R;
(04) Obtain the length of $Delay_{cs}$ and assign it to T_d;
(05) List_in.append (R);
(06) **if** size_of ($List_in$) <= T_f **then**
(07) $Graph$ = set_graph (R);
(08) send_flow_entries (DijKstra($Graph$));
(09) **else if** size_of ($List_in$) >= T_f and $flag_dp$ = = True **then**
(10) X = standardization ($List_in[-T_f, :]$);
(11) Y = de_standardization (**LSTM_forecast(X)**);
(12) $Y = Y\ [-\text{int}(T_f - T_d),:]$;
(13) $flag_dp$ = False;
(14) **else then**
(15) $Graph$ = set_graph ($Y[0]$);
(16) Y = delete ($Y[0]$);
(17) send_flow_entries (DijKstra($Graph$));
(18) **if** size (Y) = = 0 **then**
(19) $flag_dp$ = True;
(20) **end if**
(21) **end if**

Algorithm 2: ALBLP algorithm.

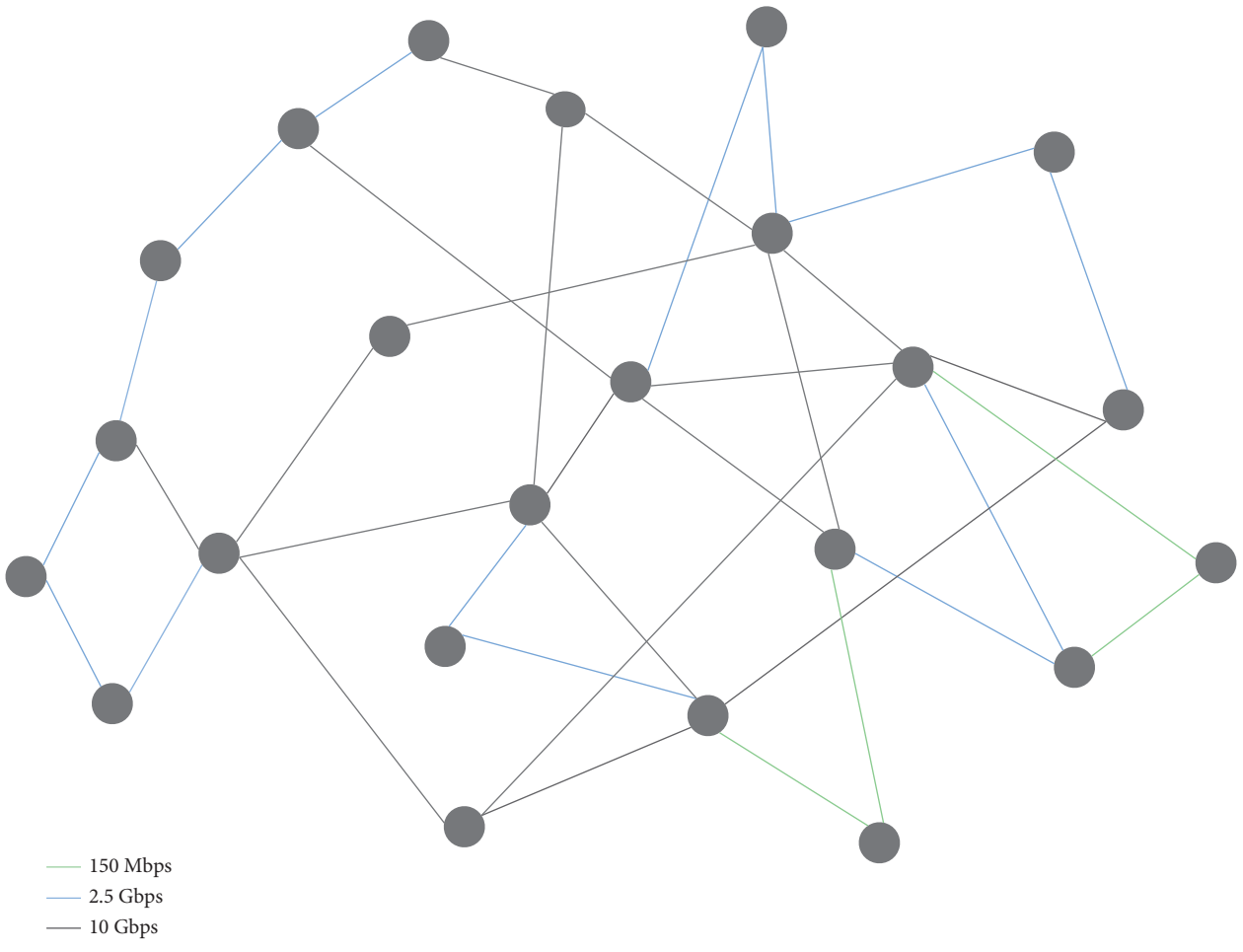


Figure 4: GÉANT network topology. The GÉANT network has 23 switches and 37 links, including 3 different bandwidth links: 10 Gbps, 2.5 Gbps, and 155 Mbps.

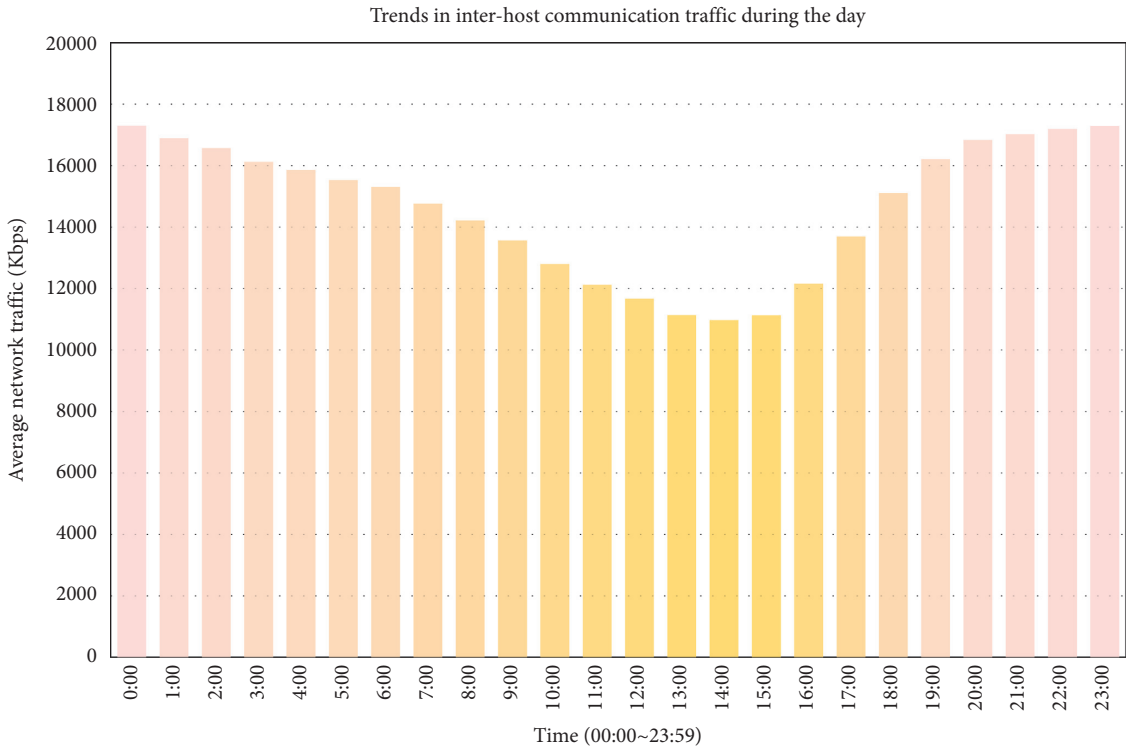


Figure 5: Trends in interhost communication traffic during the day.

5. Experiments and Analysis

For the experiment, we used the *Ubuntu* 20.4 operating system, *Mininet* network simulation, *Open vSwitch*, *Ryu* controller [30], and the OpenFlow 1.3 protocol. The computer used in the simulation experiment was equipped with an AMD Ryzen 7 2700 CPU with a base frequency of 3.2 GHz, an NVIDIA GTX1080 graphics card, and 16 GB of RAM memory.

5.1. Experimental Setup. In our experiment, we used the highly recognized GÉANT network [31], as shown in Figure 4. The GÉANT network is a data network for the European research and education community. In it, there are 23 switches and 37 links, including 3 different bandwidth links: 10 Gbps, 2.5 Gbps, and 155 Mbps. In this article, due to the performance limitations of the simulation environment, our experiment limits the link bandwidth to three bandwidths of 100 Mbps, 25 Mbps, and 1.55 Mbps. Each switch is connected to a host for receiving and sending data.

5.2. Simulation Traffic Matrix. In our experiment, we used the *Iperf* tool to send traffic. Communication traffic between hosts used the public intradomain traffic matrix [32]. The traffic matrix data was the four-month flow data in 2006, and the collection interval was 15 minutes. The average, one-day traffic packet trend is shown in Figure 5. It can be seen from Figure 5 that the peak flow is from 19 : 00 to 1 : 00 the next day in the traffic matrix. We wrote a script to send packets between hosts based on the experimental traffic matrix data.

Because the experimental link bandwidth is reduced to 1/100 of the original link bandwidth, the flow matrix between hosts is also reduced to 1/100 of the original flow matrix. We assume that the traffic matrix between the hosts is X. In order to verify the effectiveness of the algorithm under different traffic intensities, the traffic intensity set in this paper is n times of X, where n is the traffic intensity coefficient.

5.3. Experimental Results

5.3.1. LSTM Prediction Result. Our experiment used the SDN controller to collect the remaining bandwidth value of links between switches as a training data set. The experimental environment and traffic matrix are described in 5.1 and 5.2. The forwarding decision during this period is the Open Shortest Path First (OSPF) [4] algorithm. Each traffic matrix continued to send packets for 30 seconds, and the flow monitoring module collected a value every 1 second. A total of 21738240 pieces of the remaining bandwidth value were collected, of which 70% were used as a training set, 20% as a verification set, and 10% as a test set.

In our experiment, we set $W_{\text{in}} = W_{\text{out}}$, obtained from 5 to 30 link acquisition cycles in steps of 5 link acquisition cycles. The evaluation indexes are mean absolute error (MAE), root mean square error (RMSE), and symmetric mean absolute percentage error (SMAPE). The results are shown in Table 1 and Figure 6. According to the training results, we set $W_{\text{in}} = W_{\text{out}} = 20$. Other hyperparameters such as DropOut, Batch size, number of LSTM cells, and Hidden size of LSTM are shown in Table 2.

Table 1: Evaluation index of the forecast effect.

Evaluation index		MAE	RMSE	sMAPE
LSTM	$W_{in} = W_{out} = 5$	2.44	5.50	11.46
	$W_{in} = W_{out} = 10$	2.22	4.70	11.13
	$W_{in} = W_{out} = 15$	1.57	3.20	10.37
	$W_{in} = W_{out} = 20$	1.35	2.68	9.62
	$W_{in} = W_{out} = 25$	1.37	2.57	10.43
	$W_{in} = W_{out} = 30$	1.56	3.02	10.66

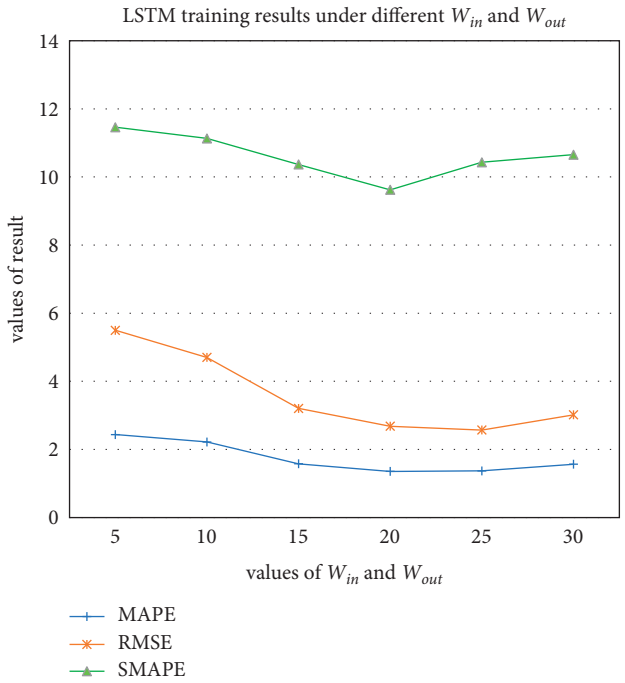


Figure 6: LSTM training results under different W_{in} and W_{out}.

Table 2: LSTM training hyperparameters.

Hyperparameters	Value
Epochs	2000
Hidden size	256
Cell	5
Batch size	128
Learning rate	0.001
DropOut	0.8
W_{in} & W_{out}	20

According to the above hyperparameters, we used the improved LSTM model to predict the remaining bandwidth of the network, as shown in Figure 7. It can be seen from Figure 7 that the predicted remaining bandwidth value is close to the actual remaining bandwidth value. The experiment results show that the improved LSTM model has a better effect in predicting the remaining bandwidth of the network link.

In the experiment, the loss of LSTM training decreases as shown in Figure 8. It essentially converges after 1000 training iterations (loss < 0.01).

5.3.2. Influence of Different Delay$_{cs}$ on the Algorithm. In order to verify the effectiveness of the Dijkstra algorithm under different Delay$_{cs}$, we experimentally observed the network load balancing of the OSPF algorithm under different Delay$_{cs}$. The unit of Delay$_{cs}$ is the link acquisition cycle. We set the link acquisition cycle as one second in the experiment. The experimental flow matrix adopted a random flow matrix in the peak flow of a day and conducted multiple experiments under the condition of the flow intensity coefficient of 1–10. Figure 9 shows a comparison of the network load-balancing factor β proposed in equation

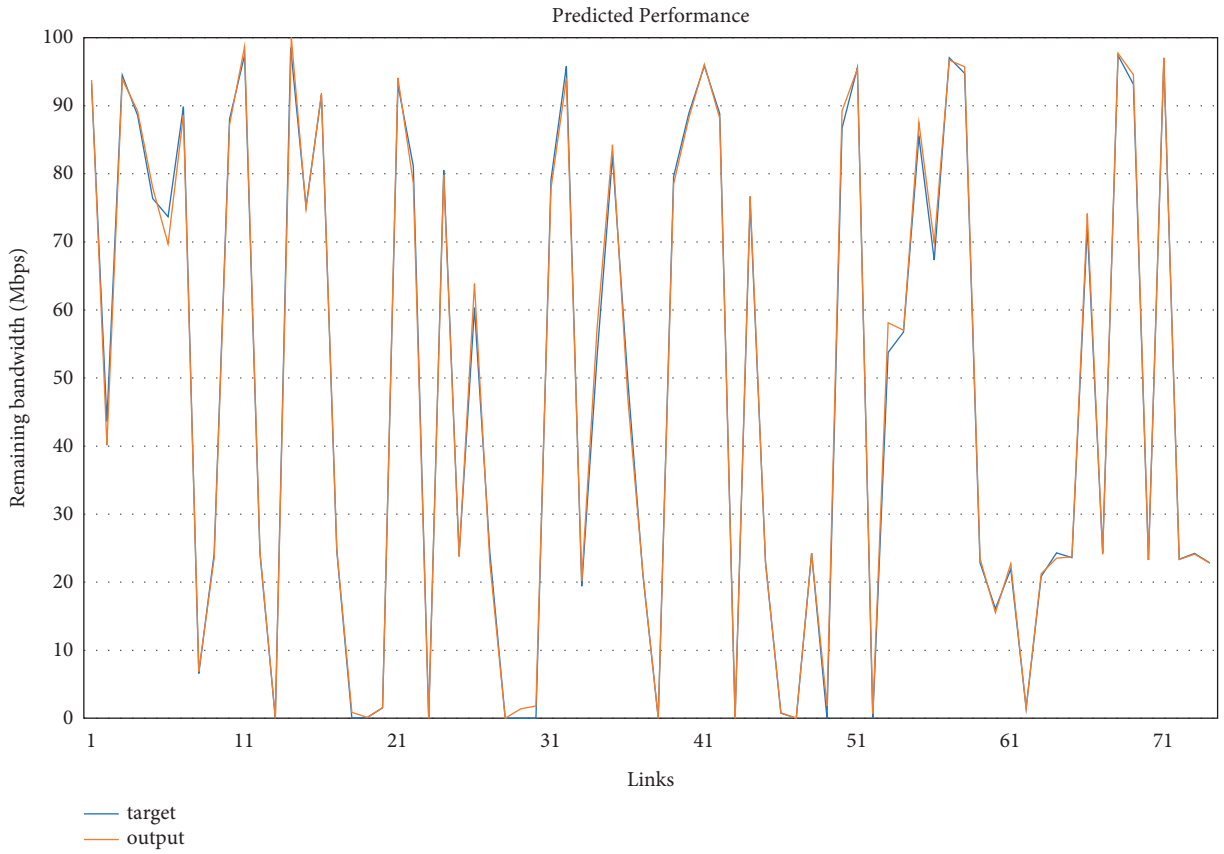


Figure 7: Display of the prediction effect of a time slice.

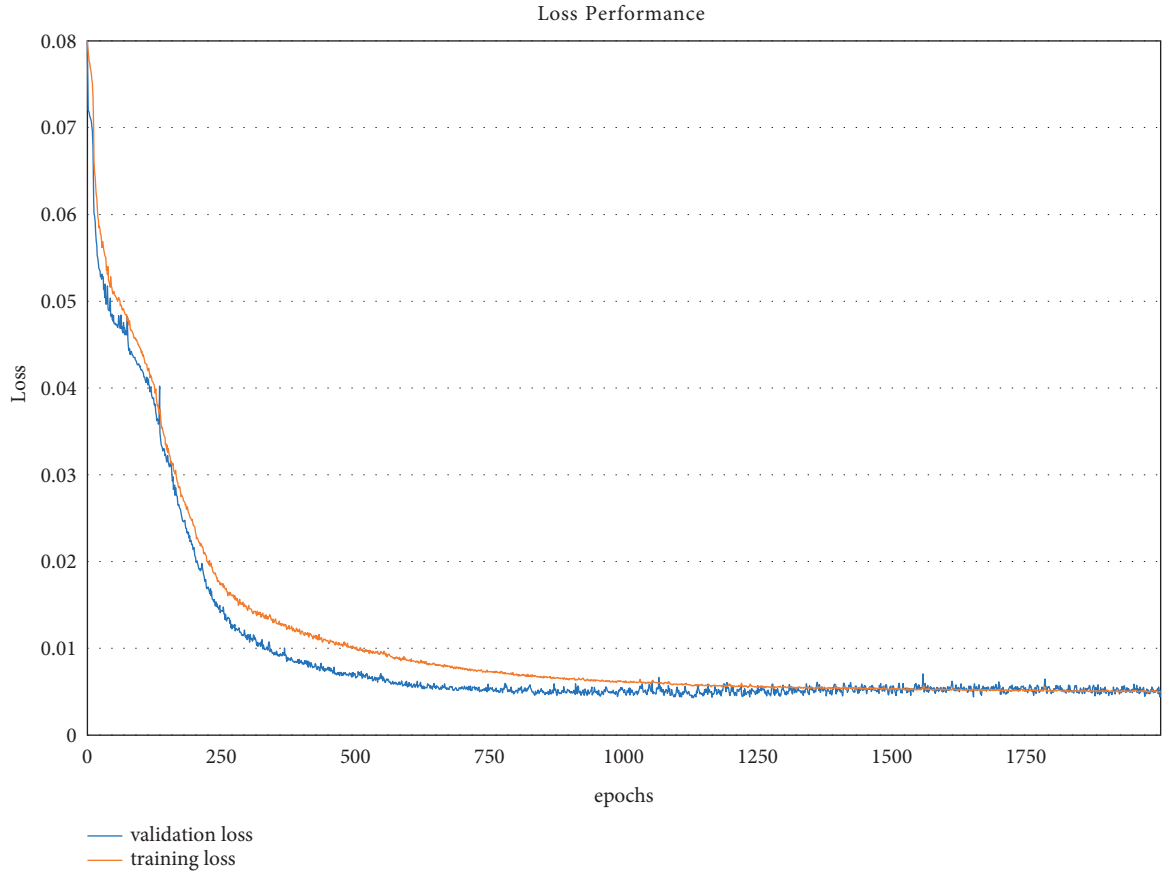


Figure 8: The loss of training and verification decreased.

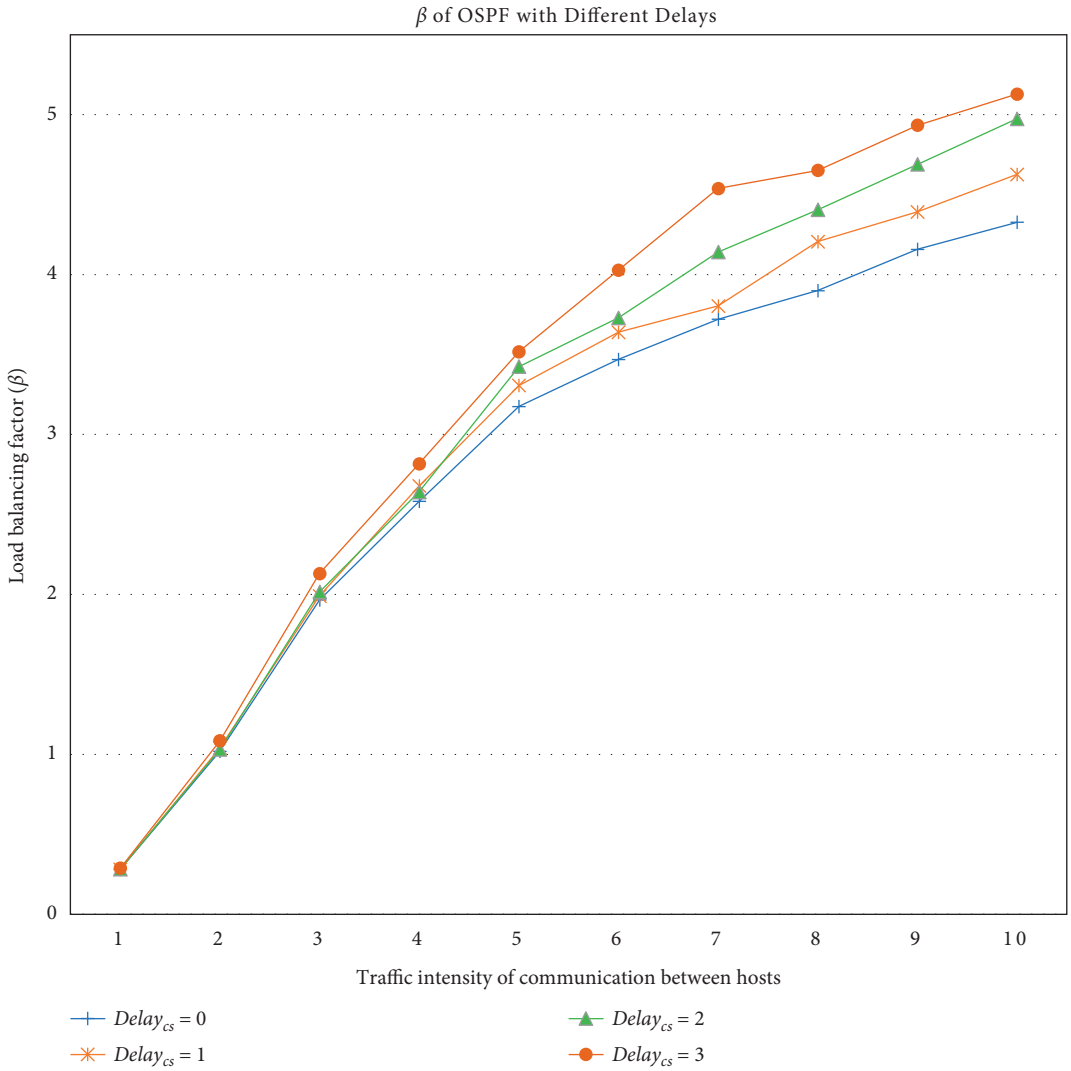


Figure 9: Load-balancing factor of OSPF with different Delay_{cs}.

(9) under different Delay_{cs}. According to the optimization goal of the load balancing in equation (11), the lower the β, the better. When the traffic intensity of communication between hosts is 1, β under each Delay_{cs} is similar. However, the higher the Delay_{cs} is, the more unbalanced the network load is with the increase in traffic intensity. Therefore, the impact of Delay_{cs} on network performance cannot be ignored.

5.3.3. Comparison of the ALBLP Algorithm, RSIR Algorithm, and OSPF Algorithm. In order to verify the effectiveness of the proposed ALBLP algorithm in solving the problem of Delay_{cs} in network performance, our experiment was designed to simulate different Delay_{cs} and conduct multiple experiments with a traffic intensity coefficient of 1–10 to compare the effectiveness of the ALBLP, RSIR [24], and OSPF [4] algorithms. The experimental results are shown in Figures 10 and 11.

Figure 10 shows a comparison of the network throughput Nt proposed in equation (10) of the three algorithms under the same traffic matrix input, where the higher the Nt, the better. The network throughput of the three algorithms is similar under low load (a traffic intensity coefficient $n < 5$). But in a high-load scenario, the network throughput of the ALBLP algorithm is greater than that of the OSPF algorithm and the RSIR algorithm. This is because, under low-load conditions, the performance of the network has been guaranteed. If the network performance is redundant, the network throughput will not fluctuate substantially. In the case of high load, if the network load-balancing problem cannot be effectively solved, part of the network will suffer from congestion, and network throughput will decrease. However, the experimental network traffic is already close to the network limit when the traffic intensity is 10. In Figure 10(a), when Delay is 0, ALBLP is not significantly improved compared to the other two algorithms, while in (b), (c), and (d), ALBLP has a higher Nt than the other two algorithms.

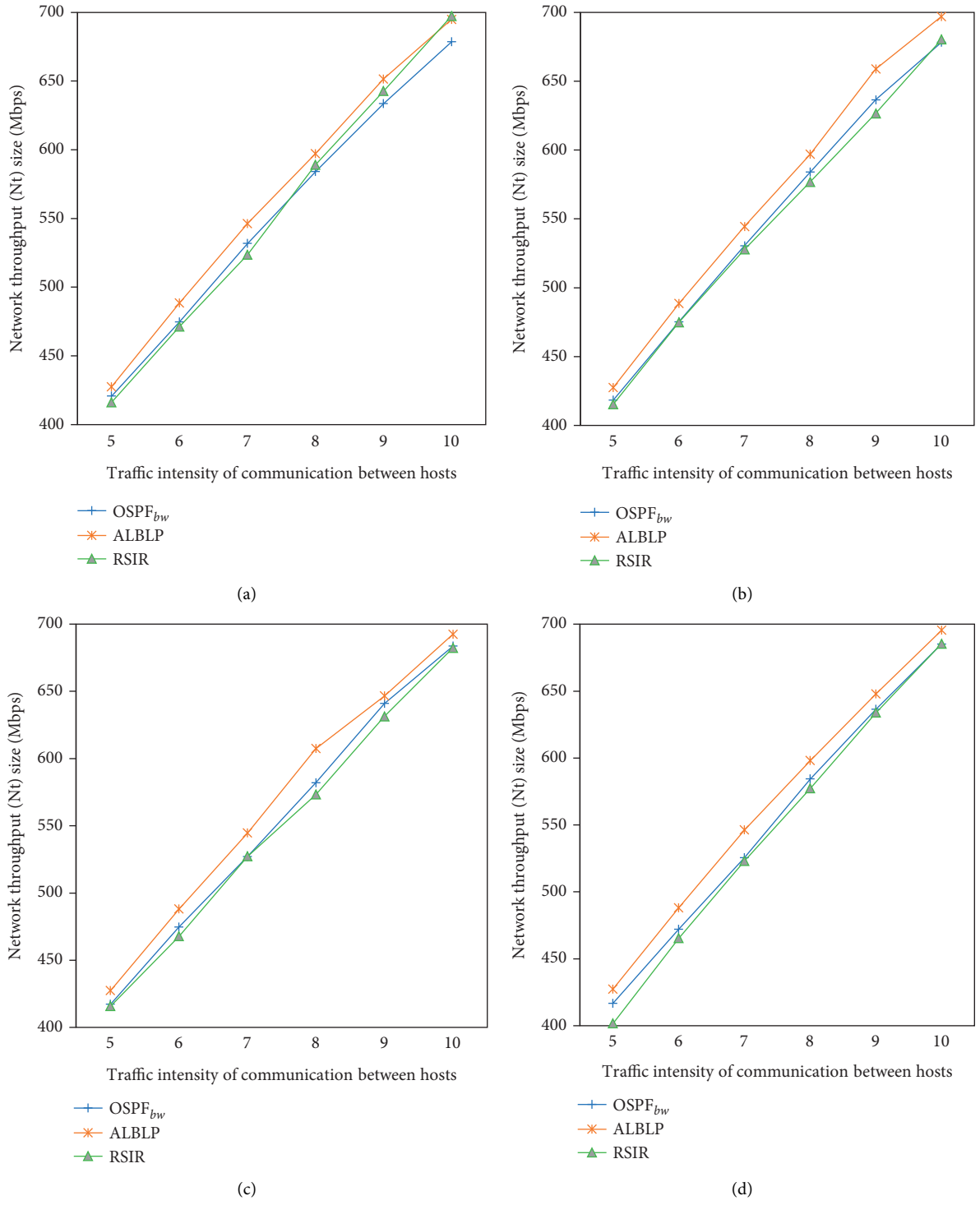


Figure 10: Network throughput of ALBLP, OSPF, and RSIR under different traffic intensities and different Delay$_{cs}$. (a) Delay$_{cs}$ = 0. (b) Delay$_{cs}$ = 1. (c) Delay$_{cs}$ = 2. (d) Delay$_{cs}$ = 3.

Figure 11 shows a comparison of the load-balancing factor β of the three algorithms under the same traffic matrix input, where the lower the β, the better. When Delay$_{cs}$ is 0, the effectiveness of the three algorithms for load balancing is similar. When Delay$_{cs}$ equals 1, 2, and 3 link acquisition cycles, respectively, the ALBLP algorithm is better than the OSPF and RSIR algorithms in load balancing under high load, and the degree of optimization increases with the increase of Delay$_{cs}$. Under low-load conditions, the network link has less occupancy rate, and the data plane has greater network redundancy. Therefore, the network load-balancing factor β does not fluctuate greatly. Under high load, the impact of time delay is greater. This is because the network is continuously transmitting a large amount of data. The

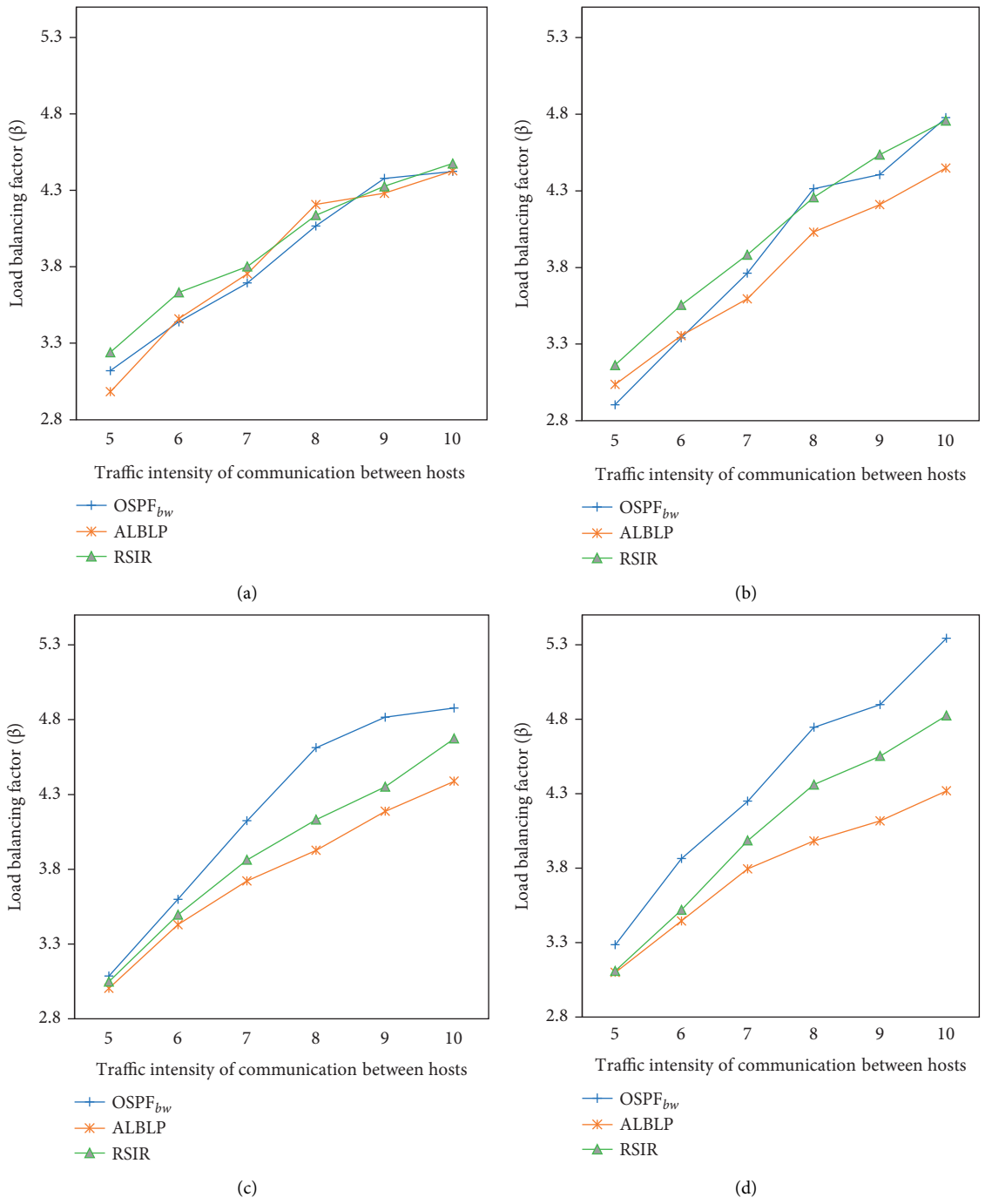


Figure 11: Load-balancing degree of ALBLP, OSPF, and RSIR under different traffic intensities and different Delay$_{cs}$. (a) Delay$_{cs}$ = 0. (b) Delay$_{cs}$ = 1. (c) Delay$_{cs}$ = 2. (d) Delay$_{cs}$ = 3.

greater the Delay$_{cs}$, the more the link information obtained by the controller lags, and the more difficult it is to make better routing decisions. If the impact of Delay$_{cs}$ cannot be solved as with the ALBLP algorithm, the network load-balancing factor β will be larger. The ALBLP algorithm uses predictive methods to ascertain the network link conditions in advance, and so the load-balancing factor β is better than that with the other two algorithms. In this experiment, the RSIR algorithm needs to retrain the Q-table and calculate the path between each host pair one by one after receiving the network link-state value. This process requires a great deal of training time, and so the network cannot make path changes in time, according to changes of network traffic, resulting in poor performance. Compared with the RSIR algorithm and the OSPF algorithm, the ALBLP algorithm has a maximum load-balancing improvement of 11.7% and 23.7%, respectively. It is worth noting that when the Delay$_{cs}$ is considerable, the RSIR and OSPF algorithms repeatedly caused

network crashes due to too much data being sent by the switch to the controller. This situation does not, however, occur in the ALBLP algorithm.

6. Conclusions and Future Work

This paper verified the influence of transmission delay between controllers and switches on network load balancing by conducting simulation experiments by setting different $Delay_{cs}$ and different network strengths. Thus, we built a network architecture based on the SDN application plane to solve the problem. Based on this construct, we designed the ALBLP algorithm to optimize network balance. The essence of the ALBLP algorithm is to use the LSTM model traffic-prediction method to offset the effect of time delay and better meet the requirements of real-time global load balancing. The experiment results show an 11.7% improvement over the RSIR algorithm in network load balancing, and 23.7% over OSPF. By comparing RSIR and OSPF, ALBLP can effectively solve the problem of transmission delay between controllers and switches under high-load network conditions. In subsequent work, we will continue to explore how to improve the efficiency of link-state prediction algorithms and how to optimize load-balancing strategies in large-scale networks.

Data Availability

The data supporting the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Natural Science Foundation of China (No. 61861013), the major program of Guangxi Natural Science Foundation (No. 2020GXNSFDA238001), the Natural Science Foundation of Guangxi (No. 2018GXNSFAA281318), and the Guangxi Project to Improve the Scientific Research Basic Ability of Middle Aged and Young Teachers (No. 2020KY05033).

References

[1] Y. Yang, G. Lu, H. Zhao, and P. Li, "Overview of the application of deep learning in software defined network research," *Journal of Software*, vol. 31, no. 7, pp. 2184–2204, 2020.

[2] J. Yan, H. Zhang, Q. Shuai, B. Liu, and X. Guo, "HiQoS: an SDN-based multipath QoS solution," *China Communications*, vol. 12, no. 5, pp. 123–133, 2015.

[3] X. Kong, S. Shen, and L. Li, "A QoS routing scheme based on software-defined networking," *Journal of Computer Research and Development*, vol. 28, no. 2, pp. 102–106, 2018.

[4] Y. Bi, G. Han, C. Lin, Y. Peng, H. Pu, and Y. Jia, "Intelligent quality of service aware traffic forwarding for software-defined networking/open shortest path first hybrid industrial internet," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 2, pp. 1395–1405, 2020.

[5] D. Kreutz, F. M. V. Ramos, P. Esteves Verissimo, C. Azodolmolky, and S. Uhlig, "Software-defined networking: a comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, 2015.

[6] X. Tao, Z. Liu, and C. Yang, "An efficient network security situation assessment method based on AE and PMU," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 1173065, 9 pages, 2021.

[7] H. Zhong, J. Fan, J. Cui, Y. Xu, and L. Liu, "Assessing profit of Prediction for SDN controllers load balancing," *Computer Networks*, vol. 191, pp. 1–10, 2021.

[8] M. Yan, H. Yuan, Z. Li, Q. Lin, and J. Li, "Energy savings of wireless communication networks based on mobile user environmental prediction," *Journal of Environmental Protection and Ecology*, vol. 22, no. 1, pp. 206–217, 2021.

[9] M. Yan, S. Li, C. A. Chan, Y. Shen, and Y. Yu, "Mobility prediction using a weighted Markov model based on mobile user classification," *Sensors*, vol. 21, no. 5, p. 1740, 2021.

[10] B. Yang, S. Sun, J. Li, and T. Yan, "Traffic flow prediction using LSTM with feature enhancement," *Neurocomputing*, vol. 332, pp. 320–327, 2019.

[11] M. Yan, W. Li, C. A. Chan, S. Bian, C. L. I, and A. F. Gygax, "PECS: towards personalized edge caching for future service-centric networks," *China Communications*, vol. 16, no. 8, pp. 93–106, 2019.

[12] M. Yan, X. Lou, and Y. Wang, "Channel noise optimization of polar codes decoding based on a convolutional neural network," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 1434347, 10 pages, 2021.

[13] A. Azzouni and G. Pujolle, "NeuTM: a neural network-based framework for traffic matrix prediction in SDN," in *Proceedings of the 2018 IEEE/IFIP International Conference on Network Operations and Management Symposium (NOMS 2018)*, pp. 1–5, IEEE, Taipei, Taiwan, April 2018.

[14] M. P. Novaes, L. F. Carvalho, J. Lloret, and M. L. Proenca, "Long short-term memory and fuzzy logic for anomaly detection and mitigation in software-defined network environment," *IEEE Access*, vol. 8, pp. 83765–83781, 2020.

[15] A. Filali, Z. Mlika, S. Cherkaoui, and A. Kobbane, "Preemptive SDN load balancing with machine learning for delay sensitive applications," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 12, pp. 15947–15963, 2020.

[16] D. Liang, J. P. Li, and C. Wang, "Applying LSTM to enable cache prefetching to optimize flow table update efficiency in SDN switches," in *Proceedings of the 7th International Conference on Information Technology: IoT and Smart City*, pp. 126–130, Shanghai, China, December 2019.

[17] J. Li, X. Chang, Y. Ren, Z. Zhang, and G. Wang, "An effective path load balancing mechanism based on SDN," in *Proceedings of the IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications (TrustCom)*, pp. 527–533, IEEE, Beijing, China, September 2014.

[18] C. A. Wang, B. Hu, S. Chen, D. Li, and B. Liu, "A switch migration-based decision-making scheme for balancing load in SDN," *IEEE Access*, vol. 5, pp. 4537–4544, 2017.

[19] H. Li, G. Yang, H. Lu, X. Fu, and Z. Shen, "Research on SDN data center network elephant flow scheduling based on ant colony algorithm," *Journal of Computer Application Research*, vol. 36, no. 12, pp. 3837–3841, 2019.

[20] X. Hai, K. K. Tae, and Y. H. Yong, "Dynamic load balancing of software-defined networking based on genetic-ant colony optimization," *Sensors (Basel, Switzerland)*, vol. 19, no. 2, pp. 1–17, 2019.

[21] M. C. Nkosi, A. A. Lysko, and S. Dlamini, "Multi-path load balancing for SDN data plane," in *Proceedings of the 2018 International Conference on Intelligent and Innovative Computing Applications (ICONIC)*, pp. 1–6, IEEE, Mon Tresor, Mauritius, December 2018.

[22] B. Mao, F. Tang, Z. M. Fadlullah et al., "A novel non-supervised deep learning based network traffic control method for software defined wireless networks," *IEEE Wireless Communications*, vol. 25, no. 4, pp. 74–81, 2018.

[23] B. Mao, F. Tang, Z. M. Fadlullah, and N. Kato, "An intelligent route computation approach based on real-time deep learning strategy for software defined communication systems," *IEEE Transactions on Emerging Topics in Computing*, vol. 9, no. 3, pp. 1554–1565, 2021.

[24] D. M. Casas-Velasco, O. M. C. Rendon, and N. L. S. da Fonseca, "Intelligent routing based on reinforcement learning for software-defined networking," *IEEE Transactions on Network and Service Management*, vol. 18, no. 1, pp. 870–881, 2021.

[25] W. Liu, J. Cai, Q. Chen, and Y. Wang, "DRL-R: deep reinforcement learning approach for intelligent routing in software-defined data-center networks," *Journal of Network and Computer Applications*, vol. 177, Article ID 102865, 2021.

[26] M. Manel, A. Kamel, and Y. Habib, "DQR: an efficient deep Q-based routing approach in multi-controller software defined WAN (SD-WAN)," *Journal of Interconnection Networks*, vol. 20, no. 4, pp. 1–26, 2020.

[27] Z. Yao, Y. Wang, L. Meng, X. Qiu, and P. Yu, "DDPG-based energy-efficient flow scheduling algorithm in software-defined data centers," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6629852, 10 pages, 2021.

[28] B. He, J. Wang, H. Sun, and J. Liao, "RTHop: real-time hop-by-hop mobile network routing by decentralized learning with semantic attention," *IEEE Transactions on Mobile Computing*, p. 17, 2021.

[29] S. Ejaz, Z. Iqbal, P. Azmat Shah, B. H. Bukhari, A. Ali, and F. Aadil, "Traffic load balancing using software defined networking (SDN) controller as virtualized network function," *IEEE Access*, vol. 7, pp. 46646–46658, 2019.

[30] J. H. Cox, S. Donovan, R. J. Clarky, and H. L. Owen, "Ryuretic: a modular framework for RYU," in *Proceedings of the 2016 IEEE Military Communications Conference (MILCOM 2016)*, pp. 1065–1070, IEEE, Baltimore, MD, USA, November 2016.

[31] P. T. Kirstein, "European international academic networking: a 20 year perspective," in *Proceedings of the Terena Conference*, pp. 1–18, Rhodes, Greece, January 2004.

[32] S. Uhlig, B. Quoitin, J. Lepropre, and S. Balon, "Providing public intradomain traffic matrices to the research community," *ACM SIGCOMM Computer Communication Review*, vol. 36, no. 1, pp. 83–86, 2006.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 4351951, 9 pages
https://doi.org/10.1155/2022/4351951

Research Article

Relative Pose Estimation for RGB-D Human Input Scans via Implicit Function Reconstruction

Pengpeng Liu,[1,2] Tao Yu,[3] Zhi Zeng,[2] Yebin Liu,[3] Guixuan Zhang,[2] and Zhen Song[4]

[1]*School of Artificial Intelligence, University of Chinese Academy of Sciences (UCAS), Beijing, China*
[2]*Institute of Automation, Chinese Academy of Sciences (CASIA), Beijing, China*
[3]*Department of Automation and BNRist, Tsinghua University, Beijing, China*
[4]*Advanced Research Center for Digitalization of Traditional Drama of the Central Academy of Drama, Beijing, China*

Correspondence should be addressed to Zhen Song; songzhen@zhongxi.cn

Received 15 October 2021; Revised 16 November 2021; Accepted 9 December 2021; Published 11 February 2022

Academic Editor: Ming Yan

Copyright © 2022 Pengpeng Liu 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 achieve a promising performance on relative pose estimation for RGB-D scans, a considerable overlap between two RGB-D inputs is often required for most existing methods. However, in many practical applications for human scans, we often have to estimate the relative poses under arbitrary overlaps, which is challenging for existing methods. To deal with this problem, this paper presents a novel end-to-end and coarse-to-fine optimization method. Our method is self-supervision which firstly combines implicit function reconstruction with differentiable render for RGB-D human input scans at arbitrary overlaps in relative pose estimation. The insight is to take advantage of the underlying human geometry prior as much as possible. First of all, for stable coarse poses, we utilize the implicit function reconstruction to dig out abundant hidden cues from unseen regions in the initialization module. To further refine the poses, the differentiable render is leveraged to establish a self-supervision mechanism in the optimization module, which is independent of standard pipelines for feature extracting and accurate correspondence matching. More importantly, our proposed method is flexible to be extended to multiview input scans. The results and evaluations demonstrate that our optimization module is robust for real-world noisy inputs, and our approach outperforms considerably than standard pipelines in non-overlapping setups.

1. Introduction

Relative pose estimation between two RGB-D scans is a fundamental problem in three-dimension (3D) vision and computer graphics. For previous multiview human motion capture [1] and human performance capture [2, 3] systems, camera extrinsic parameters (or camera poses) are required to be as accurate as possible to generate plausible results, which limit their usage. An automatic but robust autocalibration method for multiview human performance capture [4] is urgent. Examples include 3D human reconstruction from sparse views without pose parameters [5], self-calibration [6] for systems when there are disturbances to camera setups [7], and enhancing the reconstruction performance due to the inaccurate camera calibration [8].

Although there exist a lot of works focusing on relative pose estimation [9–11], they either only focus on static scene reconstruction or still require enough overlay between neighbouring RGB-D observations, which cannot be used in multiview human capture systems directly. In this paper, we are interested in relative pose estimation for human input scans, especially with nonoverlapping input scans. We present a novel end-to-end method, totally discarding the three-step paradigm [10] in the optimization module, handling the RGB-D human input scans at an arbitrary level of overlap.

Inspired from the intuition that human can estimate accurately the relative pose of two input scans, even non-overlapping, according to the underlying geometry prior. We hypothesize that the insight is to use the human prior as much as possible, especially the geometry knowledge. Therefore, this paper utilizes the human reconstruction model to enrich the underlying geometry cues of invisible regions. Taken performance, efficiency, and generalization capacity into consideration, we adopt the state-of-the-art

human reconstruction model RGB-D pixel-aligned implicit function (PIFu) [2] to obtain the detail-preserving human geometry. More importantly, we observe that multiview 3D human reconstruction is tightly related to external pose parameters, more accurate pose contributes to better reconstruction, and vice versa. It implies that we can implicitly optimize the pose parameters by supervising the human reconstruction process, free of feature extracting and matching like standard pipelines. To dig out multilevel semantic cues of human geometry prior, differentiable render [12] is used to establish a self-supervision mechanism. Thus, in the loop, we iteratively alternate between human reconstruction and relative pose estimation.

We observe that the above self-supervision optimization module is sensitive to initial result; thus, a good start is essential for faster convergence and better performance. To make our method more robust, the initialization module is added to serve the following optimization module for a stable coarse estimation. Taking inspiration from scan completion [11], firstly, we use 3D human reconstruction to complete the unseen regions and then sample point clouds with abundant geometry prior from human completion; finally, the off-the-shelf standard methods [9, 10] are used to get the coarse estimation. Based on the stable initialization, our optimization module can be more robust.

Furthermore, our coarse-to-fine framework can be easily extended to the scope of multiview inputs, owing to our special optimization mechanism via multiview implicit function reconstruction model [2]. For multiview inputs, our proposed framework can firstly optimize pairwise via the above 2-view relative pose estimation model, and then we can use multiview implicit function model and differentiable render to establish more semantic constraints, and based on the previous coarse results, we can align hierarchically the whole views to refine the pose parameters iteratively, while the most existing standard methods [9–11] can only optimize pairwise, which are error-prone for multiview extension [13].

Our contributions can be summarized as follows:

(i) The first end-to-end self-supervision relative pose estimation method for RGB-D human input scans at arbitrary overlaps, which combines the implicit function reconstruction model and differentiable render

(ii) Our proposed optimization module is robust, free of feature extraction, and correspondence matching and able to recovery good camera poses even using real-world noisy inputs. Our method also outperforms state-of-the-art standard optimization methods considerably, especially in small overlapping settings

(iii) Our method is flexible and can be easily extended to multiview input scans

2. Related Work

2.1. Relative Pose Estimation via Traditional Optimization. Traditional relative pose estimation approaches generally fall into two categories—global optimization and local optimization. Global optimization methods [7, 14] usually follow a three-step procedure [10]: extracting features, establishing correspondences of the features, and fitting a rigid transform to a subset of consistent feature correspondences. Local optimization methods [15] often focus on local pose refinement, relying on a good initialization. A popular method is geometric alignment [16], which minimizes pointwise distances between the input pairs. However, considerable overlapping regions between input scans are all required. Our approach allows input scans at any overlap and provides a stable initial result by our initialization module.

2.2. Learning-Based Relative Pose Estimation. Early works [17, 18] usually replace the modules of standard pipelines such as feature extracting or correspondence matching with deep learning networks. These methods still require massive overlaps between the input scans. Recent works [19, 20] design an end-to-end network to directly regress the relative pose. The problem of their discontinuous rotation representation such as quaternions or Euler angles is put forward in [17]. The method in [21] introduces a six-dimension (6D) continues rotation representation to alleviate the discontinuous issues. To generate probabilistic estimates, multiple rotation regressors are introduced by [22]. However, these data-driven approaches usually require large amounts of data to have a good generalization for real data. In this paper, our proposed approach is end-to-end, self-supervised, and effective for real data.

2.3. The Extreme Pose Estimation. Recently, several works addressed the extreme relative pose estimation between two input RGB-D scans, Caspi and Irani [23] focuses on image sequences and search for consistent temporal behaviour, supposing that two cameras are rigidly attached and move jointly, Yang et al. [11, 24] perform scan completion and then match the completed scans, essentially following the three-step paradigm [10]. Our initialization module is inspired from the scan completion, but our work is much different, one for that our work is well-designed for human input scans while they mainly focus on indoor scenes, not directly applied to our human input settings. Besides, our proposed optimization module discards the standard feature extracting and matching procedures and can be easily extended to multiview input settings, fully exploring the underlying human prior from the RGB-D input.

2.4. 3D Human Mesh Reconstruction via Implicit Function Representation. Recently, human reconstruction via implicit function shape representation [25, 26] has become a trend, which turns the reconstruction process more like classification task [27–29]. These methods [25, 26, 30, 31] learn a continuous implicit function representing shape using neural networks, to determine the query points inside or outside the surface. Compared with the voxel, point, and mesh representations for geometry, the implicit function representation is free of a fixed topology and not limited by the output resolution. PIFuHD [30] can produce very compelling and detailed results but struggles for complex poses, IPNet [31] focuses on 3D reconstruction from sparse and

dense point clouds and occupancy grids to produce global structure while retaining fine-scale detail, and recent work Function4D [2] achieves state-of-the-art results for 3D human reconstruction with RGB-D input scans, producing higher quality results and in real time. In this paper, we utilize [2] to obtain the detailed human geometry prior from a few RGB-D views, considering the performance and efficiency.

2.5. Differentiable Render. Differentiable render [12, 32–34] is a novel field which empowers the gradients of 3D objects to be calculated and propagated through images, which attracts increasing attention in academia and industry. The methods of differentiable render can be grouped into four categories, according to the underlying data representation: mesh, voxels, point clouds, and neural implicit functions. OpenDR [34] performs traditional rasterization in the forward pass and computes the approximate gradients in the backward pass. Recently, PyTorch3D [12] introduces a modular renderer by redesigning and exposing intermediates computed during rasterization. In this work, we use the available library PyTorch3D [12], considering its significant speed and memory improvements and convenience.

3. Approach

Given a pair of human RGB-D scans S_1 and S_2 with the camera intrinsic parameters as the input, the goal is to output the accurate rigid transformation T_{12}. To explore the hidden cues in the challenging nonoverlapping settings, we firstly exploit RGB-D PIFu [2] to recover the underlying human geometry prior from a sparse view and then propose a coarse-to-fine strategy for a robust optimization. As illustrated in Figure 1, our architecture consists of two modules; one is the initialization module producing a stable and fair good coarse pose; the other is the optimization module, refining the pose iteratively. Besides, we can easily extend our proposed method to multiview input scans.

3.1. Human Reconstruction with Implicit Function. Implicit function f is a spatially aligned and memory-efficient 3D shape representation, which consists of multilayer perceptrons [2, 30]. Instead of outputting the 3D volume, it learns a continuous function which determines the 3D voxel in space inside or outside the surface and then uses Marching Cube [35] to infer the surface based on the output Signed Distance Function (SDF) field. Recently, RGB-D PIFu proposed by [2] introduces extra depth observation to improve the reconstruction performance and accelerate the inference with filtering some exterior voxel using depth value. The network consists of an implicit function f and a fully convolutional encoder F. Following the RGB-D PIFu proposed by [2], the human surface can be defined as a level set of:

$$\begin{aligned} &f(F(\Pi(q)), q_{.z}, T(q)) = s : s \in \mathbb{R}, \\ &T(q) = \text{trunc}(q_{.z} - D(\Pi(q))), \end{aligned} \tag{1}$$

where f is a continuous implicit function represented by multilayer perceptrons. For a query point q, $\Pi(\cdot)$ is the perspective projection function, $F(\cdot)$ is the feature extracted by encoder networks, $D(\cdot)$ is a bilinear function sampling depth values on the depth image, the Probabilistic Signed Distance Function (PSDF) value of q is introduced by $q_{.z} - D(\Pi(q))$ to fully utilize the depth observation, and $T(\cdot)$ is used to truncate PSDF value to $[-\sigma_i, \sigma_i]$, eliminating the ambiguities of using global depth values. The sign of s represents whether the point is inside or outside the surface.

The RGB-D PIFu [2] meets the requirements of good generalization as discussed in [36], using view-centric coordinate systems, extracting geometry-aware feature maps, and exploiting multiview aggregation strategy for fusion. As demonstrated in Figures 2 and 3, the model shows the good generalization in real data collected by Kinect, which is trained only using 300 high-fidelity data from [2]. We observe that the reconstruction of visible parts is generally detailed and lifelike, while the unseen part completion is a little oversmooth but still human-like according to the learning prior from large data, which still empowers us abundant hidden cues. Considering the efficiency, compelling performance, and good generalization in real data, we adopt the RGB-D PIFu [2] to dig out the underlying geometry prior, which can produce a detail-preserving complete surface in real time.

3.2. Initialization Module. A good initial result can accelerate the convergence and contribute to better performance. As in Figure 1, the black arrows represent the flow of the initialization module which serves as a coarse relative pose estimation. The challenge is how to ensure a fair good result even in nonoverlapping input settings where almost all the standard pipelines [7, 9] cannot work well due to lack of enough correspondence. Enlightened by scan completion [11], human geometry prior through implicit function reconstruction is introduced to provide the hidden cues of the invisible region. Given the RGB-D scan S_i and camera intrinsic parameters K_i, for single-view input, the world coordinate can be assumed identical to the camera coordinate; thus, the completed human mesh M_i^1 can be reconstructed with single-view RGB-D PIFu without the relative pose. And then, using point clouds sampled from the mesh M_i^1, the initial relative pose can be estimated with the off-the-shelf standard method easily. In this paper, we leverage a popular registration method which combines the global registration [14] with local registration [15, 16]. Compared to the point clouds derived from depth images, the human completion mesh M_i^1 provides detail-preserving surface in visible observation and much richer hidden cues of human geometry prior in unseen parts. In addition, our initialization process is also efficient owing to the real-time property of the RGB-D PIFu [2].

3.3. Optimization Module. This module is served as a relative pose refinement. For multiview human reconstruction, adding a new view but with erroneous pose can be suicide, which may lead to artifacts, distortion, or even collapsed results. Fortunately, we can just grasp this property and establish a self-constraint end-to-end optimization mechanism via differentiable render. The insight for the proposed optimization process is that the inferred models (including

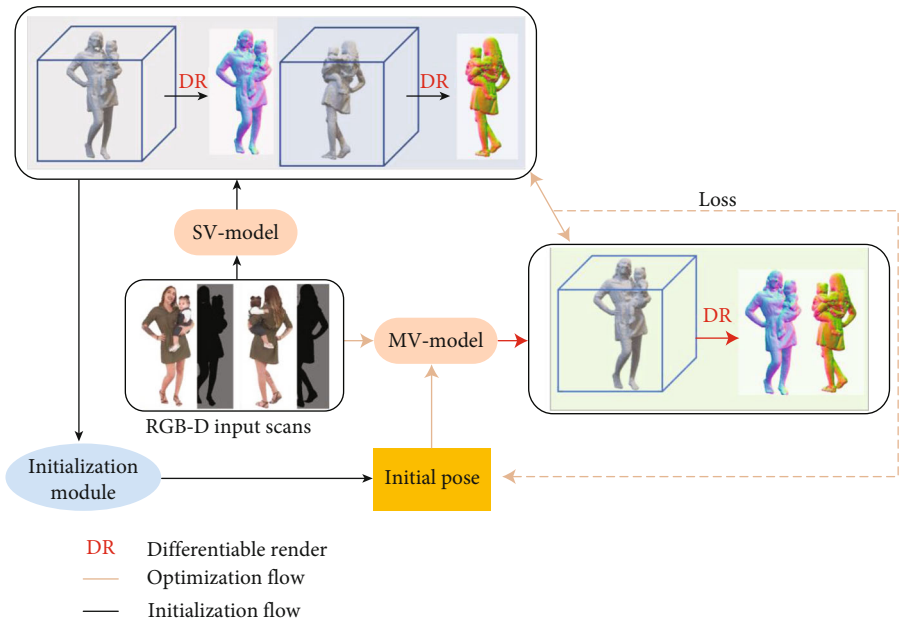


Figure 1: Methods pipeline. SV is the abbreviation of single-view, and SV model means our single-view human reconstruction model; similarly, MV means multiview and MV model means our multiview reconstruction model. The proposed framework combines the initialization module and optimization module. The black lines represent the initialization flow, and the red lines are the optimization flow. Given the RGB-D input scans, the initialization module produces a coarse initial pose; then, the optimization module refines the relative pose iteratively.

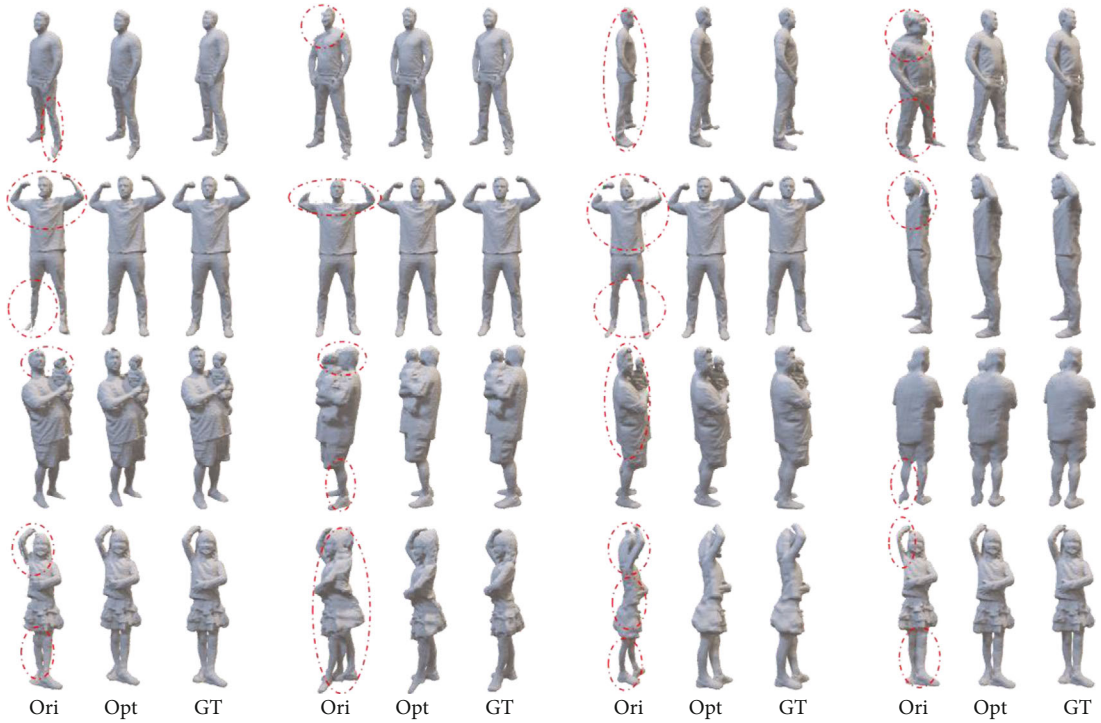


Figure 2: We qualitatively show the robustness of our optimization module. We add different small random Gauss noise to the groundtruth pose. The first column is the 2-view RGB-D PIFu reconstruction with original pose (with noise), the second column is the reconstruction after our optimization module, and the last column is the human reconstruction with groundtruth pose.

single-view inferred models from each view and also the multi-view inferred model) should be consistent with each other. If the multiview inferred model is quite different from the single-view inferred models, the relative pose must not be accurate. Moreover, the multiview inferred model should also match the real RGB-D observations from different

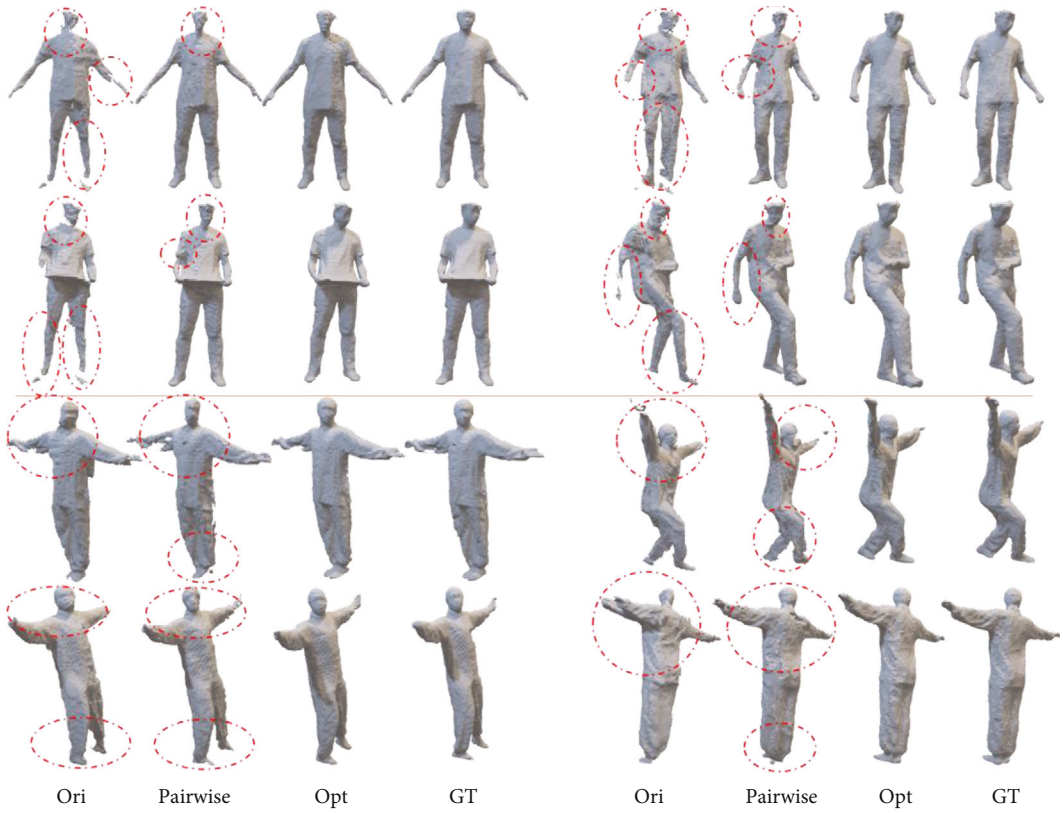


FIGURE 3: We show the performance of our multi-view extension. The first two rows show the 3-view input scans while the last two rows with 4-view input scans. We still add different small disturbances to the groundtruth poses. The first column is the reconstruction with original pose, the second column is the reconstruction with only pairwise optimization, the third column is the human reconstruction with our proposed optimization method, and the last column with the groundtruth pose.

views, which means that the multi-view inferred model should not violate the two-dimension (2D) semantic constraints from the real observations. Specifically, the differentiable render can be firstly used to render the human mesh M_i^1 from the input perspective to get the groundtruth semantic label such as mask $\hat{m}_i$ and normal image $\hat{n}_i$, for that the lifelike rendering results could be ensured from the visible perspective. Note that additional supervisions can also be added by rendering from novel views if the rendering results are enough realistic.

Given the initial relative pose parameters $T_{i0} = [R, t]$, the human mesh M_i^2 can be generated by our 2-view RGB-D PIFu. Similarly, the predicted semantic label such as mask $\hat{m}_i$ and normal image $\hat{n}_i$ can be rendered via differentiable render. Then, the loss terms can be proposed based on the following principles and be back propagated to update the relative pose parameters iteratively.

For loss terms, instead of penalizing the relative pose directly (which is also impossible), we implicitly supervise the multiview human reconstruction process. Specifically, we compute (R, t) by solving

$$\min_{R,t} a_1 L_{2D}(R, t) + a_2 L_{\text{sdf}}(R, t) + a_3 L_{\text{orth}}(R), \tag{2}$$

where a_i is the weight coefficient for each energy term. From Equation (2), our loss is composed of three parts, the 2D semantic loss $L_{2D}(\bullet)$, the SDF field loss $L_{\text{sdf}}(\bullet)$, and the orthogonal constraint $L_{\text{orth}}(\bullet)$.

The first term $L_{2D}(\bullet)$ is a 2D semantic supervision to keep the 2D semantic consistency. We mainly use mask and normal as 2D semantic priors to make sure that the rendered images (from the multiview RGBD-PIFu result) should be consistent with the input images. Thus, our 2D semantic loss can be described as $L_{2D} = b_1 l_{\text{mask}} + b_2 l_{\text{normal}}$. We empirically set the weight coefficient $b_1 = b_2 = 1$. The normal loss l_{normal} is a $L2$ norm loss, described as $\|n_i - \hat{n}_i\|_2$, enhancing the normal consistency. The mask loss penalizes the small overlap of the two mask images, described as $\|m_i^{\text{dist}} - \hat{m}_i^{\text{dist}}\|_2$, the superscript dist representing distance transformation. Instead of calculating mean square error (MSE) for original mask images directly, we firstly using distance transform and then compute the MSE, which is more robust by experience.

The SDF field loss $L_{\text{sdf}}(\bullet)$ is a 3D supervision, ensuring that the multiview reconstruction mesh is human-like. It can be described as $\|\text{sdf}^2 - \text{sdf}^1\|_2$, the superscript representing the view numbers, sdf^1 representing the predicted SDF of the single-view RGB-D PIFu. We observe that only supervised by the sparse view 2D projection is not enough, it is probably to be trapped into local minima that satisfy each perspective's constraint but looking like deformed and

weird. To solve the problem, we introduce the SDF field loss to make the consistency between the multiview SDF field and each single-view SDF field, to avoid constraining SDF filed into an abnormal field.

The last constraint $L_{\text{orth}}(\bullet)$ is an orthogonal constraint which is used to keep the orthogonality of rotation matrix R, described as $\|RR^T - I\|_2$, where I is identity matrix.

3.4. Multiview Extension. The above model is aimed at estimating the relative pose of a pair of images, while the goal of multiview extension is to output the relative pose of each view, given multi-view RGB-D input scans of the same person. However, most existing relative pose estimation approaches [11, 14, 24] are not suitable for multiview input, for that they mainly focus on pairwise input, which can only optimize pairwise for multi-view setting. As [13] has observed, only pairwise registration is error-prone, leading to odometry drift due to sensor noise and false pairwise alignments. Some multiway registrations [13] align multiple pieces of geometry in a global space via pose graph optimization [37], they combine local optimization such as point-to-plane ICP [9] with global optimization such as Levenberg-Marquardt algorithm [38], yet they still rely on large overlapping region to match considerable and accurate correspondences.

Our proposed approach can be flexibly extended to multiview setups owing to the good generalization of our multi-view implicit function human reconstruction model RGB-D PIFu. For multiview settings, the n-view optimization can be based on the results of $n-1$ views, just like the pyramid structure; thus, we can optimize hierarchically in a coarse-to-fine strategy. As shown in Figure 4, the relative pose of the three views is agnostic, assuming the view_1 as the world coordinate system; we firstly optimize pairwisely, view_{12}, view_{13}, and view_{23}, note that the initial pose of view_{23} can be referred after view_{12} and view_{13} optimizations. Then, we align globally via 3-view pose optimization module to refine the pose. The architecture of 3-view pose optimization is analogous to 2-view optimization as Figure 1; using 3-view RGB-D PIFu and differentiable render to establish self-supervision mechanism, the loss term can be described as the following equation:

$$\min_{R,t} \sum_{i=1}^{n} a_{i1} L^i_{\text{2D}}(R,t) + a_{i2} L^i_{\text{sdf}}(R,t) + \text{a}_3 L_{\text{orth}}(R). \quad (3)$$

In practice, we can skip some middle-level optimization in practice, because of the diminishing marginal utility [39]; the more top-level optimization, the less effect it improves.

4. Results and Discussion

In this section, we present an experimental evaluation of the proposed approach. We firstly describe our evaluation dataset and evaluation metrics. Then, we compare our method with several baseline techniques, assessing the performance quantitively on different overlapping RGB-D input pairs. Finally, we present an ablation study to quantitively examine the impact of our initialization module and qualitatively demonstrate the robustness of our proposed optimization module.

4.1. Dataset. We perform experimental evaluation on two types of data. One is the synthesized data, rendering 200 high-quality scans from 60 views with rotation and random shifts. Note that to keep consistent with the real data, for the colour image, we use the PRT-based render as in [25]; for the depth image, we first render the groundtruth depth maps and then add the TOF sensor noise on top of them following [2]. Finally, we synthesize RGB-D data with resolution 512×512. The other is real data collected by multiview Kinects, which includes large poses, various clothes, and different people. We firstly segment the human mask provided by Kinect. Then, we align the colour image and depth image with the pose parameters of Kinect depth camera, getting the final real RGB-D data with resolution 640×576.

For a more comprehensive and detailed analysis, we classify all the data into four categories according to the overlap rate: overlap rate 0-5%, overlap rate 10%-30%, overlap rate 40%-70%, and overlap rate over 80%. Then, we select about 400 characteristic data pairs for each category as our final evaluation data, besides each category is mixed by half the real data and half the synthetic data. Note that for the RGB-D PIFu, we train it using 500 high-quality scans following [2].

4.2. Evaluation Metrics. We evaluate the rotation matrix R and translation part t of the relative pose $T = (R, t)$, respectively. We follow the standard protocol of reporting the rotation angle error $\arccos((tr(R^*R^T) - 1)/2)$ and translation error $\|t - t^*\|_2$, let (R^*, t^*) be the groundtruth relative pose and (R, t) be the predicted pose.

4.3. Quantitative Evaluation. We consider four baseline approaches: Super4PCS [7], Greg [10], ICP [9], and combine Greg with ICP. Super4PCS is a widely used global scan matching method between two 3D point clouds. Greg is another state-of-the-art global registration, combining cutting-edge features and reweighted least squares for rigid pose registration. ICP (iterative closet point), a local optimization algorithm, has been a mainstay of geometric registration in both research and industry. In this paper, we use point-to-plane ICP which has a faster convergence. We also combine global registration with local optimization as a baseline; the former provides an initial pose, and the latter refines the pose, just as our coarse-to-fine strategy.

Table 1 provides the quantitative results of our approach and baseline methods. We show the mean error for rotation and translation components for overlapping rate ($0-5\%$, $10\%-30\%$, $40\%-70\%$, and $\geq 80\%$) scan pairs, respectively. Overall, we observe that the less overlap rate, the greater advantage of our method. Our approach outperforms baseline approaches considerably in small-overlapping and almost overlapping settings, while performing slightly better in large-overlapping settings. In the four baselines, combining Greg with ICP is much better than others, especially in the significant overlap scenes, while in small-overlapping settings, all the baselines perform badly, making no difference.

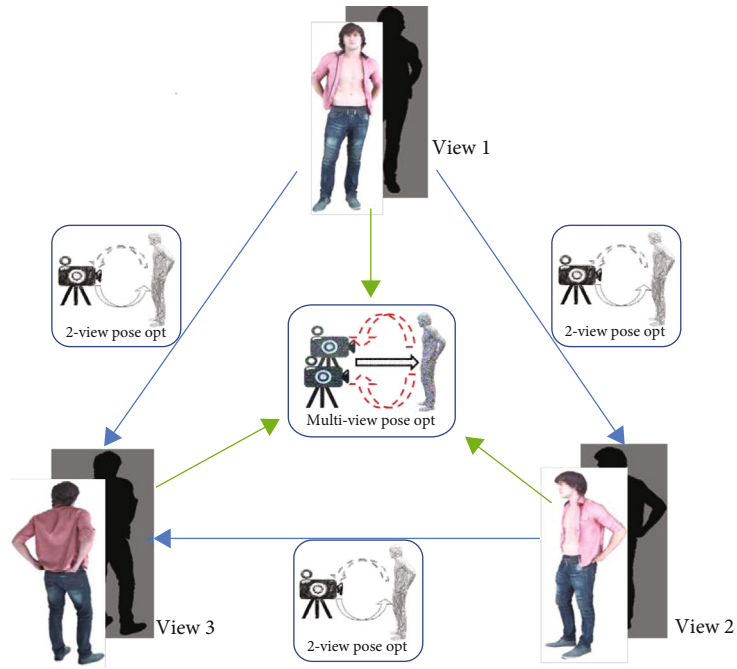


Figure 4: Multiview optimization. Opt is the abbreviation of optimization, 2-view pose Opt in picture means our 2-view relative pose optimization, whose structure is as described in Figure 1; similarly, multiview pose Opt means our multiview relative pose optimization. We optimize the pose parameters hierarchically, just as the pyramid structure. We firstly optimize pairwise to get a coarse pose and then hierarchically align the whole views based on the coarse results via our proposed multiview optimization module.

Table 1: The relative pose estimation evaluation on our dataset. We divide all the data into four categories according to the overlap rate: 0-5%, 10%-30%, 40%-70%, and over 80%, and report the mean error of rotation angle (Rot. (°)) and translation (Trans. (m)), respectively.

	Overlap (0-5%)		Overlap (10-30%)		Overlap (40-70%)		Overlap (≥80%)	
	Rot.	Trans.	Rot.	Trans.	Rot.	Trans.	Rot.	Trans.
Super4PCs [7]	126.08	1.715	103.95	1.468	38.99	0.574	4.46	0.084
Greg [10]	124.41	1.679	92.50	1.538	64.47	1.109	18.44	0.325
ICP [9]	165.97	2.252	122.02	1.943	67.86	1.208	27.75	0.540
Greg+ICP	126.72	1.664	103.88	1.461	35.10	0.508	1.27	0.0262
Ours-initial	7.15	0.138	6.54	0.130	5.90	0.106	1.29	0.0259
Ours	6.16	0.127	5.88	0.119	4 .60	0.084	1.15	0.0209

In small overlap (overlapping rate 10%-30%) or almost no overlap (overlapping rate 0-5%), all the baselines perform very badly, with over 120 rotation errors and over 1.5 m translation errors. These methods rely on accurate correspondence in the overlap regions; thus, they cannot handle the small-overlapping or nonoverlapping settings. Even so, our approach performs much better, with mean errors in rotation/translation 6.16/0.127 m for almost no overlap and 5.88/0.119 m for small overlap. It fully demonstrates the effectiveness of our proposed approach for nonoverlapping settings, thanks to the coarse-to-fine optimization strategy.

In middle overlap (overlapping rate 40%-70%), although the Greg with ICP is much better than other baselines, with rotation/translation errors 35.10/0.508 m, our approach achieves much better results, with corresponding errors 4.60/0.084 m.

In significant overlap (overlapping rate over 80%), there is no significant difference between all the methods. All the baselines perform fairly good, especially, the Greg with ICP with small rotation/translation errors 1.27/0.0262 m. It further shows that the standard baselines require input scans possessing considerable overlapping regions for superior performance. However, our approach is still competitive, performing slightly better with mean rotation/translation errors 1.15/0.0209 m.

In this experiment, all the four baselines rely highly on the large overlap of input scans, not able to handle the extreme pose setting, while our approach performs stable and good in all settings.

4.4. Ablation Study. We conduct two experiments to evaluate the effectiveness of our proposed module. Firstly, we evaluate quantitatively our initialization module. From Table 1,

we find that our initialization module performs very stably. Even in the almost nonoverlapping settings, it can generally provide fair good initial results with small mean rotation/translation errors 7.15/0.138 m. In this module, we use RGB-D PIFu to reconstruct the full geometry, digging out the abundant hidden cues of the unseen regions. Therefore, our proposed initialization module can be free of overlap of the inputs.

Secondly, we evaluate qualitatively the robustness of our optimization module. This module serves as a refinement, conditioned on a not much bad pose. Thus, during the experiment, we add an arbitrary small Gauss random noise to the groundtruth pose parameter as a coarse initial pose. For a more intuitive distinction in the results, we qualitatively demonstrate the reconstruction performance, considering that the reconstruction procedure will amplify the impact of small deviation. Figure 2 shows the performance of different small noises. We can see that our proposed optimization can generally recovery a fair perfect human mesh in different small disturbances, which fully proves the robustness of our model.

Our approach can be easily extended to multiview input settings. As demonstrated in Figure 3, our proposed methods perform better than only pairwise optimization, closely to the reference reconstruction.

5. Conclusions

This paper proposes an end-to-end approach which consists of an initialization module and an optimization module, to estimate the relative pose between the RGB-D human input scans. We do not limit the input scans to have large overlap. Our initialization module can handle nonoverlapping settings effectively by incorporating the learned 3D human shape prior. Our optimization module refines our pose parameters by implicitly supervising the human reconstruction with multilevel constraints via implicit function reconstruction and differentiable render. The experiments demonstrate our optimization module is robust to small disturbances. Besides, our method can easily extend the scope to multiview input scans. Through evaluation on different overlap data, our method considerably outperforms the state-of-the-art baselines, especially for nonoverlapping scans. We believe the proposed method will stimulate the wide spread of multiview human reconstruction systems by eliminating the sophisticated camera pose calibration process.

Data Availability

We collect about 500 human models from Twindom dataset for training RGB-D PIFu. You can purchase the Twindom dataset from the webpage: https://web.twindom.com/. We provide part of our evaluation data, you can download from the hyperlink: https://drive.google.com/file/d/1SB3eTvbcG2b6CJbpF84jm_pApNMCTyvf/view?usp=sharing.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This work was supported by "Beijing Outstanding Young Scientist Program" (BJJWZYJH01201910048035) and "the Fundamental Research Funds for the Central Universities" (YNZDA1805).

References

[1] W. Li, Y. Zeng, Q. Zhang, Y. Wu, and G. Chen, "Human motion capture based on incremental dimension reduction and projection position optimization," *Wireless Communications and Mobile Computing*, vol. 2021, 9 pages, 2021.

[2] T. Yu, Z. Zheng, K. Guo, P. Liu, Q. Dai, and Y. Liu, "Function 4d: real-time human volumetric capture from very sparse consumer rgbd sensors," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5746–5756, 2021.

[3] T. Yu, Z. Zheng, K. Guo et al., "DoubleFusion: real-time capture of human performances with inner body shapes from a single depth sensor," in *IEEE conference on computer vision and pattern recognition*, pp. 7287–7296, Salt Lake City, USA, 2018.

[4] Z. Li, T. Yu, Z. Zheng, K. Guo, and Y. Liu, "POSEFusion: pose-guided selective fusion for single-view hu-man volumetric capture," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14162–14172, 2021.

[5] X. Li, S. Liu, K. Kim et al., "Self-supervised single-view 3d reconstruction via semantic consistency," in *European Conference on Computer Vision*, Springer, 2020.

[6] O. D. Faugeras, Q. T. Luong, and S. J. Maybank, "Camera self-calibration: theory and experiments," in *European Conference on Computer Vision*, pp. 321–334, Springer, 1992.

[7] N. Mellado, D. Aiger, and N. J. Mitra, "Super 4PCS fast global pointcloud registration via smart indexing," *Computer Graphics Forum*, vol. 33, no. 5, pp. 205–215, 2014.

[8] J. Chen, K. Benzeroual, and R. S. Allison, "Calibration for high-definition camera rigs with marker chess-board," in *IEEE Computer Society Conference on Coputer Vision and Pattern Recognition Workshops*, pp. 29–36, Providence, RI, USA, 2012.

[9] S. Bouaziz, A. Tagliasacchi, and M. Pauly, "Sparse iterative closest point," *Computer Graphics Forum*, vol. 32, no. 5, pp. 113–123, 2013.

[10] G. K. Tam, Z. Cheng, Y. Lai et al., "Registration of 3d point clouds and meshes: a survey from rigid to nonrigid," *IEEE Transactions on Visualization and Computer Graphics*, vol. 19, no. 7, pp. 1199–1217, 2012.

[11] Z. Yang, J. Z. Pan, L. Luo, X. Zhou, K. Grauman, and Q. Huang, "Extreme relative pose estimation for RGB-D scans via scene completion," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4531–4540, California, USA, 2019.

[12] N. Ravi, J. Reizenstein, D. Novotny et al., "Accelerating 3d deep learning with PyTorch3D," 2020, https://arxiv.org/abs/2007.08501.

[13] S. Choi, Q. Zhou, and V. Koltun, "Robust reconstruction of indoor scenes," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5556–5565, Boston, USA, 2015.

[14] D. Aiger, N. J. Mitra, and D. Cohen-Or, "4-points congruent sets for robust pairwise surface registration," *ACM SIGGRAPH*, vol. 27, no. 3, pp. 1–10, 2008.

[15] P. J. Besl and N. D. McKay, "A method for registration of 3-d shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 2, pp. 239–256, 1992.

[16] Y. Chen and G. Medioni, "Object modelling by registration of multiple range images," *Image and Vision Computing*, vol. 10, no. 3, pp. 145–155, 1992.

[17] S. Kim, S. Lin, S. R. Jeon, D. Min, and K. Sohn, "Recurrent transformer networks for semantic correspondence," *Advances in Neural Information Processing Systems*, vol. 3, pp. 6126–6136, 2018.

[18] R. Ranftl and V. Koltun, "Deep fundamental matrix estimation," in *Computer Vision – ECCV 2018: European Conference on Computer Vision (ECCV)*, Lecture Notes in Computer Science, pp. 284–299, Springer Link, 2018.

[19] E. Insafutdinov and A. Dosovitskiy, "Unsupervised learning of shape and pose with differentiable point clouds," 2018, https://arxiv.org/abs/1810.09381.

[20] I. Melekhov, J. Ylioinas, J. Kannala, and E. Rahtu, "Relative camera pose estimation using convolutional neural networks," in *International Conference on Advanced Concepts for Intelligent Vision Systems*, pp. 675–687, Springer, 2017.

[21] Y. Zhou, C. Barnes, J. Lu, J. Yang, and H. Li, "On the continuity of rotation representations in neural networks," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5745–5753, California, USA, 2019.

[22] V. Peretroukhin, B. Wagstaff, and J. Kelly, "Deep probabilistic regression of elements of so (3) using quaternion averaging and uncertainty injection," in *CVPR Workshops*, pp. 83–86, California, USA, 2019.

[23] Y. Caspi and M. Irani, "Aligning non-overlapping sequences," *International Journal of Computer Vision*, vol. 48, no. 1, pp. 39–51, 2002.

[24] Z. Yang, S. Yan, and Q. Huang, "Extreme relative pose network under hybrid representations," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2455–2464, 2020.

[25] S. Saito, Z. Huang, R. Natsume, S. Morishima, A. Kanazawa, and H. Li, "PIFu: pixel-aligned implicit function for high-resolution clothed human digitization," in *IEEE/CVF International Conference on Computer Vision*, pp. 2304–2314, California, USA, 2019.

[26] Z. Li, T. Yu, C. Pan, Z. Zheng, and Y. Liu, "Robust 3d self-portraits in seconds," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1344–1353, 2020.

[27] M. Yan, S. Li, C. A. Chan, Y. Shen, and Y. Yu, "Mobility prediction using a weighted Markov model based on Mobile user classification," *Sensors*, vol. 21, no. 5, p. 1740, 2021.

[28] M. Yan, H. Yuan, Z. Li, Q. Lin, and J. Li, "Energy savings of wireless communication networks based on mobile user environmental prediction," *Journal of Environmental Protection and Ecology*, vol. 22, no. 1, pp. 206–217, 2021.

[29] M. Yan, X. Lou, and Y. Wang, "Channel noise optimization of polar codes decoding based on a convolutional neural network," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 1434347, 10 pages, 2021.

[30] S. Saito, T. Simon, J. Saragih, and H. Joo, "PIFuHD: multi-level pixel-aligned implicit function for high-resolution 3d human digitization," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 84–93, 2020.

[31] B. L. Bhatnagar, C. Sminchisescu, C. Theobalt, and G. Pons-Moll, "Combining implicit function learning and parametric models for 3d human reconstruction," 2020, https://arxiv.org/abs/2007.11432.

[32] S. Lombardi, T. Simon, J. Saragih, G. Schwartz, A. Lehrmann, and Y. Sheikh, "Neural volumes: learning dynamic renderable volumes from images," 2019, https://arxiv.org/abs/1906.07751.

[33] S. Liu, Y. Zhang, S. Peng, B. Shi, M. Pollefeys, and Z. Cui, "DIST: rendering deep implicit signed distance function with differentiable sphere tracing," in *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2019–2028, 2020.

[34] W. Yifan, F. Serena, S. Wu, C. Öztireli, and O. Sorkine-Hornung, "Differentiable surface splatting for point-based geometry processing," *ACM Transactions on Graphics (TOG)*, vol. 38, no. 6, pp. 1–14, 2019.

[35] W. E. Lorensen and H. E. Cline, "Marching cubes: a high resolution 3d surface construction algorithm," *ACM siggraph computer graphics*, vol. 21, no. 4, pp. 163–169, 1987.

[36] M. A. Bautista, W. Talbott, S. Zhai, N. Srivastava, and J. M. Susskind, "On the generalization of learning-based 3d reconstruction," in *IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2180–2189, 2021.

[37] L. Carlone, R. Tron, K. Daniilidis, and F. Dellaert, "Initialization techniques for 3d slam: a survey on rotation estimation and its use in pose graph optimization," in *IEEE international conference on robotics and automation (ICRA)*, pp. 4597–4604, Seattle, WA, USA, 2015.

[38] J. J. Mor'e, "The Levenberg-Marquardt algorithm: implementation and theory," in *Numerical Analysis*, pp. 105–116, Springer, Berlin, Heidelberg, 1978.

[39] T. Veblen, "The limitations of marginal utility," *Journal of Political Economy*, vol. 17, no. 9, pp. 620–636, 1909.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 8116846, 12 pages
https://doi.org/10.1155/2022/8116846

Research Article

DFAN: Dual Feature Aggregation Network for Lightweight Image Super-Resolution

Shang Li,[1,2] **Guixuan Zhang**,[2] **Zhengxiong Luo**,[1,2] **and Jie Liu**[2]

[1]*School of Artificial Intelligence, University of Chinese Academy of Sciences (UCAS), China*
[2]*Institute of Automation, Chinese Academy of Sciences (CASIA), China*

Correspondence should be addressed to Jie Liu; jie.liu@ia.ac.cn

Received 12 October 2021; Revised 12 November 2021; Accepted 6 December 2021; Published 24 January 2022

Academic Editor: Ming Yan

Copyright © 2022 Shang 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.

With the power of deep learning, super-resolution (SR) methods enjoy a dramatic boost in performance. However, they usually have a large model size and high computational complexity, which hinders the application in devices with limited memory and computing power. Some lightweight SR methods solve this issue by directly designing shallower architectures, but it will adversely affect the representation capability of convolutional neural networks. To address this issue, we propose the dual feature aggregation strategy for image SR. It enhances feature utilization via feature reuse, which largely improves the representation ability while only introducing marginal computational cost. Thus, a smaller model could achieve better cost-effectiveness with the dual feature aggregation strategy. Specifically, it consists of Local Aggregation Module (LAM) and Global Aggregation Module (GAM). LAM and GAM work together to further fuse hierarchical features adaptively along the channel and spatial dimensions. In addition, we propose a compact basic building block to compress the model size and extract hierarchical features in a more efficient way. Extensive experiments suggest that the proposed network performs favorably against state-of-the-art SR methods in terms of visual quality, memory footprint, and computational complexity.

1. Introduction

Single image super-resolution (SISR) aims to reconstruct a visually natural high-resolution (HR) image from its low-resolution (LR) counterpart, which is an inherently ill-posed inverse problem. Due to the essential role in video processing [1], surveillance system [2], and object restoration [3], super-resolution (SR) is still an active research area.

Recently, deep learning-based image super-resolution methods [4–7] have shown prominent performance over conventional methods such as Bicubic interpolation and Lanczos resampling. After the proposal of residual learning [8], which simplifies the optimization of deep convolutional neural networks (CNNs), SR networks tend to become even deeper and larger. However, it is impractical to simply pursue performance gains without considering the model size and computational complexity. For devices with limited memory and battery capacity, cost-effective methods are preferred, which encourages the design of lightweight SR models. To reduce the number of parameters, some approaches adopt a recursive manner or parameter sharing scheme [9, 10]. However, to compensate for the performance drop, these methods have to increase the network width or depth, thus, resulting in high computational complexity as shown in Figure 1. Some other methods directly design shallower network architectures, which reduce parameters and calculations simultaneously. For example, [11, 12] are such compact models with fewer than 40 layers. However, their representation ability is restricted by the shallow architecture.

Towards these drawbacks, we propose Dual Feature Aggregation Network (DFAN) that can strike a better trade-off between SR performance and computational cost as illustrated in Figure 1. The key component of DFAN is the dual feature aggregation strategy. It aggregates local features and global features in a coarse-to-fine manner and

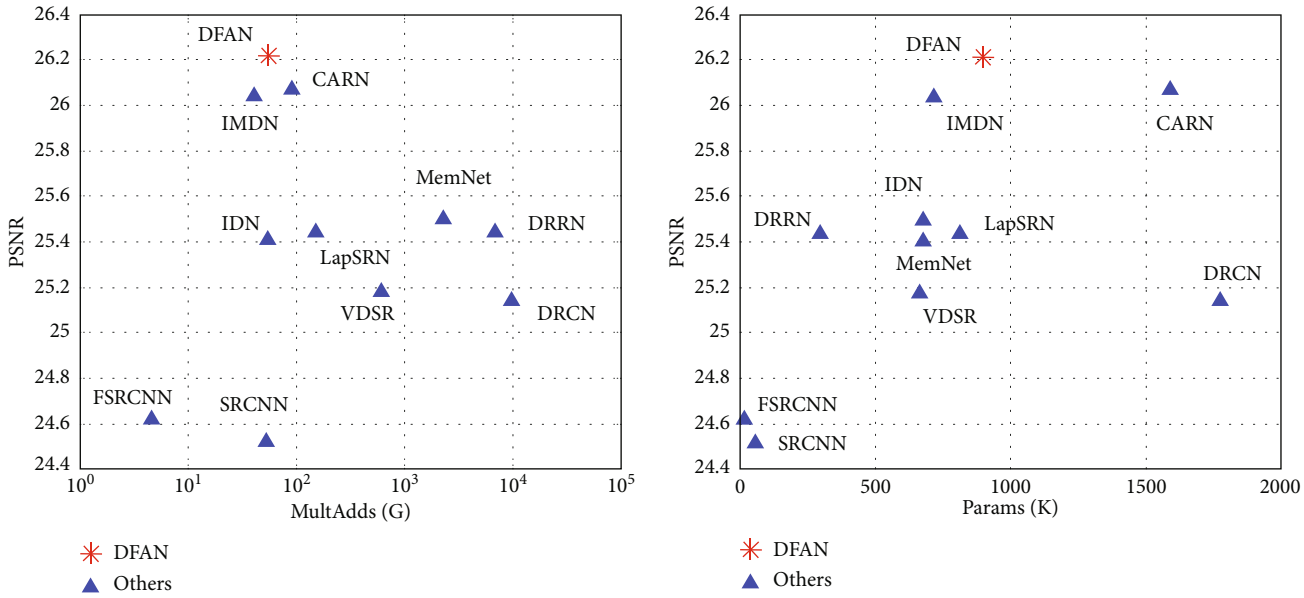


Figure 1: Average PSNR on Urban100 for 4× SR with respect to the number of multiply-adds (MultAdds) and parameters (Params). DFAN achieves the best performance with relatively fewer parameters and lower computational complexity.

could largely improve feature utilization via feature reuse. Specifically, the dual feature aggregation strategy consists of two modules: Local Aggregation Module (LAM) and Global Aggregation Module (GAM). LAM uses an efficient connection method and one convolutional layer to adaptively fuse hierarchical features along the channel dimension. Then, GAM further fuses the local aggregated features along the spatial dimension in an iterative manner. This progressive aggregation strategy fully leverages all hierarchical features, which enables the lightweight model to achieve better SR performance. In this paper, we also design an Efficient Convolutional Block (ECB) as the basic building block of DFAN. It comprises group convolutional layers with channel shuffle operation. Although ECB is compact, DFAN can still achieve competitive results with the help of the dual feature aggregation strategy.

In summary, our main contributions are as follows:

(i) We propose DFAN, which can achieve better SR performance with limited computational cost. It is more practical in real applications

(ii) We propose the dual feature aggregation strategy which aggregates local and global features in a progressive manner. It could make full use of all hierarchical features through feature reuse, which enhances the feature utilization while introducing only marginal computation cost. With our dual feature aggregation strategy, the lightweight SR model can achieve better cost-effectiveness

(iii) We also propose ECB as the basic building structure, which can extract hierarchical features in a computationally economical way

(iv) We show through extensive experiments that our model can achieve competitive results against state-of-the-art methods with relatively fewer parameters and calculations

2. Related Work

2.1. Lightweight SISR. Since Dong et al. [4] first applied CNNs to design Super-Resolution Convolutional Neural Network (SRCNN) and achieved significant improvement, deep learning based SISR methods have been actively explored and shown great advantages in representation capability. To obtain more powerful features for image reconstruction, they continue to enlarge the model size or network depth. Most existing SR methods have hundreds of convolutional layers, such as Residual Channel Attention Network (RCAN) [13], Residual Dense Network (RDN), and Deep Alternating Network (DAN) [14]. However, these methods are computationally expensive for real application. Thus, more and more lightweight SR methods are proposed. Deep Recursive Residual Network (DRRN) [9] and Memory Network (MemNet) [10] introduce recursive learning or weight sharing schemes to reduce parameters. However, they need to increase the computational complexity to compensate for the performance drop. Another idea is to build relatively shallower models, which can cut down the model size and calculations at the same time. Cascading Residual Network (CARN) [11], Information Distillation Network (IDN) [12], and Information Multi-Distillation Network (IMDN) [15] are all lightweight networks that have fewer than 40 layers. However, the shallow architecture could restrict their representation ability to some extent. For our method, we improve the feature utilization through dual feature aggregation, which can better balance the SR performance and computational cost.

2.2. Group Convolution. There has been rising interest in designing small and efficient neural networks [16–19] since

many deep and complicated neural networks are infeasible in practical applications. Group convolution is an important method for designing efficient neural networks. The application of the group convolution method dates back to [20] where the model is distributed over two GPUs, resulting in gains in accuracy and convergence speed. Depthwise convolution is a special case of group convolution and is originally introduced in [21]. In depthwise convolution, the number of groups is equal to the number of channels. Based on the depthwise convolution, Mobile Network (MobileNet) [18] gains state-of-the-art results among lightweight models in many visual tasks. Then, group convolution and depthwise convolution are generalized in a novel form in [22]. Channel shuffle operation is also proposed in [22] to overcome the side effect of group convolution. Recently, group convolution has been used in some lightweight image super-resolution methods. Ahn et al. [11] proposed efficient residual block containing group convolutional layers, and Hui et al. [12] introduced group convolution to some specific layers. However, there is still room for improvement in the reconstruction performance of these two models. In our DFAN, group convolution is used as a basic building unit without affecting the reconstruction performance.

2.3. Deep Feature Aggregation. As the feature representation capability of a single network layer is limited [23, 24], deep feature aggregation is typically used to fuse features of different layers, which can improve the representation capability in a computationally economical way. For instance, the Densely Connected Network (DenseNet) [25] and the Feature Pyramid Network (FPN) [26] are the dominant architectures for semantic feature aggregation and spatial feature aggregation [27]. DenseNet can better propagate features and gradients through dense connections that connect each layer to every other layer in a feed-forward fashion. FPN can equalize resolution and standardize semantics across the levels of a pyramidal feature hierarchy through top-down and lateral connections. Besides, Residual Network (ResNet) [8] is also a typical feature aggregation method which aggregates features via simple element-wise summation. Recently, Yu et al. [28] proposed an iterative aggregation method and a hierarchical aggregation method, which can further improve the performance of the aforementioned dominant architectures in many visual tasks. Inspired by this work, we introduce an iterative and adaptive global feature aggregation module to DFAN, obtaining more comprehensive information and improving reconstruction performance.

3. Proposed Method

3.1. Network Architecture. As depicted in Figure 2(a), DFAN mainly consists of four parts: the shallow feature extraction layer, stacked local feature aggregation modules, the global feature aggregation module, and the upsampling module.

The shallow feature extraction layer contains only one convolutional layer. It extracts shallow features F_0 from the LR image. Then, F_0 is input into the stacked LAMs for global residual learning. There are M stacked LAMs, and the local aggregated feature from the m^{th} LAM can be formulated as

$$F_m = f_{\text{LAM}}^m(F_{m-1}) = f_{\text{LAM}}^m\left(f_{\text{LAM}}^{m-1}\left(\cdots\left(f_{\text{LAM}}^1(\text{F}_0)\right)\right)\right), \quad (1)$$

where f_{LAM}^m refers to the operation of the m^{th} LAM, and F_m is the local aggregated feature from it. As shown in Figure 2, each LAM is composed of a series of ECBs, therefore, f_{LAM}^m can be viewed as a composite function.

After that, GAM fully leverages local aggregated features from LAMs in an iterative way, which can be expressed as

$$F_A = f_{\text{GAM}}(F_1, F_2, \cdots, F_G), \quad (2)$$

where F_A is the global aggregated feature. f_{GAM} denotes the operation of GAM. Then, the global long skip connection adds F_0 to F_A, obtaining the final aggregated feature F. The global skip connection can better propagate information and gradients, thus, stabilizing the training of DFAN.

Finally, we use an upscale module proposed in [29] to restore the final SR image I_{SR}. That is,

$$I_{\text{SR}} = f_{\text{conv}}\left(f_{\uparrow}\left(f_{g\text{conv}}(F)\right)\right) = f_{\text{DFAN}}(I_{\text{LR}}), \quad (3)$$

where $f_{g\text{conv}}$ denotes the group convolution, f_{conv} indicates the standard convolution, and $f_{\uparrow}$ is the upscaling module.

3.2. Local Feature Aggregation. Since features of different layers contain different weighted information, adaptively aggregating all hierarchical features could effectively improve the representation ability. Referring to [28], the key axes of feature fusion are semantic and spatial, which are closely related to channel and spatial dimensions, respectively. Thus, we propose the dual feature aggregation strategy, in which features are locally aggregated along the channel dimension, and then globally aggregated along the spatial dimension. In this subsection, we first explain the local feature aggregation.

3.2.1. Efficient Convolutional Block. As depicted in Figure 2(c), ECB is the basic building block of LAM. ECB is a residual learning module consisting of two group convolutional layers with channel shuffle operation [22] and a channel attention module [7]. Group convolution with channel shuffle operation can extract useful features in a computationally economical way. Assuming the group size of an $s \times s$ group convolutional kernel is g, the parameter amount and computation complexity of this group convolutional kernel will be both $1/g$ of an $s \times s$ standard convolutional kernel. Moreover, the channel shuffle operation enhances the information exchange among channels without extra parameters and calculations. There are B ECBs in each LAM. LAM fuses hierarchical features from ECBs by exploring the interchannel relationship. The local aggregated feature F_m from the m^{th} LAM can be obtained by

$$F_m = f_{\text{conv}}\left(\left[F_m^1, F_m^2 \cdots, F_m^B\right]\right) + F_{m-1}, \quad (4)$$

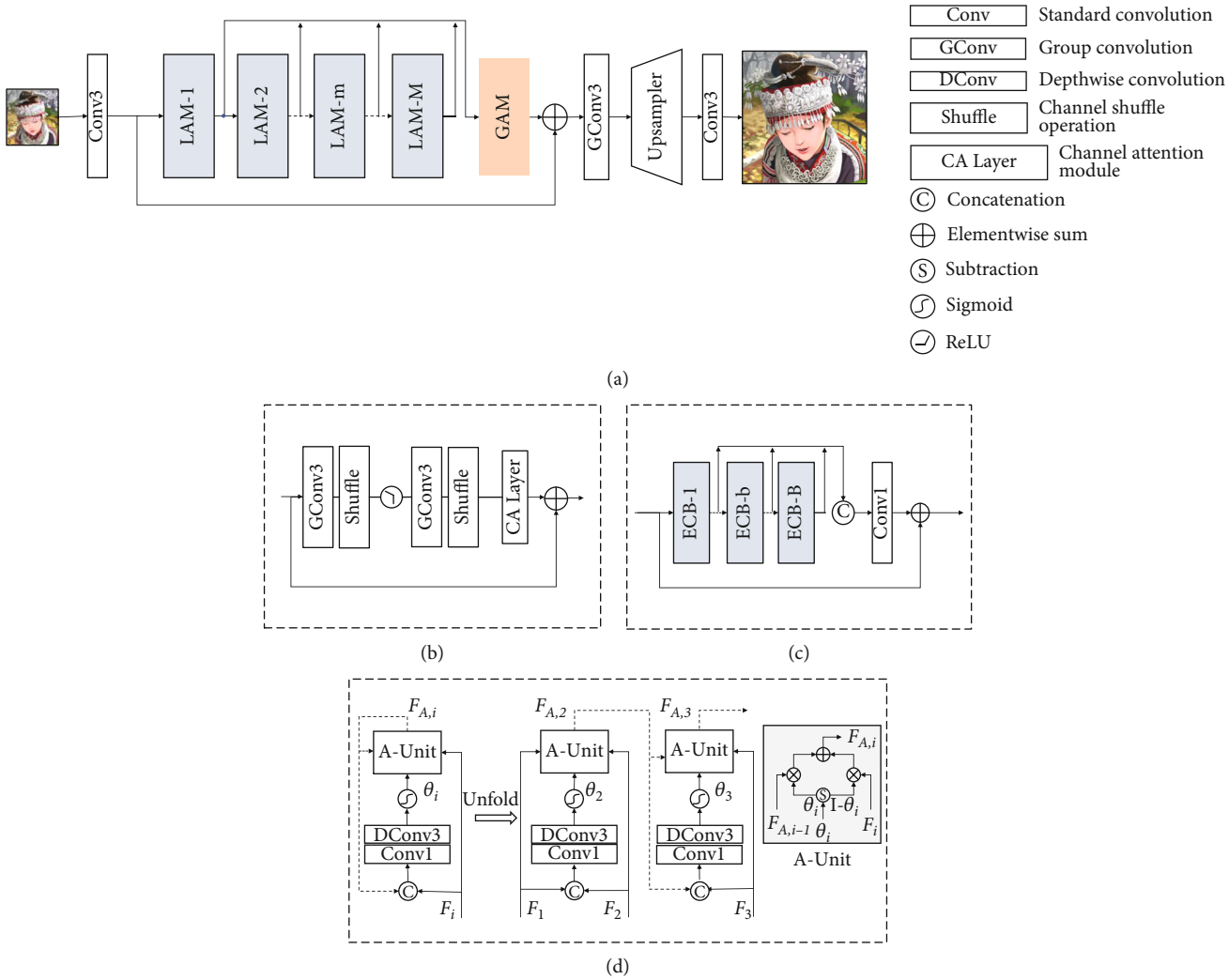


Figure 2: Framework of DFAN and its submodules. (a) Architecture of DFAN. (b) Efficient convolutional block (ECB). (c) Local aggregation module (LAM). (d) Global aggregation module (GAM).

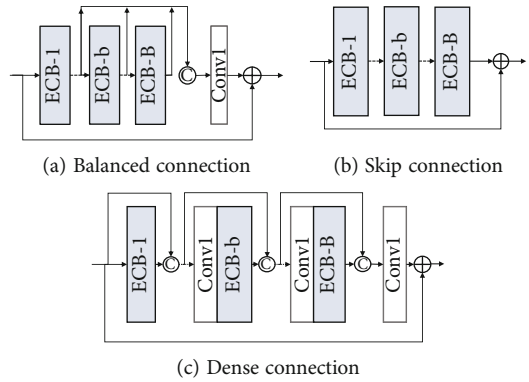
where $[F_m^1, F_m^2 \cdots, F_m^B]$ represents the concatenation of local features from ECBs in the m^{th} LAM.

3.2.2. Balanced Connection. The connection method in LAM is what we call balanced connection. As shown in Figure 3, compared with two commonly used connection methods in SR, i.e., skip connection and dense connection, our balanced connection is more flexible than skip connection and more lightweight than dense connection. The analysis is as follows:

(1) *Difference to Skip Connection.* As shown in Figure 3(b), for each LAM, if we only use skip connection which makes the elementwise sum of the hierarchical feature maps, all hierarchical features will contribute equally to the final aggregated feature. It may be inflexible since different features contain information of different importance. Our balanced connection can simply solve this issue by a 1×1 convolutional kernel. This convolutional kernel assigns specific learned weights to each pixel of local features, thus, adaptively aggregating them along the channel dimension

(2) *Difference to Dense Connection.* As shown in Figure 3(c), dense connection connects each ECB and all preceding ECBs to be concatenated and compressed as inputs to all subsequent ECBs, which requires more 1×1 convolutional kernels and harms the overall efficiency. However, our balanced connection directly connects each ECB for feature aggregation, which not only fully uses local features but also greatly reduces the number of parameters and computation operations

3.3. Global Feature Aggregation. The spatial dimension is orthogonal to the channel dimension. Thus, further fusing local aggregated features along the spatial dimension could supplement more information. Besides, since local aggregated features contain abundant information, it could be suitable to aggregate them in a coarse to fine fashion.



(a) Balanced connection (b) Skip connection

(c) Dense connection

Figure 3: Comparisons of three different connection methods.

Therefore, we design GAM, which can further fuse local aggregated features with spatial attention mechanism in an iterative manner.

In Figure 2(d), $F_{A,i}$ represents the global aggregated feature in the i^{th} iteration, and F_i represents the output of the i^{th} LAM. The iterative fusion of GAM can be formulated as

$$F_{A,i} = \begin{cases} F_1, & i = 1, \\ f_G([F_{A,i-1}, F_i]), & i > 1, \end{cases} \tag{5}$$

where $F_{A,i}$ is initialized with F_1, which is the output of the first LAM. f_G represents the global aggregation of GAM.

The main parts of GAM are (1) spatial attention generation and (2) iterative feature aggregation. First, the spatial attention θ_i is generated by the following operation,

$$\theta_i = \sigma(f_{d\text{conv}}(f_{\text{conv}}([F_{A,i-1}, F_i]))), \tag{6}$$

where f_{conv} denotes a 1×1 convolutional kernel that reduces the channel number of $[F_{A,i}, F_i]$ by half. $f_{d\text{conv}}$ denotes a 3×3 depthwise convolutional kernel to extract spatial information. Depthwise convolution applies a single filter to each input channel, which is more efficient than common convolution in terms of memory and computation. σ is the Sigmoid activation function constraining the spatial attention to $(0, 1)$. The spatial attention θ_i is the same size as F_i and $F_{A,i-1}$. Second, as shown in Figure 2(d), the feature fusion in A-Unit can be formulated as

$$F_{A,i} = \theta_i \otimes F_{A,i-1} + (I - \theta_i) \otimes F_i, \tag{7}$$

where $\otimes$ denotes the Hadamard product. I is the tensor with all elements being 1.

After the G^{th} iteration, we obtain the final global aggregated feature F_A. The overall iterative global aggregation can be summarized as follows,

$$\begin{cases} \sum_{i=1}^{G} \Theta_i = I, \\ F_A = \sum_{i}^{G} \Theta_i \otimes F_i, \end{cases} \tag{8}$$

where Θ_i denotes the final spatial attention for F_i, which is determined by all the local aggregated features from LAMs, thus, highly comprehensive. Additionally, Eq. (8) indicates that the global feature aggregation strategy satisfies the convex combination.

Table 1: Investigation of group convolution and standard convolution. We observe the best performance on PSNR with scaling factor $\times 4$.

	Conv	GConv	Set5	Set14	BSD100	Urban100
DFAN_W	✓	—	32.00	28.45	27.48	25.80
DFAN_D	✓	—	32.27	28.61	27.58	26.20
DFAN	—	✓	32.29	28.65	27.60	26.22

4. Experiments

4.1. Experimental Setup

4.1.1. Datasets and Metrics. We use the training set of DIV2K [30] to train all of our models. For testing, we use five standard benchmark datasets: Set5 [31], Set14 [32], BSD100 [33], Urban100 [34], and Manga109 [35]. The visual quality of SR results is evaluated with Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity (SSIM) [36] on the Y channel (i.e., luminance) of transformed YCbCr space. We also represent the number of parameters and multiply-adds to evaluate the memory footprint and computation complexity, respectively.

4.1.2. Degradation Models. To fully demonstrate the effectiveness of our DFAN, we use two degradation models to

Table 2: Study on the dual feature aggregation strategy. PSNR for 4× SR. "LA" denotes local feature aggregation, and "GA" denotes global feature aggregation.

	LAM	GAM	Set5	Set14	BSD100	Urban100
LAM0_GAM0	—	—	32.11/0.8937	28.50/0.7797	27.52/0.7343	25.96/0.7809
LAM1_GAM0	✓	—	32.16/0.8944	28.57/0.7814	27.57/0.7361	26.11/0.7864
LAM1_GAM1	✓	✓	32.29/0.8964	28.65/0.7832	27.60/0.7373	26.22/0.7898

simulate LR images. The first is the bicubic degradation model. The bicubic degradation model simulates LR images on scale $\times 2$, $\times 3$, and $\times 4$. The second is the blur-down degradation model that blurs HR images by 7×7 Gaussian kernel with a standard deviation 1.6. The blurred image is then downsampled on scale $\times 3$.

4.1.3. Training Details. The size of LR patches is 48×48. During training, we randomly rotate input images by 90°, 180°, or 270° and flip them horizontally or vertically. The batch size is 32. We use $L1$ loss as the loss function. We use Adam as the optimizer. The initial learning rate is $2e-4$, decayed by half every 200 epochs. We train our model for 1000 epochs.

4.2. Study on Efficient Convolutional Block. Different from most of the super-resolution networks, our DFAN uses group convolutional kernels instead of standard convolutional kernels in an ECB to extract features. Since group convolution is a basic operation of our ECB, we design DFAN_W and DFAN_D to validate the effectiveness of ECB. These two models have the same structure as DFAN, but group convolutional kernels in ECBs are replaced with standard convolutional kernels. All three models have similar number of parameters and computation operations, i.e., approximately 900 K and 60 G, respectively.

We denote the number of ECBs in each LAM as B, the number of LAMs as M, and the number of channels of each intermediate feature as C. For our DFAN, we set B, M, and C to 10, 6, and 64, respectively, and the group number of each group convolutional kernel in ECBs is 8. We set B, M, and C to 3, 2, and 64, respectively, for DFAN_W, and these hyperparameters to 10, 6, and 27, respectively, for DFAN_D. In other words, the width of DFAN_W is the same as DFAN. While the depth of DFAN_D is the same as DFAN.

As shown in Table 1, group convolution makes an outstanding trade-off between representation capability and computational costs. Compared with standard convolution, group convolution can make the model deeper or wider with limited parameters and calculations, which is beneficial to obtain richer hierarchical information.

4.3. Study on Dual Feature Aggregation. In this section, we experimentally investigate the effectiveness of the dual feature aggregation strategy. LAM0_GAM0 is the baseline network by removing balanced connections in LAM and GAM from DFAN. LAM1_GAM0 is built by removing GAM from DFAN. LAM1_GAM1 has both LAM and GAM, which is the same as DFAN. As shown in Table 2, when only LAM is added, PSNR is improved by approximately 0.05 dB. When both LAM and GAM are added, the performance is improved by a large margin (PSNR: +0.15 dB on Set14).

4.3.1. LAM Analysis. To intuitively show the effectiveness of LAM, we plot the training curves of LAM0_GAM0 and LAM1_GAM0 in Figure 4(a). Benefitting from the balanced connection in LAM, gradients could be better propagated. The margin between the two curves indicates that LAM could not only help the network converge faster but also help it converge to a better point. Additionally, the weight distribution is visualized in Figure 4(b). This indicates how much information of each ECB in an LAM contributes to the local aggregated feature generated by this LAM. Features from different ECBs contribute differently to local aggregated features, which suggest that LAM could adaptively aggregate hierarchical features to improve the final performance.

4.3.2. GAM Analysis. We experimentally prove that GAM also works well for some other networks. We use a shallower RCAN [7] as the baseline network (denoted as sRCAN). To facilitate network training, we set the RG number to 3, and the RCAB number to 5 for sRCAN. Then, we apply our GAM to sRCAN, which is denoted as sRCAN_GAM. As Table 3 shows, with only a small increase in parameters and computational complexity (Paramerters:+9 K, MultAdds:+0.9G), GAM can significantly improve the SR performance on all the benchmark datasets with scaling factor $\times 4$. Therefore, GAM could be used as a general lightweight tool to improve the performance of some existing SR methods.

To better understand the adaptive and iterative aggregation strategy of GAM, we visualize the spatial attention heatmaps generated by GAM in Figure 5. The 3D attention is transformed to 2D by taking the absolute mean along the channel dimension and then normalized to $[0, 1]$ over the spatial dimension. We can see that (1) spatial attention for different LAMs focuses on regions of different frequencies. For example, the spatial attention for LAM_1 (Figure 5(a)) focuses on low-frequency regions such as the background. While the spatial attention for LAM_6 (Figure 5(f)) focuses more on high-frequency regions with rich textures. Thus, both high-frequency and low-frequency information is important for SR. (2) Although some spatial attention focuses on high-frequency regions, they emphasize different parts. In LAM_6, more attention is given to regions of the main object. But in LAM_5, high-frequency regions in the background are emphasized. It indicates that GAM provides

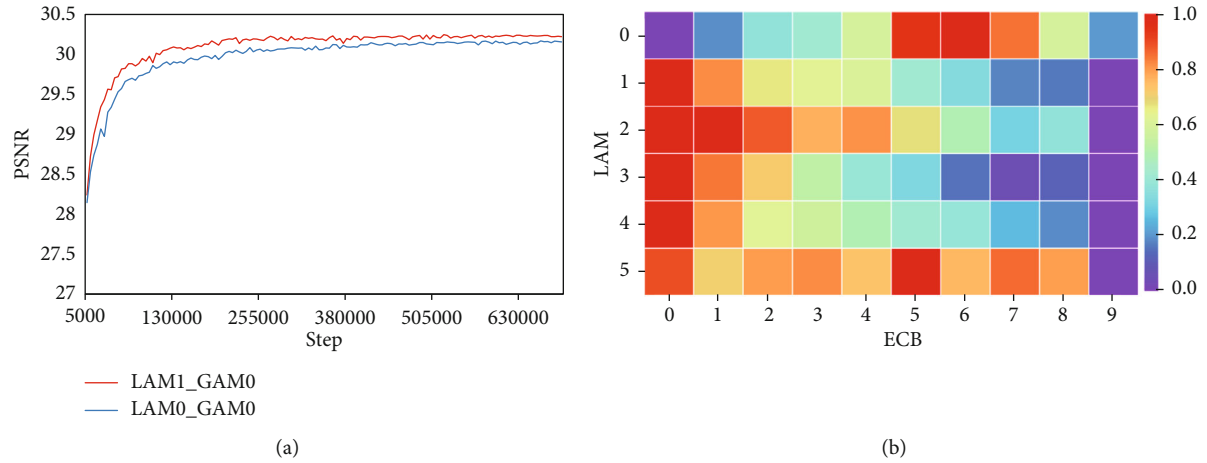


Figure 4: (a) Training curves of LAM0_GAM0 and LAM1_GAM0. (b) Weight distribution of ECBs in each LAM. The color pixel (b, m) encodes the average $L1$ norm of the weights connecting the feature from ECB-b in LAM-m to the local aggregated feature of LAM-m.

Table 3: Study on GAM. The computation cost and average PSNR/SSIM for $4\times$ SR.

Method	sRCAN	sRCAN_GAM
Params(K)/MultAdds(G)	208/16.1	217/17.0
Set5	31.69/0.8878	31.84/0.8901
Set14	28.29/0.7741	28.36/0.7763
BSD100	27.38/0.7292	27.42/0.7311
Urban100	25.53/0.7667	25.60/0.7702
Manga109	29.59/0.8955	29.73/0.8985

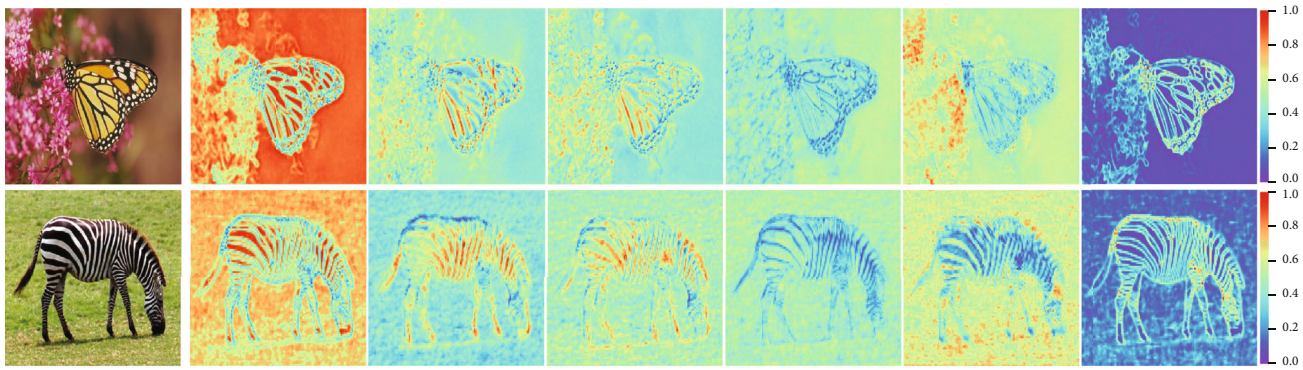


Figure 5: Spatial attention heatmaps generated by GAM for each LAM.

additional flexibility to deal with different types of information, which could enhance the representation capability.

4.4. Results with Bicubic Degradation Model. We compare DFAN with other state-of-the-art methods: SRCNN [4], Fast Super-Resolution Convolutional Neural Network (FSRCNN) [37], Very Deep Super-Resolution(VDSR) [5], Deeply-Recursive Convolutional Network (DRCN) [38], DRRN [9], MemNet [10], CARN [11], IDN [12], and IMDN [15].

4.4.1. Quantitative Results. We evaluate the average PSNR and SSIM on five benchmark datasets. In particular, we also calculate the number of parameters and multiply-adds of these models by assuming the HR image size to be 720p (1280×720). In Table 4, the proposed DFAN performs favorably against these methods on all benchmark datasets for $2\times$, $3\times$, and $4\times$ SR. Note that the number of parameters of our method is inconsistent for different scales because we apply the pixelshuffle operation [29] for upscaling, and the convolutional kernels in the upscaling module are of different sizes for different scales. CARN [11] used to be a strong baseline for lightweight SR models, but our DFAN outperforms it by a large margin (PSNR:+0.17 dB,

Table 4: Quantitative results with the bicubic degradation model. Best results are highlighted.

Method	Scale	Params (K)	MultAdds (G)	Set5		Set14		BSD100		Urban100		Manga109	
				PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Bicubic	×2	—	—	33.66	0.9299	30.24	0.8688	29.56	0.8431	26.88	0.8403	30.80	0.9339
SRCNN [4]	×2	57	52.7	36.66	0.9542	32.42	0.9063	31.36	0.8879	29.50	0.8946	35.60	0.9663
FSRCNN [37]	×2	12	6.0	37.00	0.9558	32.63	0.9088	31.53	0.8920	29.88	0.9020	36.67	0.9710
VDSR [2]	×2	665	612.6	37.53	0.9587	33.03	0.9124	31.90	0.8960	30.76	0.9140	37.22	0.9750
DRCN [38]	×2	1774	9788.7	37.63	0.9588	33.04	0.9118	31.85	0.8942	30.75	0.9133	37.55	0.9732
LapSRN [39]	×2	813	29.9	37.52	0.9590	33.08	0.9130	31.80	0.8950	30.41	0.9100	37.27	0.9740
DRRN [9]	×2	297	6796.9	37.74	0.9591	33.23	0.9136	32.05	0.8973	31.23	0.9188	37.88	0.9749
MemNet [10]	×2	677	2262.4	37.78	0.9597	33.28	0.9142	32.08	0.8978	31.31	0.9195	37.72	0.9740
IDN [12]	×2	677	168.4	37.83	0.9600	33.30	0.9148	32.08	0.8985	31.27	0.9196	38.01	0.9749
IMDN [15]	×2	694	158.8	38.00	0.9605	33.63	0.9177	32.19	0.8996	32.17	0.9283	38.88	0.9774
CARN [11]	×2	1592	222.8	37.76	0.9590	33.52	0.9166	32.09	0.8978	31.51	0.9312	38.36	0.9765
DFAN	×2	877	201.4	38.11	0.9609	33.74	0.9189	32.21	0.9002	32.36	0.9305	38.93	0.9774
Bicubic	×3	—	—	30.39	0.8682	27.55	0.7742	27.21	0.7385	24.46	0.7349	26.95	0.8556
SRCNN [4]	×3	57	52.7	32.75	0.9090	29.28	0.8209	28.41	0.7863	26.24	0.7989	30.48	0.9117
FSRCNN [37]	×3	12	5.0	33.16	0.9140	29.43	0.8242	28.53	0.7910	26.43	0.8080	31.10	0.9210
VDSR [5]	×3	665	612.6	33.66	0.9213	29.77	0.8314	28.82	0.7976	27.14	0.8279	32.01	0.9340
DRCN [38]	×3	1774	9788.7	33.82	0.9226	29.76	0.8311	28.80	0.7963	27.15	0.8276	32.24	0.9343
DRRN [9]	×3	297	6796.9	34.03	0.9244	29.96	0.8349	28.95	0.8004	27.53	0.8378	32.71	0.9379
MemNet [10]	×3	677	2262.4	34.09	0.9248	30.00	0.8350	38.96	0.8001	27.56	0.8376	32.51	0.9369
IDN [12]	×3	677	84.4	34.11	0.9253	29.99	0.8354	28.95	0.8031	27.42	0.8359	32.71	0.9381
IMDN [15]	×3	703	71.5	34.36	0.9270	30.32	0.8417	29.09	0.8046	28.17	0.8519	33.61	0.9445
CARN [11]	×3	1592	118.8	34.29	0.9255	30.29	0.8407	29.06	0.8034	27.38	0.8404	33.50	0.9440
DFAN	×3	900	92.6	34.49	0.9280	30.37	0.8426	29.13	0.8059	28.31	0.8551	33.78	0.9458
Bicubic	×4	—	—	28.42	0.8104	26.00	0.7027	25.96	0.6675	23.14	0.6577	24.89	0.7866
SRCNN [4]	×4	57	52.7	30.48	0.8628	27.49	0.7503	26.90	0.7101	24.52	0.7221	27.58	0.8555
FSRCNN [37]	×4	12	4.6	30.71	0.8657	27.59	0.7535	26.98	0.7150	24.62	0.7280	27.90	0.8610
VDSR [5]	×4	665	612.6	31.35	0.8838	28.01	0.7674	27.29	0.7251	25.18	0.7524	28.83	0.8870
DRCN [38]	×4	1774	9788.7	31.53	0.8854	28.02	0.7670	27.23	0.7233	25.14	0.7510	28.93	0.8854
LapSRN [39]	×4	813	149.4	31.54	0.8850	28.19	0.7720	27.32	0.7280	25.44	0.7638	29.09	0.8900
DRRN [9]	×4	297	6796.9	31.68	0.8888	28.21	0.7720	27.38	0.7284	25.44	0.7638	29.45	0.8946
MemNet [10]	×4	677	2262.4	31.74	0.8893	28.26	0.7723	27.40	0.7281	25.50	0.7630	29.42	0.8942
IDN [12]	×4	677	54.9	31.82	0.8903	28.25	0.7730	27.41	0.7297	25.41	0.7632	29.41	0.8942
IMDN [15]	×4	715	40.9	32.21	0.8948	28.58	0.7811	27.56	0.7353	26.04	0.7838	30.45	0.9075
CARN [11]	×4	1592	90.9	32.13	0.8937	28.60	0.7806	27.58	0.7349	26.07	0.7837	30.47	0.9084
DFAN	×4	896	55.8	32.30	0.8964	28.65	0.7832	27.60	0.7373	26.22	0.7898	30.59	0.9097

SSIM:+0.0024 on Set5) with 42% fewer parameters and 37% fewer MultAdds on scale ×4. It indicates that our method can achieve a better trade-off between computational cost and effectiveness. Therefore, feature aggregation has promising prospects in the research of lightweight image SR.

4.4.2. Visual Results. In Figure 6, we show visual comparisons on scale ×4. Our method restores the letter "*g*" in "*ppt3*" more clearly, while most other methods encounter artifacts or edge distortion. For "*img030*" in Urban100 and "*img86000*" in BSD100, most methods do not reconstruct the contour of the window well, but our method can reconstruct these edges better.

4.5. Results with Blur-Down Degradation Model. As mentioned in the main submission, we further apply our method

Figure 6: Visual results for 4× SR with the bicubic degradation model.

Table 5: Quantitative results for 3× SR with the blur-down degradation model. Best results are highlighted.

Model	Scale	Params (K)	MultAdds (G)	Set5		Set14		BSD100		Urban100		Manga109	
				PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Bicubic	×3	—	—	28.78	0.8308	26.38	0.7271	26.33	0.6918	23.52	0.6862	25.46	0.8149
SRCNN [4]	×3	57	52.7	32.75	0.9090	29.28	0.8209	28.41	0.7863	26.24	0.7989	29.47	0.8924
FSRCNN [37]	×3	12	5.0	33.16	0.9140	29.43	0.8242	28.53	0.7910	26.43	0.8080	23.04	0.7927
VDSR [5]	×3	665	612.6	33.66	0.9213	29.77	0.8314	28.82	0.7976	27.14	0.8279	31.06	0.9234
IDN [12]	×3	677	84.4	34.30	0.9256	30.33	0.8400	29.08	0.8034	27.91	0.8455	33.54	0.9427
IMDN [15]	×3	703	71.5	34.38	0.9263	30.32	0.8401	29.09	0.8038	28.03	0.8477	33.70	0.9438
CARN [11]	×3	1592	118.8	34.40	0.9264	30.39	0.8415	29.13	0.8046	28.07	0.8489	33.69	0.9442
DFAN	×3	900	92.6	34.50	0.9274	30.43	0.8419	29.17	0.8058	28.27	0.8526	33.99	0.9456

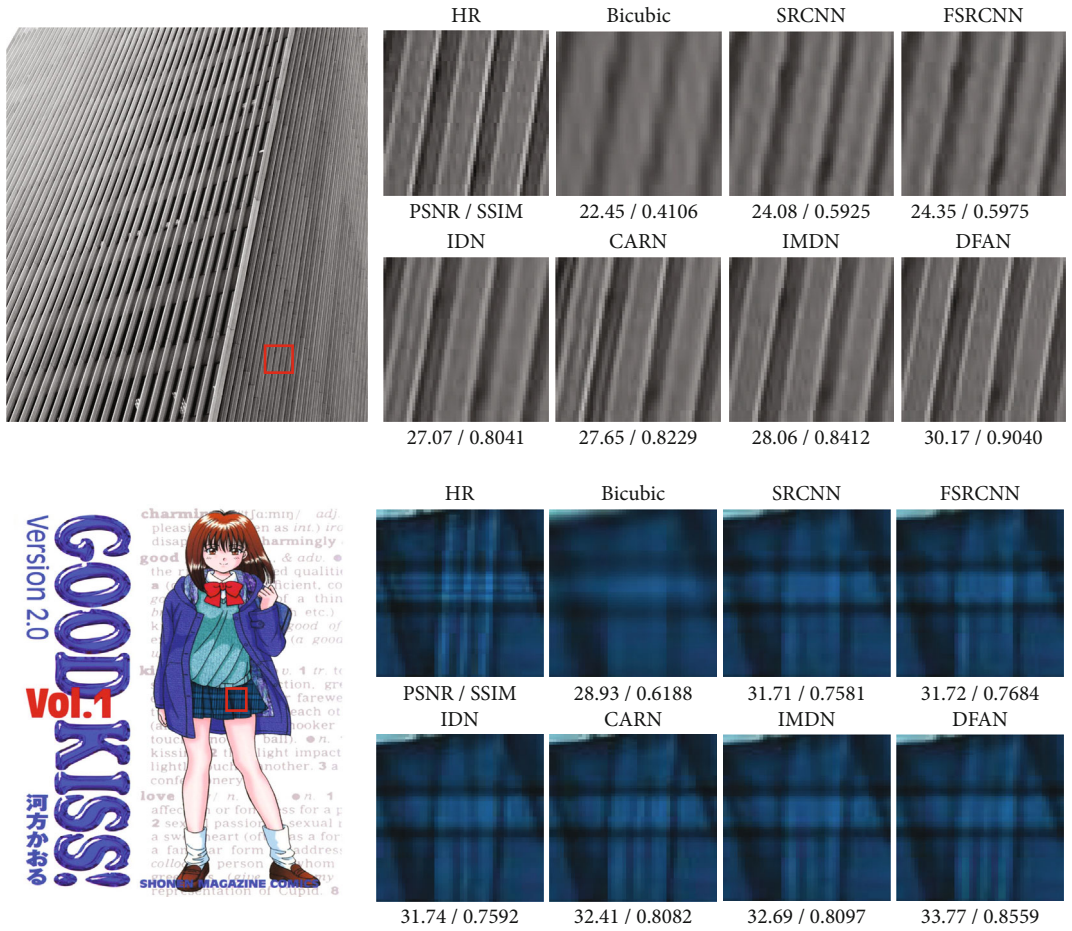


Figure 7: Visual results for 3× SR with the blur-down degradation model.

to super-resolve images with blur-down degradation, which is also commonly used in [7, 13]. We compare DFAN with SRCNN [4], FSRCNN [37], VDSR [5], CARN [11], IDN [12], and IMDN [15].

As shown in Table 5, compared with the networks that are stacked by several elaborately designed building blocks, such as CARN, IDN, and IMDN, our lightweight network with the dual feature aggregation strategy can better leverage the hierarchical features. In addition, the visual comparison in Figure 7 also demonstrates the superiority of our method.

5. Conclusions

We propose DFAN that can strike a better trade-off between SR performance and computational cost. The proposed dual feature aggregation strategy makes local and global feature aggregations adaptively. Through feature reuse, it could simultaneously improve feature utilization and representation ability. Benefitting from the dual feature aggregation strategy, our network achieves competitive performances with fewer parameters and lower computational complexity, which is more practical for real applications.

Data Availability

The image datasets supporting this work are from previously reported studies and datasets, which have been cited. The processed data are available at the repository: BasicSR(https://github.com/xinntao/BasicSR/blob/master/docs/DatasetPreparation.md#Image-Super-Resolution).

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Key R&D Program of China (2019YFB1406200) and was also the research achievement of the Key Laboratory of Digital Rights Services. It is based on our previous teamwork, Lightweight Image Super-Resolution via Dual Feature Aggregation Network, presented in 2021 at the 2nd International Conference on Culture-oriented Science & Technology (ICCST).

References

[1] X. Wang, K. C. Chan, K. Yu, C. Dong, and C. Change Loy, "Edvr: video restoration with enhanced deformable convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, Long Beach California, USA, 2019.

[2] J. Kim, G. Li, I. Yun, C. Jung, and J. Kim, "Edge and identity preserving network for face super-resolution," *Neurocomputing*, vol. 446, pp. 11–22, 2021.

[3] J. Peng, K. Fu, Q. Wei, Y. Qin, and Q. He, "Improved multiview decomposition for single-image high-resolution 3D object reconstruction," *Wireless Communications and Mobile Computing*, vol. 2020, 14 pages, 2020.

[4] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 2, pp. 295–307, 2016.

[5] J. Kim, J. Kwon Lee, and K. Mu Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1646–1654, Las Vegas Nevada, USA, 2016.

[6] B. Lim, S. Son, H. Kim, S. Nah, and K. Mu Lee, "Enhanced deep residual networks for single image super resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 136–144, Honolulu Hawaii, USA, 2017.

[7] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proceedings of the European Conference on Computer Vision*, pp. 286–301, 2018.

[8] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 770–778, Las Vegas Nevada, USA, 2016.

[9] Y. Tai, J. Yang, and X. Liu, "Image super-resolution via deep recursive residual network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3147–3155, Honolulu, Hawaii, 2017.

[10] Y. Tai, J. Yang, X. Liu, and C. Xu, "Memnet: a persistent memory network for image restoration," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 4539–4547, Venice, Italy, 2017.

[11] N. Ahn, B. Kang, and K.-A. Sohn, "Fast, accurate, and lightweight super-resolution with cascading residual network," in *Proceedings of the European Conference on Computer Vision*, pp. 252–268, Munich, Germany, 2018.

[12] Z. Hui, X. Wang, and X. Gao, "Fast and accurate single image super-resolution via information distillation network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 723–731, Salt Lake City Utah, 2018.

[13] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2472–2481, Salt Lake City Utah, 2018.

[14] Z. Luo, Y. Huang, S. Li, L. Wang, and T. Tan, "Unfolding the alternating optimization for blind super resolution," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 33, 2020.

[15] Z. Hui, X. Gao, Y. Yang, and X. Wang, "Lightweight image super-resolution with information multidistillation network," in *Proceedings of the 27th ACM International Conference on Multimedia*, pp. 2024–2032, Nice, France, 2019.

[16] M. Wang, B. Liu, and H. Foroosh, "Factorized convolutional neural networks," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 545–553, Venice, Italy, 2017.

[17] F. N. Iandola, S. Han, M. W. Moskewicz, K. Ashraf, W. J. Dally, and K. Keutzer, "Squeezenet: Alexnet-level accuracy with 50x fewer parameters and<0.5 mb model size," 2016, https://arxiv.org/abs/1602.07360.

[18] A. G. Howard, M. Zhu, B. Chen et al., "Mobilenets: efficient convolutional neural networks for mobile vision applications," 2017, https://arxiv.org/abs/1704.04861.

[19] J. Wu, C. Leng, Y. Wang, Q. Hu, and J. Cheng, "Quantized convolutional neural networks for mobile devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4820–4828, Las Vegas Nevada, USA, 2016.

[20] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in Neural Information Processing Systems*, vol. 25, pp. 1097–1105, 2012.

[21] L. Sifre and S. Mallat, "Rigid-motion scattering for texture classification," 2014, https://arxiv.org/abs/1403.1687.

[22] X. Zhang, X. Zhou, M. Lin, and J. Sun, "Shufflenet: an extremely efficient convolutional neural network for mobile devices," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 6848–6856, Salt Lake City Utah, USA, 2018.

[23] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *European Conference on Computer Vision*, pp. 818–833, Springer, Cham, 2014.

[24] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3431–3440, Boston Massachusetts, USA, 2015.

[25] G. Huang, Z. Liu, L. Van Der Maaten, and K. Q. Weinberger, "Densely connected convolutional networks," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4700–4708, Honolulu Hawaii, USA, 2017.

[26] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2117–2125, 2017.

[27] J. Liu, R. Jia, W. Li, F. Ma, and X. Wang, "Image dehazing method of transmission line for unmanned aerial vehicle inspection based on densely connection pyramid network," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8857271, 9 pages, 2020.

[28] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep layer aggregation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2403–2412, Salt Lake City Utah, USA, 2018.

[29] W. Shi, J. Caballero, F. Huszár et al., "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1874–1883, Las Vegas Nevada, USA, 2016.

[30] E. Agustsson and R. Timofte, "Ntire 2017 challenge on single image super-resolution: dataset and study," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 126–135, Honolulu Hawaii, USA, 2017.

[31] M. Bevilacqua, A. Roumy, C. Guillemot, and M.-L. A. Morel, "Low-complexity single-image superresolution based on nonnegative neighbor embedding," in *British Machine Vision Conference*, BMVA Press, 2012.

[32] R. Zeyde, M. Elad, and M. Protter, "On single image scale-up using sparse-representations," in *International Conference on Curves and Surfaces*, pp. 711–730, Springer, 2010.

[33] D. Martin, C. Fowlkes, D. Tal, and J. Malik, "A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics," in *Proceedings Eighth IEEE International Conference on Computer Vision. ICCV*, pp. 416–423, Vancouver, BC, Canada, 2001.

[34] J.-B. Huang, A. Singh, and N. Ahuja, "Single image superresolution from transformed self-exemplars," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5197–5206, Boston Massachusetts, USA, 2015.

[35] Y. Matsui, K. Ito, Y. Aramaki et al., "Sketch-based manga retrieval using manga109 dataset," *Multimedia Tools and Applications*, vol. 76, no. 20, pp. 21811–21838, 2017.

[36] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[37] C. Dong, C. C. Loy, and X. Tang, "Accelerating the superresolution convolutional neural network," in *European Conference on Computer Vision*, pp. 391–407, Springer, 2016.

[38] J. Kim, J. Kwon Lee, and K. Mu Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1637–1645, Las Vegas Nevada, USA, 2016.

[39] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep Laplacian pyramid networks for fast and accurate super-resolution," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 624–632, Honolulu Hawaii, USA, 2017.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 7398387, 13 pages
https://doi.org/10.1155/2022/7398387

Research Article

Dynamics of 2SIH2R Rumor-Spreading Model in a Heterogeneous Network

Yan Wang[1,2] **Feng Qing**[1,2] **and Ming Yan**[3]

[1]*School of Data Science and Intelligent Media, Communication University of China, Beijing 100024, China*
[2]*State Key Laboratory of Media Convergence and Communication, Communication University of China, Beijing 100024, China*
[3]*School of Information and Communications Engineering, Communication University of China, Beijing 100024, China*

Correspondence should be addressed to Ming Yan; yanm@cuc.edu.cn

Received 10 July 2021; Revised 7 October 2021; Accepted 15 December 2021; Published 17 January 2022

Academic Editor: Enrico M. Vitucci

Copyright © 2022 Yan 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.

Due to the development of social media, the threshold for information dissemination has become lower than ever before. As a special kind of information, rumors are usually harmful and are usually accompanied by a high degree of ambiguity that makes them difficult to immediately identify, but "rumors stop at wise men." When someone identifies a rumor as false and begins spreading the truth instead, a confrontational relationship obtains between the rumor and the truth that leads to the stifling of the former. Given this, we developed a 2SIH2R model in this study that contains mechanisms of discernment and confrontation in a heterogeneous network to examine the dissemination of the rumor and the truth. By using mean-field equations of the 2SIH2R model, the threshold of the spreading of each can be determined separately in three cases. The results of a numerical simulation show that under the same conditions, the greater is the mechanism of discernment or confrontation, the smaller is the instantaneous maximum influence and the final range of influence of the rumor. It can be also concluded that the earlier release of the truth about the event by the government can significantly control the rumor. Secondly, it is more effective to publish the truth in advance than after the rumor has appeared. Thirdly, it is more important for the government to increase education and improve the ability of citizens to reveal the rumor than to increase the spread of the truth after the rumor occurs. These results can be used to help reduce the harmful effects of rumors.

1. Introduction

With the advancement of the Internet and social network services (SNS) such as WeChat and Facebook, the speed, efficiency, and range of information dissemination have significantly improved compared with the era of word of mouth. As a special kind of information, rumors usually negatively affect people's psychology and behavior, may damage the government's credibility, cause social shocks, endanger public safety, and harm public interests [1–3]. The impact of rumors is especially significant nowadays because they can be spread more quickly and widely through SNS. Therefore, it is necessary to study the phenomenon of spreading rumors.

Rumors are special information that has not been officially confirmed, and that might be eventually found as true or false [4, 5]. But in this research, we assume that the rumors are false. In 1947, Allport and Postman [6] have proposed a quantitative formula to describe the spread of rumors:rumors = $i \times a$ (where i and a represent the importance of the given information and the degree of unknowability of the relevant event). This formula suggests that when a rumor has a certain importance—that is, i is not equal to zero—the higher the degree of indiscernibility of the rumor is, the greater is its impact. Rumors that can be disseminated have a higher degree of indiscernibility so that they cannot be immediately identified as such. Although rumors are confusing and have a large impact over a short period of time, an ancient Chinese idiom holds that "rumors stop at wise men." When someone identifies a rumor and may begin spreading the truth, a confrontational relationship obtains between the rumor and the truth [7] that leads to the former being stifled. Therefore, establishing a model based on this idea will better simulate the spreading process

of rumors and study the influence of various factors on this process. Through this model, we can study how to reduce the spread of rumors and expand the spread of truth; how does the time interval between the appearance of the truth and the rumors affect the entire spreading process, etc. so that the government can take corresponding measures to reduce the possible impact of rumors.

In light of the apparent similarities between the transmission of epidemic diseases and the spread of rumors, models for the spread of the former are widely used to represent the dissemination of the latter [8]. Daley and Kendall [9, 10] proposed the first rumor spreading model, called the DK model, that divides a given population into three categories (the people who have not the heard rumor; the people who actively spreading rumor; the people who no longer spreading rumor). The three categories correspond to "susceptible," "infective," and "recovered" in the SIR epidemic model, respectively. In the DK model, when two individuals in different groups come into contact, they transfer their identities with a certain probability. The most significant shortcoming of the DK model is that it ignores the topological structures of social networks. Thirty-five years after it was proposed, this problem was solved with the development of complex network theory. In 2001, Zanette [11, 12] first applied complex network theory to research on rumor spreading in a small-world network proposed by Watts and Strogatz [13] in 1998. For a more general scenario, Moreno et al. [14] established a rumor-spreading model in 2004 and analyzed the results of a simulation of a scale-free network proposed by Barabási and Albert [15]. On the basis of this work, a number of scholars have proposed adjustments to the classic model. Wang et al. [16] proposed that the ignorant (people who have never heard of the rumor) in the SIRaRu model will refuse to believe the rumor with a certain probability after hearing it. In the same year, Wang et al. [17] considered the existence of two rumors in the same network and proposed the 2SI2R model, and Zhao et al. [18, 19] proposed the SIHR model with mechanisms for forgetting and remembering in homogeneous and heterogeneous networks, respectively.

There are many reasons and factors for why rumors can spread widely [20], including their degree of indiscernibility and the behavior of the individual. Huo and Cheng [21] introduced the degree of indiscernibility of the rumor to the IWSR model and quantitatively described it. In this model, the author describes indiscernible degree of events as a parameter m and then uses the function $f(m)$ to describe transfer rate that ignorant accept and spread rumor. Xia et al. [22] proposed the SEIR model for homogeneous and heterogeneous networks with a hesitation mechanism. In this model, Xia defines the degree of fuzziness of the rumor as the parameter m which conforms to the human mind that people like to explore the truth. The results of their model showed that reducing the fuzziness of events can reduce the impact of rumors. It has been shown that improving the behavior of individuals can make them better reveal a rumor and reduce its impact [23, 24]. Considering mechanisms for the loss of interest and immunity, Huo et al. [25] proposed the SIbInIu model that divides a given population into four categories and assumed that believers spontaneously lose interest of the rumor with probability ω. It is concluded that the loss of the mechanisms of interest and immunity have a negative impact on the scale of the final spreading of rumors. In social networks, different people may have different views of a rumor. Zan et al. [7] introduced mechanisms for a counterattack and self-resistance to the SICR model, which means that "counterattack" individuals will persuade "infective" people (people who spread the rumor) to stop spreading the rumor. The degree of indiscernibility of rumors and the behavior of individuals play an important role in the process of their spreading.

Some scholars consider the impact of the antirumor on the spread of the rumor. Unlike most researches on the spreading process of the rumor using epidemic models, Askarizadeh [26] proposed a method based on game theory to study the spreading process of the rumor. This model considers a rumor control mechanism, which is to send antirumor information through the rumor control center, and studies the factors that affect people's decision-making, including social anxiety, people's attitude toward rumor/antirumor, strength of rumor/antirumor, influence of rumor control centers, and participation of people in discussions. Through simulation, it is concluded that a persuasive antirumor and rumor control center can significantly affect the spread of the rumor. Zhang [27] improved the SIR model, taking into account 8 factors such as information attraction, objective identification of rumors, subjective identification of people, the degree of trust of information media, spread probability, reinforcement coefficient, block value, and expert effects. In this paper, the people are divided into 8 categories, and the influence of various factors on the spread of the rumor is studied by means of simulation. It is concluded that higher subjective and objective identification will prevent rumor spreading. On the basis of the SIR model, Ji [28] considered the two spreading mechanisms of the time when the antirumor joined the network and studied the spreading process of the antirumor on complex networks. Through the simulation method, the relevant conclusions about the time threshold for the antirumor to join the network are drawn: the time threshold is a quantity related to the network structure; it decreases with the increase of the average degree until it decreases to 0; the earlier the joining time, the rumor can be controlled as soon as possible; when the antirumor exceeds the time threshold, the time effect will not be able to control the rumor. Jiang [29] separately modeled the two processes of rumor generation and the addition of "anti-rumor" information. Through the parameter sensitivity analysis, the critical point of whether to join the antirumor intervention is obtained; through the sensitivity analysis of the joining time, it is concluded that after the rumors appear, the officials will take actions as soon as possible to reduce the loss and impact caused by the rumor. The above several articles have considered the role of "antirumors," and all adopt simulation methods to conduct steady-state analysis of their respective models and draw relevant conclusions, but the focus is different. Askarizadeh uses the idea of game theory to build the model, while the other three articles are model improvements based on the

infectious disease model. Comparing the Ji and Jiang, the others focus more on the research on the factors that affect the spread of rumors, so as to find out how the factors affect the spread of rumors, while Ji and Jiang more focused on researching the time when "anti-rumors" joined the network.

Due to the confusing nature of rumors and the differences in individual behaviors, rumors can be widely spread in social networks. The abovementioned literature more or less considered the above two elements and established a spreading dynamic model to study the phenomenon of rumor spreading, but most articles did not consider these two elements at the same time. Secondly, only one type of spreader has been considered (the person who spreads the rumor) in most spreading models. In practice, when a major rumor is spread, it is identified by authoritative media or the government, which then spreads the truth. Thirdly, in prevalent research is that a number of models have low versatility on the model topology, which means that they are rendered inapplicable such that they cannot be adjusted if certain conditions change. And most scholars believe that after interacting with rumor spreader, the ignorant will turn into spreader or exposed (SEIR) with series topology in SIR spreading model or SEIR spreading model separately.

Therefore, we use past research to build a 2SIH2R (spreader1-spreader2-ignorant-hesitant1-stiflers1-stiflers2) model with parallel topology, discernibility mechanism, and confrontation mechanism in a heterogeneous network. People in this model are divided into six categories and two kinds of spreaders are simultaneously considered (people who spread the rumor and those who spread the truth). This 2SIH2R model can be converted into the traditional SIR or the SEIR model by changing the relevant parameters.

To sum up, in theory, the 2SIH2R model is compatible with the basic assumptions of the traditional model and incorporates more realistic elements. From a perspective, our model is more convincing. For rumor control in real life, this model can quantitatively study how the discernible and confrontation mechanisms affect the spread of rumors. So, the government can take corresponding realistic measures to adjust the discernible and confrontation mechanisms to achieve the goal of information governance.

The remainder of this paper is organized as follows: In Section 2, we define the 2SIH2R model and establish its mean-field equations. In Section 3, a steady-state analysis and the spreading threshold are discussed in three cases. Section 4 details a numerical simulation to test the proposed 2SIH2R model on a heterogeneous network, and Section 5 offers a discussion of the results and summarizes the conclusions of this study.

2. 2SIH2R Rumor-Spreading Model

In this section, we assume that there are two kinds of information in the entire population, called rumor and truth, and that people in the heterogeneous network can be divided into six categories: spreader1 (S_1), spreader2 (S_2), ignorant (I), hesitant1 (H), stiflers1 (R_1), and stiflers2 (R_2). Of them, spreader1 and spreader2 represent people who spread the rumor and the truth, respectively. The ignorant are people who have never heard of the rumor or the truth. The term "hesitant1" represents people who have heard the rumor, but do not spread it temporarily, and stiflers1 and stiflers2, respectively, represent people who have heard the rumor and the truth but do not spread them. Referring to the previous research [30], the rumor-spreading process of the 2SIH2R is shown in Figure 1.

In Figure 1, the solid/dashed line originating from the "ignorant" shows that the ignorant comes into contact with spreader1/spreader2, and the rules of rumor spreading of the 2SIH2R model can be summarized as follows:

(1) From the perspective of equation rumors = $i \times a$ and reality, considering that different rumor events have different confusion, which takes impact to the people is also different, so, we define m as the rate of discernibility of the rumor and $f(m)$ as the ability of people to reveal it. The function of f is to map the characteristics of the rumor to those of the people. We assume that there is a positive correlation between m and $f(m)$. The greater m is, the more likely the rumor is to be revealed, and the greater the value of $f(m)$ is, the more likely people are to not immediately believe the rumor

(2) When an ignorant person comes into contact with a spreader1, there are three possible scenarios: (i) the ignorant is likely to believe the rumor and spread it with probability $(1 - f(m))\lambda_1$, where λ_1 is the rate of spread of the rumor. (ii) The ignorant does not believe the rumor immediately and hesitates to spread it with a probability $f(m)\eta$, where η is the potential rate of spreading of the rumor. (iii) The ignorant makes no response to the rumor owing to a lack of interest in it and continues being unaware of it

(3) When an ignorant comes into contact with a spreader2, there are two possible scenarios: (i) The ignorant is likely to believe the truth and spread it with a probability λ_2, called the truth-spreading rate. (ii) The ignorant may make no response to the truth owing to a lack of interest in the truth and continue to be unaware. Because the truth is generally released by an authority or government, it is relatively more objective, accurate, and clear. This makes it easier for the ignorant to judge it, which in turn renders them less likely to be hesitant to spread it

(4) People regarded as hesitant1 have a desire to spread information. Based on suspicion of the rumor and the influence of external circumstances, they do not spread it over a short period. In the process of hesitating, hesitant1 may choose to believe and spread the rumor, or they may find out and spread the truth. Hence, we suppose that in each step, hesitant1 spontaneously transforms into a spreader of information with a certain probability, that is, becomes spreader1 or spreader2 with probability θ_1 or θ_2, respectively

(5) When a spreader1 (spreader2) comes into contact with another spreader1 (spreader2), the exchange of information leads them to believe that the rumor (truth) has been widely known. Thus, spreader1

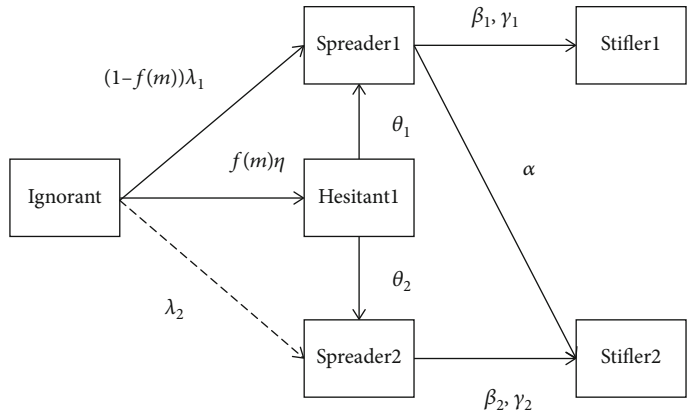


Figure 1: 2SIH2R rumor-spreading process.

(spreader2) loses enthusiasm for continually spreading the rumor (truth) and becomes a stifler1 (stifler2), with a rate of loss of interest of β_1 (β_2).

(6) Considering the information forgetting mechanism, in each step, spreader1 (spreader2) spontaneously transforms into stifler1 (stifler2) with probability γ_1 (γ_2), which means the rumor- (truth-) forgetting rate. This is because people may forget the rumor after a certain period

(7) Considering the mechanism of confrontation between the truth and the rumor, when a spreader1 comes into contact with a spreader2, the former believes the truth rather than the rumor with probability α, which is the called the rate of confrontation

In a heterogeneous network, $I_k(t)$, $S_{1k}(t)$, $S_{2k}(t)$, $H_k(t)$, $R_{1k}(t)$, and $R_{2k}(t)$ represent the densities of the ignorant, spreader1, spreader2, hesitant1, stifler1, and stifler2 with degree k, respectively, at time t. Based on the above rules of spreading of the 2SIH2R model, the mean-field equations can be given as follows:

$$\frac{dI_k(t)}{dt} = -[\lambda_1(1-f(m)) + f(m)\eta]kI_k(t)\sum_{k'} S_{1k'}(t)P\left(k'/k\right) - \lambda_2 kI_k(t)\sum_{k'} S_{2K'}(t)P\left(k'/k\right), \tag{1}$$

$$\frac{dS_{1k}(t)}{dt} = \lambda_1(1-f(m))kI_k(t)\sum_{k'} S_{1k'}(t)P\left(k'/k\right) - \alpha kS_{1k}(t)\sum_{k'} S_{2k'}(t)P\left(k'/k\right) - \beta_1 kS_{1k}(t)\sum_{k'}(S_{1k'}(t) + R_{1k'}(t) + H_{k'}(t))P\left(k'/k\right) - \gamma_1 S_{1k}(t) + \theta_1 H_k(t), \tag{2}$$

$$\frac{dS_{2k}(t)}{dt} = \lambda_2 kI_k(t)\sum_{k'} S_{2k'}(T)P\left(k'/k\right) - \beta_2 kS_{2k}(t)\sum_{k'}(S_{2k'}(t) + R_{2k'}(t))P\left(k'/k\right) - \gamma_2 S_{2k}(t) + \theta_2 H_k(t), \tag{3}$$

$$\frac{dH_k(t)}{dt} = f(m)\eta kI_k(t)\sum_{k'} S_{1k'}(t)P\left(k'/k\right) - (\theta_1+\theta_2)H_k(t), \tag{4}$$

$$\frac{dR_{1k}(t)}{dt} = \beta_1 kS_{1k}(t)\sum_{k'}(S_{1k'}(t) + R_{1k'}(t) + H_{k'}(t))P\left(k'/k\right) + \gamma_1 S_{1k}(t), \tag{5}$$

$$\frac{dR_{2k}(t)}{dt} = \alpha kS_{1k}(t)\sum_{k'} S_{2k'}(T)P\left(k'/k\right) + \beta_2 kS_{2k}(t)\sum_{k'}(S_{2k'}(t) + R_{2k'}(t))P\left(k'/k\right) + \gamma_2 S_{2k}(t), \tag{6}$$

where $I_k = N_{I,K}/N_k$ represents the number of the ignorant with degree k divided by the total number of people with degree k, $P(k'/k)$ is the probability that a node with degree k is linked to one with degree k', and $\sum_{k'} S_{1k'}(t)P(k'/k)$ is the probability that an edge of a node with degree k links to a spreader1 node at time t.

3. Steady-State Analysis

In this section, we provide the condition whereby rumors and the truth can spread successfully in a heterogeneous network. It is assumed that when the model arrives at a steady state, neither spreader1 nor spreader2 is present. We can thus obtain the conclusions$S_1 = \lim_{t\longrightarrow\infty} S_1(t) = 0$, $S_2 = \lim_{t\longrightarrow\infty} S_2(t) = 0$, $H = \lim_{t\longrightarrow\infty} H(t) = 0$, and $\lim_{t\longrightarrow\infty}(I(t) + R_1(t) + R_2(t)) = 1$. Moreover, the normalization condition is $I(t) + S_1(t) + S_2(t) + H(t) + R_1(t) + R_2(t) = 1$. The degree–degree correlations can be described as $P(k'/k) = k'P(k')/<k> = q(k')$, where

$P(k')$ is the degree distribution function. Without loss of generality, we set the density of the ignorant to $I_k(0) = I(0) \approx 1$. We now study the spreading threshold of the model in three cases.

3.1. Steady-State Analysis of Rumor. In the early stages of its process of spreading, few people know the rumor and no one knows the truth. Thus, in this case, without loss of generality, we assume that there is only one spreader1 who spreads the rumor, and there is no truth. It is assumed that there are N nodes in the network. Based on the above, the initial condition can be given as follows: $S_2(0) = 0$, $S_1(0) = 1/N \approx 0$, $I(0) = N - 1/N \approx 1$, and $H(0) = R_1(0) = R_2(0) = 0$, and the model can be adjusted as follows:

$$\frac{dI_k(t)}{dt} = -[\lambda_1(1 - f(m)) + f(m)\eta]kI_k(t)\sum_{k'} S_{1k'}(t)q\left(k'\right), \tag{7}$$

$$\begin{aligned}\frac{dS_{1k}(t)}{dt} &= \lambda_1(1 - f(m))kI_k(t)\sum_{k'} S_{1k'}(t)q\left(k'\right) \\ &\quad - \beta_1 kS_{1k}(t)\sum_{k'}(S_{1k'}(t) + R_{1k'}(t) + H_{k'}(t))q\left(k'\right) \\ &\quad - \gamma_1 S_{1k}(t) + \theta_1 H_k(t),\end{aligned} \tag{8}$$

$$\frac{dH_k(t)}{dt} = f(m)\eta kI_k(t)\sum_{k'} S_{1k'}(t)q\left(k'\right) - \theta_1 H_k(t), \tag{9}$$

$$\frac{dR_{1k}(t)}{dt} = \beta_1 kS_{1k}(t)\sum_{k'}(S_{1k'}(t) + R_{1k'}(t) + H_{k'}(t))q\left(k'\right) + \gamma_1 S_{1k}(t). \tag{10}$$

In this case, we can integrate Eq. (7) directly and get

$$I_k(t) = e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi(t)}, \tag{11}$$

where $\varphi(t)$ can be given as

$$\varphi(t) = \int_0^t \sum_{k'} S_{1k'}\left(t'\right)q\left(k'\right)dt' = \int_0^t << S_{1k'}\left(t'\right) >> dt', \tag{12}$$

and the shorthand notation $<<O(k) >> = \sum_k O(k)q(k)$ has been used.

Multiplying Eq. (8) with $q(k)$, summing over k, and integrating the equation with respect to t, the following derivation can be obtained:

$$\begin{aligned}\frac{d\varphi(t)}{dt} &= \lambda_1(1 - f(m))\sum_k q(k)\int_0^t ke^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi\left(t'\right)} \\ &\quad << S_{1k'}\left(t'\right) >> dt' - \beta_1\int_0^t \sum_k kS_{1k}\left(t'\right)q(k)\sum_{k'}\left(S_{1k'}\left(t'\right)\right. \\ &\quad \left. + R_{1k'}\left(t'\right) + H_{k'}\left(t'\right)\right)q\left(k'\right)dt' \\ &\quad - \gamma_1\int_0^t \sum_k S_{1k}\left(t'\right)q(k)dt' + \theta_1\int_0^t \sum_k H_k\left(t'\right)q(k)dt' \\ &= \frac{\lambda_1(1 - f(m))}{\lambda_1(1 - f(m)) + f(m)\eta}\left[1 - << e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi(t)} >>\right] \\ &\quad - \gamma_1\varphi(t) + \theta_1\psi(t) - \beta_1\int_0^t \\ &\quad << kS_{1k}\left(t'\right) >> \left[1 - << e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi\left(t'\right)} >>\right]dt',\end{aligned} \tag{13}$$

where $\psi(t) = \int_0^t << H_k(t') >> dt'$.

Similarly, by multiplying Eq. (9) with $q(k)$, summing over k, and integrating the equation, we can get

$$\frac{d\psi(t)}{dt} = \frac{f(m)\eta}{\lambda_1(1 - f(m)) + f(m)\eta}\left[1 - << e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi(t)} >>\right] - \theta_1\psi(t). \tag{14}$$

When $t \longrightarrow \infty$, $d\varphi/dt = 0$, and $d\psi/dt = 0$, the following derivation can be obtained:

$$\begin{aligned}0 &= \frac{\lambda_1(1 - f(m))}{\lambda_1(1 - f(m)) + f(m)\eta}\left[1 - << e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi_\infty} >>\right] \\ &\quad - \gamma_1\varphi_\infty + \theta_1\psi(\infty) - \beta_1\int_0^\infty << kS_{1k}\left(t'\right) >> \\ &\quad \cdot \left[1 - << e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi\left(t'\right)} >>\right]dt',\end{aligned} \tag{15}$$

$$\begin{aligned}0 &= \frac{f(m)\eta}{\lambda_1(1 - f(m)) + f(m)\eta}\left[1 - << e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi_\infty} >>\right] \\ &\quad - \theta_1\psi(\infty),\end{aligned} \tag{16}$$

where $\varphi_\infty = \varphi(\infty)$.

Combining Eqs. (15) and (16),

$$\begin{aligned}0 &= \left[1 - << e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi_\infty} >>\right] \\ &\quad - \beta_1\int_0^\infty << kS_{1k}\left(t'\right) >> \left[1 - << e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi\left(t'\right)} >>\right]dt' \\ &\quad - \gamma_1\varphi_\infty.\end{aligned} \tag{17}$$

By integrating Eqs. (8) and (9), we have

$$S_{1k}(t) = \frac{\lambda_1(1-f(m))}{\lambda_1(1-f(m))+f(m)\eta}\left(1-e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi(t)}\right) - \gamma_1\int_0^t S_{1k}\left(t'\right)dt' + \theta_1\int_0^t H_k\left(t'\right)dt' - \beta_1 k\int_0^t S_{1k}\left(t'\right)\left(1-<<e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi(t)}>>\right)dt', \tag{18}$$

$$H_k(t) = \frac{f(m)\eta}{\lambda_1(1-f(m))+f(m)\eta}\left(1-e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi(t)}\right) - \theta_1\int_0^t H_k\left(t'\right)dt'. \tag{19}$$

By adding Eq. (18) to Eq. (19), we have

$$S_{1k}(t) = 1-e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi(t)} - H_k(t) - \gamma_1\int_0^t S_{1k}\left(t'\right)dt' - \beta_1 k\int_0^t S_{1k}\left(t'\right)\cdot\left(1-\ll e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi(t)}\gg\right)dt'. \tag{20}$$

By solving Eq. (19), we can obtain

$$H_k(t) = \frac{f(m)\eta}{\lambda_1(1-f(m))+f(m)\eta}\cdot\left[1-e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi(t)} - \theta_1\int_0^t\left(1-e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi\left(t'\right)}\right)e^{\theta_1\left(t'-t\right)}dt'\right]. \tag{21}$$

By inserting Eq. (21) into Eq. (20), we can get

$$S_{1k}(t) = \frac{\lambda_1(1-f(m))}{\lambda_1(1-f(m))+f(m)\eta}\left(1-e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi(t)}\right) + O(\beta_1) - \gamma_1\int_0^t\left[\frac{\lambda_1(1-f(m))}{\lambda_1(1-f(m))+f(m)\eta}\cdot\left(1-e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi\left(t'\right)}\right)e^{\gamma_1\left(t'-t\right)}dt' - \gamma_1\int_0^t\theta_1\left[\int_0^{t'}\frac{f(m)\eta}{\lambda_1(1-f(m))+f(m)\eta}\cdot\left(1-e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi\left(t''\right)}\right)e^{\theta_1\left(t''-t'\right)}dt''\right]e^{\gamma_1\left(t'-t\right)}dt'. + \frac{f(m)\eta}{\lambda_1(1-f(m))+f(m)\eta}\theta_1\int_0^t\cdot\left(1-e^{-(\lambda_1(1-f(m))+f(m)\eta)k\varphi\left(t'\right)}\right)e^{\theta_1\left(t'-t\right)}dt' \tag{22}$$

Both $\varphi(t)$ and φ_∞ are very small when the size of the rumor tends to be stable. There is thus a finite function $g(t)$ that satisfies $\varphi(t) = \varphi_\infty g(t)$, and Eq. (22) then becomes

$$S_{1k}(t) = k\varphi_\infty\left\{\lambda_1(1-f(m))g(t) - \gamma_1\lambda_1(1-f(m))\int_0^t g\left(t'\right)e^{\gamma_1\left(t'-t\right)}dt' + O(\beta_1) - \gamma_1\theta_1 f(m)\eta\int_0^t e^{\gamma_1\left(t'-t\right)}\int_0^{t'} g\left(t''\right)e^{\theta_1\left(t''-t'\right)}dt''dt' + f(m)\eta\theta_1\int_0^t g\left(t'\right)e^{\theta_1\left(t'-t\right)}dt' + O\left(\varphi_\infty^2\right)\right\}. \tag{23}$$

Inserting Eq. (23) into Eq. (17),

$$\begin{aligned}0 = 1-<<1-(\lambda_1(1-f(m))+f(m)\eta)k\varphi_\infty &+ \frac{[(\lambda_1(1-f(m))+f(m)\eta)k]^2\varphi_\infty^2}{2} + O\left(\varphi_\infty^3\right)>> \\ -\beta_1\int_0^\infty <<k\varphi_\infty k\Big\{&\lambda_1(1-f(m))g\left(t'\right) \\ &-\gamma_1\lambda_1(1-f(m))\int_0^{t'} g\left(t''\right)e^{\gamma_1\left(t''-t'\right)}dt'' \\ &+f(m)\eta\theta_1\int_0^{t'} g\left(t''\right)e^{\theta_1\left(t''-t'\right)}dt'' \\ &-\gamma_1\theta_1 f(m)\eta\int_0^{t'} e^{\gamma_1\left(t''-t'\right)}\int_0^{t''} g\left(t'''\right)e^{\theta_1\left(t'''-t''\right)}dt'''dt''\Big\} \\ &+O\left(\varphi_\infty^2\right)+O(\beta_1)>> \\ \times\left[1-<<1-(\lambda_1(1-f(m))+f(m)\eta)k\varphi_\infty g\left(t'\right)+O\left(\varphi_\infty^2\right)>>\right]&dt' \\ -\gamma_1\varphi_\infty,&\end{aligned} \tag{24}$$

$$0 = \varphi_\infty\{[\lambda_1(1-f(m))+f(m)\eta]<<k>> - \frac{(\lambda_1(1-f(m))+f(m)\eta)^2}{2}<<k^2>>\varphi_\infty - \beta_1<<k^2>>I(\lambda_1(1-f(m))+f(m)\eta)<<k>>\varphi_\infty - \gamma_1\} + O\left(\varphi_\infty^3\right) + O\left(\beta_1^2\right). \tag{25}$$

where $I = \int_0^\infty\{\lambda_1(1-f(m))g(t') - \gamma_1\lambda_1(1-f(m))\int_0^{t'} g(t'')e^{\gamma_1(t''-t')}dt'' + f(m)\eta\theta_1\int_0^{t'} g(t'')e^{\theta_1(t''-t')}dt'' - \gamma_1\theta_1 f(m)\eta\int_0^{t'} e^{\gamma_1(t''-t')}\int_0^{t'} g(t''')e^{\theta_1(t'''-t'')}dt'''dt''\}g(t')dt'$, which is a finite and positive-definite integral.

From Eq. (25), we have

$$0 = \varphi_\infty\{[\lambda_1(1-f(m))+f(m)\eta]<<k>> - \frac{(\lambda_1(1-f(m))+f(m)\eta)^2}{2}<<k^2>>\varphi_\infty - \beta_1<<k^2>>><<k>>I(\lambda_1(1-f(m))+f(m)\eta)\varphi_\infty - \gamma_1\}. \tag{26}$$

$\varphi_\infty = 0$ is one solution to Eq. (26), and the nontrivial

solution is obtained as follows:

$$\varphi_{\infty} = \frac{[\lambda_1(1-f(m))+f(m)\eta]<<k>>-\gamma_1}{<<k^2>>[\lambda_1(1-f(m))+f(m)\eta]\{1/2[\lambda_1(1-f(m))+f(m)\eta]+\beta_1<<k>>I\}}. \tag{27}$$

Note that $\ll k \gg = <k^2>/<k>$; so, $[\lambda_1(1-f(m))+f(m)\eta](<k^2>/<k>)-\gamma_1 \geq 0$, that is, $\lambda_1 \geq <k>\gamma_1/<k^2> -f(m)\eta/(1-f(m))$, which yields a positive value for φ_{∞}. This means that the spreading threshold in this case is

$$\lambda_{1c} = \frac{(<k>/<k^2>)\gamma_1 - f(m)\eta}{(1-f(m))}. \tag{28}$$

3.2. Steady-State Analysis of Truth. In this case, we consider the truth of some well-known events has been already spread by the government or authoritative media. In this period of time, there are no rumors, but only the truth. Based on the above, the initial condition can be given as follows: $S_1(0) = 0$, $S_2(0) = 1/N \approx 0$, $I(0) = N - 1/N \approx 1$, and $H(0) = R_1(0) = R_2(0) = 0$. We can prove that in this case, the model can be adjusted as follows:

$$\begin{aligned}
\frac{dI_k(t)}{dt} &= -\lambda_2 k I_k(t)\sum_{k'} S_{2k'}(t)P\left(k'/k\right),\\
\frac{dS_{2k}(t)}{dt} &= \lambda_2 k I_k(t)\sum_{k'} S_{2k'}(t)q\left(k'\right) - \beta_2 k S_{2k}(t)\sum_{k'}(S_{2k'}(t)\\
&\quad + R_{2k'}(t))q\left(k'\right) - \gamma_2 S_{2k}(t),\\
\frac{dR_{2k}(t)}{dt} &= \beta_2 k S_{2k}(t)\sum_{k'}(S_{2k'}(t)+R_{2k'}(t))P\left(k'/k\right) + \gamma_2 S_2(t).
\end{aligned} \tag{29}$$

We follow the proof procedure in the previous section to get

$$\lambda_{2c} = \frac{<k>}{<k^2>}\gamma_2. \tag{30}$$

3.3. Steady-State Analysis of the 2SIH2R Model. In this case, we consider a more general situation where the rumor and the truth exist in the network at the same time. Without loss of generality, we assume that there is one spreader1 and one spreader2 at time $t = 0$, and the initial condition can then be given as follows: $I(0) = N - 2/N \approx 1$, $S_1(0) = S_2(0) = 1/N \approx 0$, and $H(0) = R_1(0) = R_2(0) = 0$. Without loss of generality, it is assumed that $I_k(0) = 1$, $S_{1k}(0) = S_{2k}(0) \approx 0$, and $H_k(0) = R_{1k}(0) = R_{2k}(0) = 0$.

For $I_k = N_{I,K}/N_k$ and $I_k(t) + S_{1k}(t) + S_{2k}(t) + H_k(t) + R_{1k}(t) + R_{2k}(t) = 1$, i.e., if the number of five types of people is fixed, there is only one corresponding value of the remaining one. So, we will discuss the system (1)-(5). Obviously, the system (1)-(5) has a disease-free equilibrium $E_0 = (I_k(0), S_{1k}(0), S_{2k}(0), H_k(0), R_{1k}(0), R_{2k}(0)) = (1, 0, 0, 0, 0, 0)$. The above Jacobin matrix at E_0 can be written as

$$J(E_0) = \begin{bmatrix} A_{11} & A_{12} & \cdots & A_{1n} \\ A_{21} & A_{22} & \cdots & A_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ A_{n1} & A_{n2} & \cdots & A_{nn} \end{bmatrix}, \text{ where}$$

$$A_{ij} = \begin{pmatrix} 0 & -[\lambda_1(1-f(m))+f(m)\eta]jq(j) & -\lambda_2 jq(j) & 0 & 0 \\ 0 & \lambda_1(1-f(m))jq(j)-\gamma_1 & 0 & \theta_1 & 0 \\ 0 & 0 & \lambda_2 jq(j)-\gamma_2 & \theta_2 & 0 \\ 0 & f(m)\eta jq(j) & 0 & -(\theta_1+\theta_2) & 0 \\ 0 & \gamma_1 & 0 & 0 & 0 \end{pmatrix} (i=j),$$

$$A_{ij} = \begin{pmatrix} 0 & -[\lambda_1(1-f(m))+f(m)\eta]iq(j) & -\lambda_2 iq(j) & 0 & 0 \\ 0 & \lambda_1(1-f(m))iq(j) & 0 & 0 & 0 \\ 0 & 0 & \lambda_2 iq(j) & 0 & 0 \\ 0 & f(m)\eta iq(j) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix} (i \neq j). \tag{31}$$

By mathematical induction method, the characteristic equation can be calculated as

$$\begin{aligned}
0 &= (-\mu)^{2n}(-\gamma_2-\mu)^{n-1}\left\{\lambda_2\sum_n nq(n)-\gamma_2-\mu\right\}[-(\theta_1+\theta_2)-\mu)(-\gamma_1-\mu)]^{n-1}\\
&\times\left\{(-(\theta_1+\theta_2)-\mu)\left[\lambda_1(1-f(m))\sum_n nq(n)-\gamma_1-\mu\right]-\theta_1 f(m)\eta\sum_n nq(n)\right\}.
\end{aligned} \tag{32}$$

For $q(k) = kP(k)/<k>$, we have $\sum_k kq(k) = \sum_k k^2 P(k)/<k> = <k^2>/<k>$. So, the characteristic equation can be rewritten as

$$\begin{aligned}
0 &= (-\mu)^{2n}(-\gamma_2-\mu)^{n-1}\left\{\lambda_2\frac{<k^2>}{<k>}-\gamma_2-\mu\right\}\\
&\cdot[(-(\theta_1+\theta_2)-\mu)(-\gamma_1-\mu)]^{n-1}\{\mu^2+b\mu+c\},
\end{aligned} \tag{33}$$

where

$$\begin{aligned}
b &= -\left[\lambda_1(1-f(m))\frac{<k^2>}{<k>}-\gamma_1-(\theta_1+\theta_2)\right],\\
c &= -\left[\lambda_1(1-f(m))\frac{<k^2>}{<k>}-\gamma_1\right](\theta_1+\theta_2)-\theta_1 f(m)\eta\frac{<k^2>}{<k>}.
\end{aligned} \tag{34}$$

According to the Routh-Hurwitz criteria, the system is stable at E_0 only when all eigenvalues are not greater than 0. Thus, a sufficient condition of $(I, S_1, S_2, H, R_1, R_2)$ to be locally asymptotically stable at E_0 is $\max\{1/2(-b+\sqrt{b^2-4c}), \lambda_2(<k^2>/<k>)-\gamma_2\} \leq 0$. In other words, if rumors or truth can spread in a heterogeneous network,

the following conditions need to be satisfied:

$$\frac{1}{2}\left(-b+\sqrt{b^2-4c}\right)>0 \text{ or } \lambda_2\frac{<k^2>}{<k>}-\gamma_2>0. \quad (35)$$

Consequently,

$$\lambda_1(1-f(m))+\frac{\theta_1}{\theta_1+\theta_2}f(m)\eta>\frac{<k>}{<k^2>}\gamma_1 \text{ or } \lambda_2>\frac{<k>}{<k^2>}\gamma_2. \quad (36)$$

Thus, if the rumor and the truth can spread widely in the generated network, which is closed and heterogeneous, λ_1 and λ_2 should satisfy Eq. (36). We then set $\lambda_2=0$, $\gamma_2=0$, and $\theta_2=0$ to get the Eq. (28). By setting $\lambda_1=0$, $f(m)=0$, we can get Eq. (30). We can conclude that the third general case contains the first two special cases.

4. Numerical Simulation

In this section, we study important features of the 2SIH2R model in a heterogeneous network generated through a numerical simulation. By referring to the research [30–34], the following initial conditions can be given: $I(0)=N-2/N\approx 1$, $S_1(0)=S_2(0)=1/N\approx 0$, and $H(0)=R_1(0)=R_2(0)=0$. We executed the simulation in the Barabási-Albert scale-free network with $N=100000$, and three edges attach a new node to already existing nodes.

Figure 2 shows the general trends of the density of the six categories (spreader1, spreader2, stifler1, stifler2, ignorant, and hesitant) in the 2SIH2R model over time, with parameters $m=0.3$, $f(m)=0.7m$, $\lambda_1=\lambda_2=0.7$, $\eta=0.8$, $\theta_1=0.5$, $\theta_2=0.3$, $\beta_1=\beta_2=0.3$, $\gamma_1=\gamma_2=0.1$, and $\alpha=0.5$. Unless otherwise specified, the above parameters are used in this section. The numerical simulation showed that the number of ignorant people declined sharply until it reached the steady state. At the same time, the densities of spreader1 and spreader 2 reached a peak and then declined. When the system became stable, which means that neither the rumor nor the truth spreads, only ignorant, stifler1, and stifler2 remain in the network.

To illustrate how the mechanism affects the spread of the rumor, we plot in Figures 3–5. Figures 3 and 4 show how spreader1 and stifler1 change over time with different values of m. Figure 5 shows how the final sizes of stifler1 and stifler2 change with different values of $f(m)$. Figure 3 shows that the peak of spreader1 decreased with increasing m, which means that the stronger the mechanism was, the easier it was to reveal the rumor, and the smaller the impact of the rumor was. The same conclusion can be also obtained from Figure 4. The final size of stifler1 also decreased with increasing m. However, the impact of the rumor decreased with increasing m, but the times to peak for both spreader1 and stifler1 did not change significantly. Figure 5 shows that the final size of stifler1 decreases with increasing values of $f(m)$, but that of stifler2 increases. It can be concluded that as $m(f(m))$ increases, the short-term and final impacts of the rumor decrease while those of the truth increase.

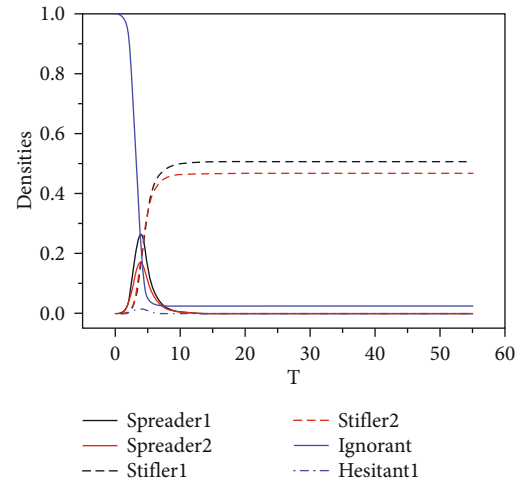


Figure 2: Density of six categories of people in the homogeneous network.

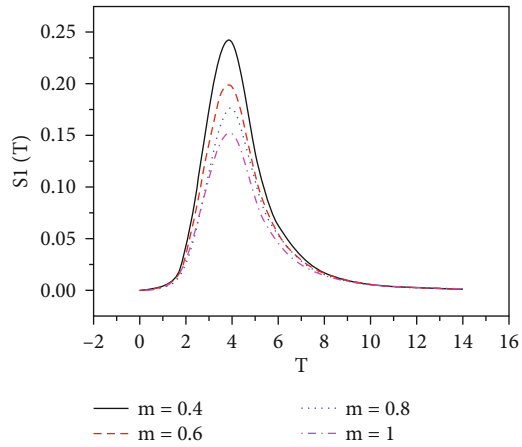


Figure 3: Density of spreader1 over time for different values of m.

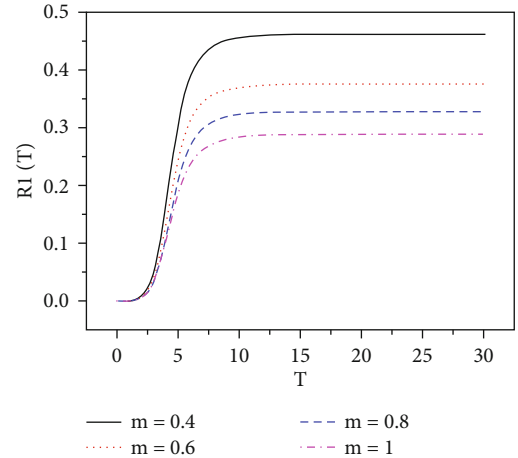


Figure 4: Density of stifler1 over time for different values of m.

To illustrate how the confrontation mechanism affected the spread of the rumor, we plot in Figures 6–8. Figures 6 and 7 describe how spreader1 and stifler1 change over time with different values of α. Figure 8 describes how the final

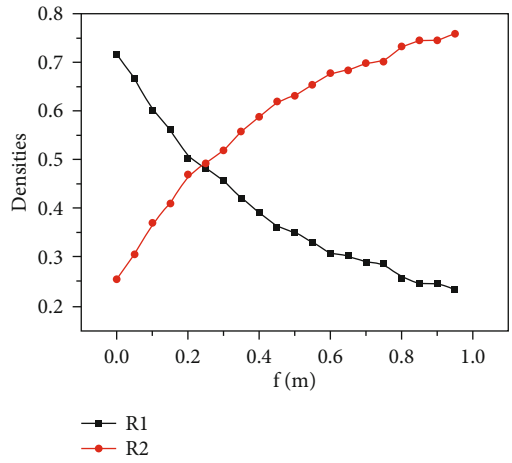


Figure 5: Densities of stifler1 and stifler2 for different values of $f(m)$ in the steady state.

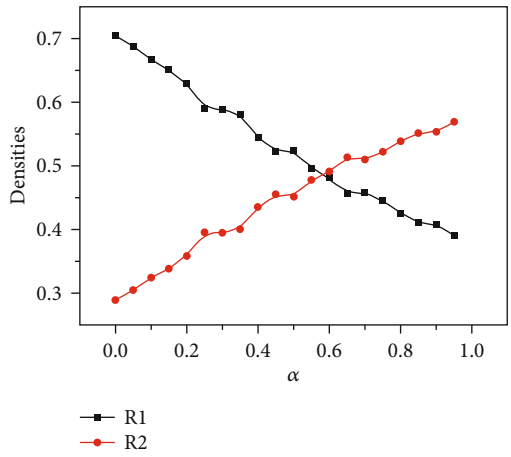


Figure 8: Densities of stifler1 and stifler2 for different values of α in the steady-state.

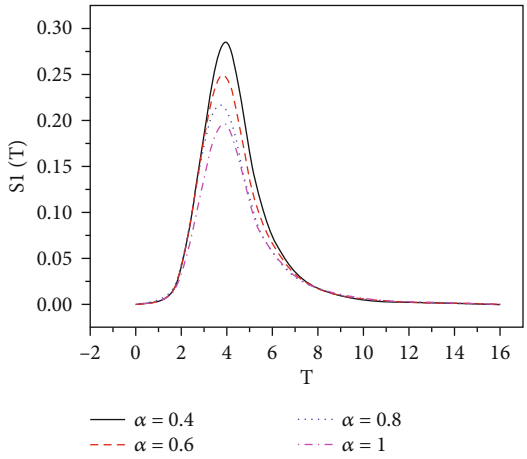


Figure 6: Density of spreader1 over time for different values of α .

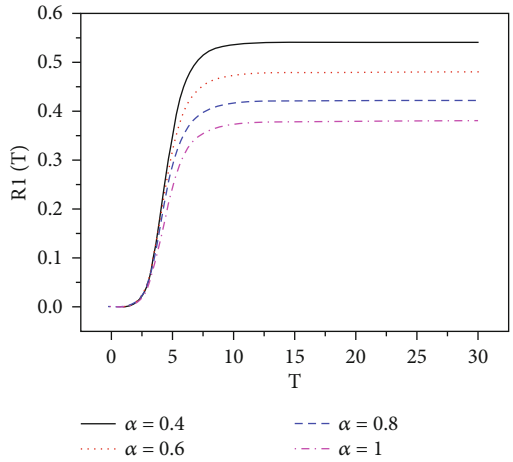


Figure 7: Density of stifler2 over time for different values of α .

sizes of stifler1 and stifler2 change with different values of α .Figure 6 shows that the peak of spreader1 decreased with increasing α, which means that the stronger the confrontation mechanism was, the easier it was to reveal the rumor, and the smaller its impact was. The same conclusion can be obtained from Figure 7. The final size of stifler1 also decreased with increasing α. However, the impact of the rumor decreased with increasing α, but the times to peak of spreader1 and stifler1 did not change significantly. Figure 8 shows that the final size of stifler1 decreases with increasing values of α, but that of stifler2 increases. It can be concluded that as α increases, the short-term and final impacts of the rumor will decrease while those of the truth increase.

To illustrate how the mechanisms and $T_{\text{truth_in}}$ affected the spread of the rumor, we plot in Figures 9 and 10. In Figures 9 and 10, the time when the truth begins to spread in the network is called $T_{\text{truth_in}}$, and when $T_{\text{truth_in}} = 0$, the rumor begins to spread. Figures 9 and 10 describe how the final size of stifler1 and stifler2 change over $T_{\text{truth_in}}$ with different values of $f(m)$ and α. From Figures 9 and 10, we can find that the earlier the truth joins the network, the influence range of the rumor will be reduced and the truth will be enlarged. As shown in Figure 9, the growth rate of R_1 and the decline rate of R_2 both decreased with increasing $T_{\text{truth_in}}$. In other words, if the truth begins to spread too late, the time factor will not significantly affect the spread of rumors. The same conclusion can also be obtained from Figure 10. Comparing Figures 9 and 10, we can find that if the truth can be spread at an earlier time (E.g., $T_{\text{truth_in}} < 5$), the influence range of the rumor will be reduced and the truth will be enlarged significantly with increasing $f(m)$ or α. But as shown in Figure 10, if the truth begins to spread too late (E.g., $T_{\text{truth_in}} > 5$), the confrontation mechanism will become ineffective while the discernibility mechanism is still effective.

To illustrate how the mechanisms and $T_{\text{rumor_in}}$ affected the spread of the rumor, we plot in Figures 11 and 12. In Figures 11 and 12, the time when the rumor begins to spread

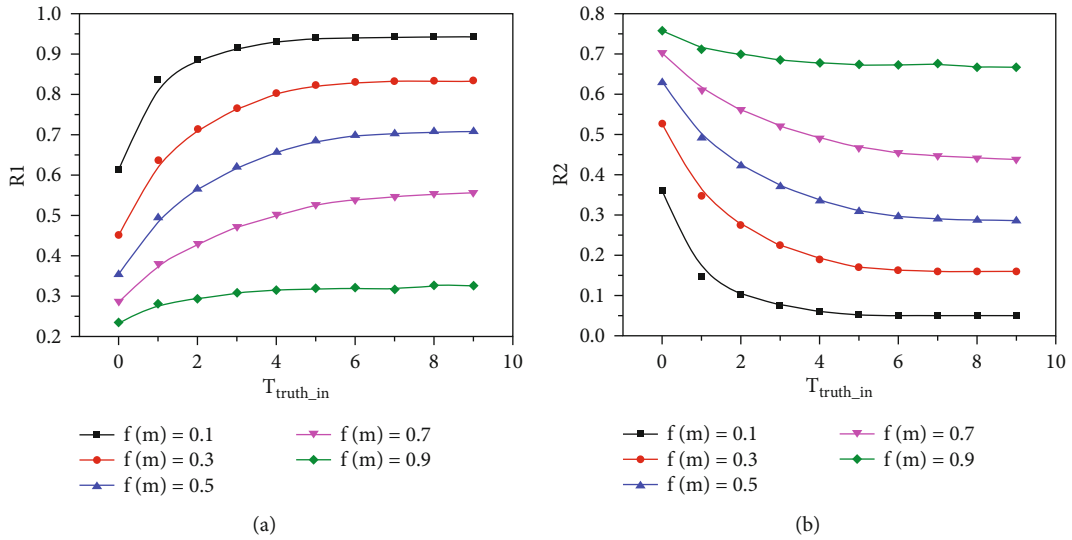


Figure 9: Densities of stifler1 and stifler2 for different values of $f(m)$ and T_{truth_in} in the steady-state.

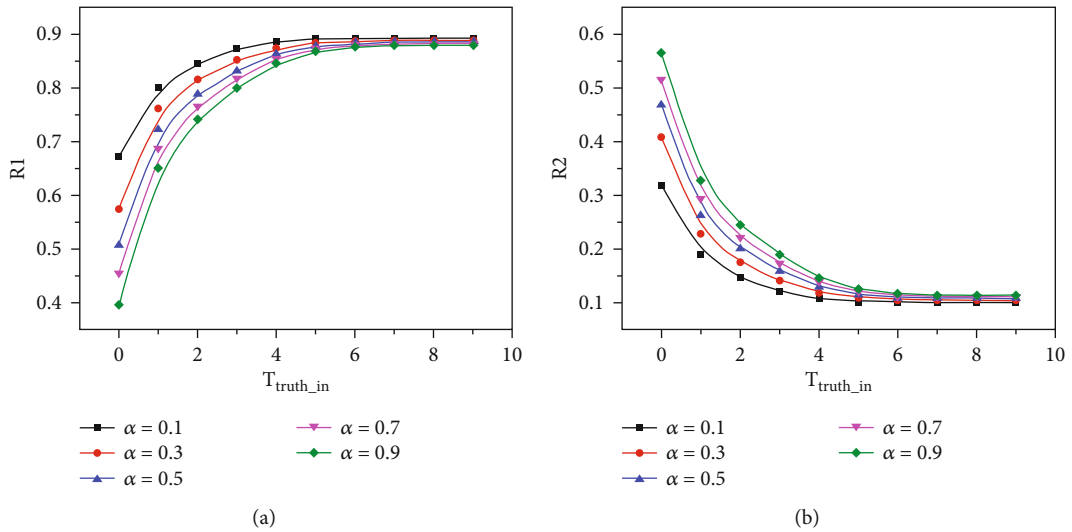


Figure 10: Densities of stifler1 and stifler2 for different values of α and T_{truth_in} in the steady-state.

in the network is called T_{rumor_in}, and when $T_{rumor_in} = 0$, the truth begins to spread. Figures 11 and 12 describe how the final size of stifler1 and stifler2 change over T_{truth_in} with different values of $f(m)$ and α. From Figures 11 and 12, we can find that the earlier the rumor joins the network, the influence range of the rumor will be enlarged, and the truth will be reduced. In addition, comparing Figures 11 and 12, we can find that if the rumor can be spread at an earlier time (E.g., $T_{rumor_in} < 2$), the influence range of the rumor will be reduced, and the truth will be enlarged significantly with increasing $f(m)$ or α. But if the rumor begins to spread too late (E.g., $T_{rumor_in} > 2$), the mechanisms of confrontation and discernibility will become ineffective, and the rumor cannot spread.

In summary, it can be concluded that the earlier release of the truth about the event by the government can significantly control the rumor. Even the truth of some well-known events has been already spread can suppress the appearance of the rumor. This also verifies the hypothesis of chapter Steady-state Analysis of Truth. Secondly, it is more effective to publish the truth in advance than after the rumor has appeared. This is because as shown in Figures 11 and 12, if the truth is spread earlier than the rumors, the rumors will be almost impossible to spread on the Internet. Thirdly, it is more important for the government to increase education and improve the ability of citizens to reveal the rumor than to increase the spread of the truth after the rumor occurs. Parameter $f(m)$ represents the ability of people to reveal the rumor, and from the comparison of Figures 9 and 10, improving the self-discrimination ability of citizens ($f(m)$) will permanently benefit the control of rumors and the spread of truth, while the confrontation mechanism may become ineffective (E.g.,$T_{truth_in} > 5$).

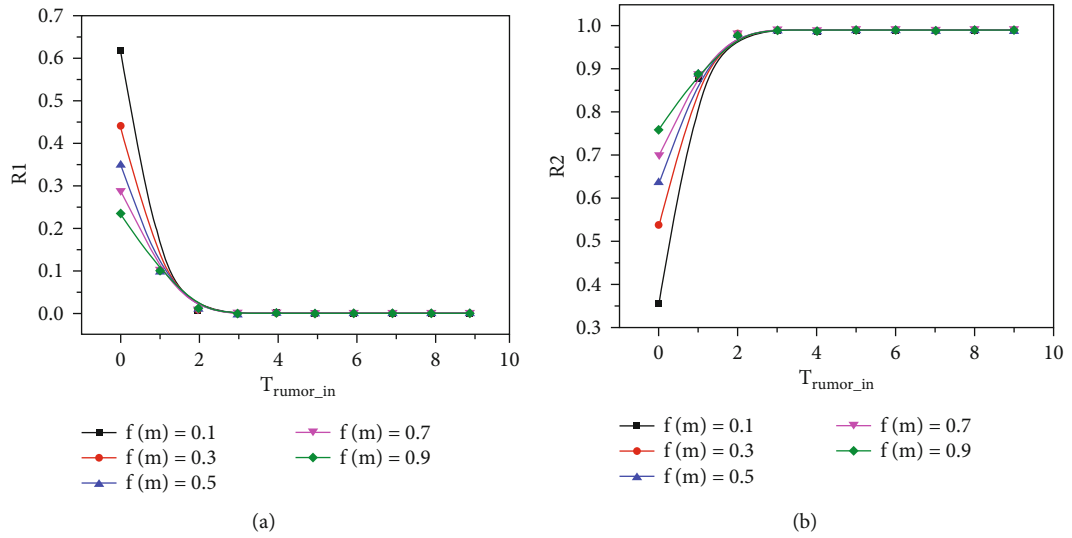


FIGURE 11: Densities of stifler1 and stifler2 for different values of $f(m)$ and $T_{\text{rumor_in}}$ in the steady-state.

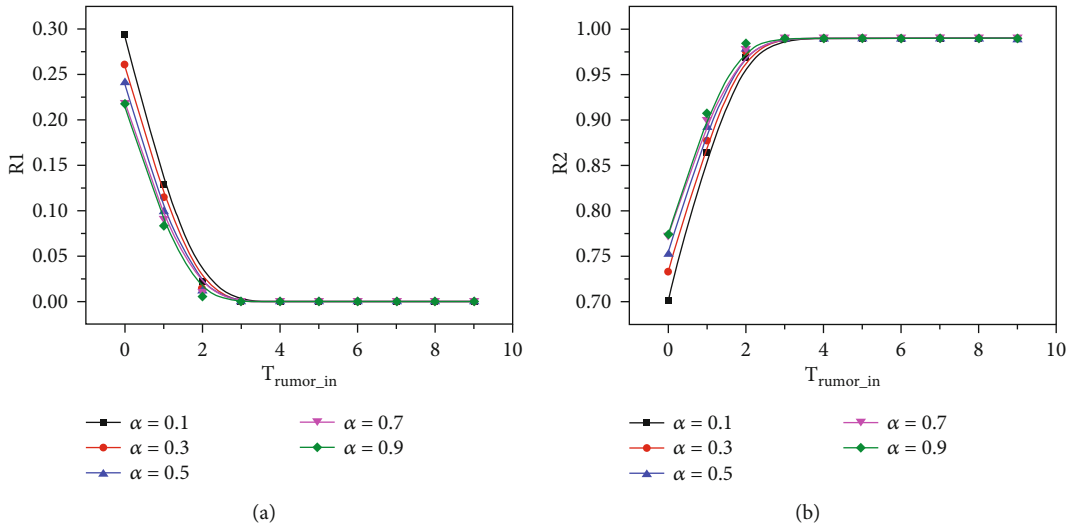


FIGURE 12: Densities of stifler1 and stifler2 for different values of α and $T_{\text{rumor_in}}$ in the steady-state.

5. Conclusions

Rumors, as a special kind of information, usually damage reputation, cause social harm and can lead to a crisis for an individual or even an entire country. To minimize the damage caused by rumors, it is important to establish a reasonable model to study the process of rumor spreading. In this paper, we proposed an adjusted model in a generated heterogeneous network, called the 2SIH2R rumor-spreading model, based on the traditional SIR model. The following significant results were obtained:

(1) Mechanisms of discernibility and confrontation were introduced to the model. Moreover, the 2SIH2R rumor-spreading model can be simplified into the traditional SIR and SEIR models under special circumstances

(2) After creating the mean-field equations to describe the dynamics of the 2SIH2R model, we obtained the spreading threshold under three cases by mathematical derivation. When only the rumor existed in the network, the spreading threshold was $\lambda_{1c} = (<k>/<k^2>)\gamma_1 - f(m)\eta/(1-f(m))$. When only the truth existed in the network, the spreading threshold was $\lambda_{2c} = (<k>/<k^2>)\gamma_2$. When there was one spreader1 and one spreader2 at $t = 0$, the condition of the spreading threshold of the 2SIH2R model was $\lambda_1(1-f(m)) + (\theta_1/\theta_1 + \theta_2)f(m)\eta > (<k>/<k^2>)\gamma_1$ or $\lambda_2 > (<k>/<k^2>)\gamma_2$. If the condition was not satisfied, the rumor or the truth could not spread widely in the heterogeneous network

(3) In a numerical simulation, we changed the strength of the discernible and confrontation mechanisms to

study how they affected the impact of the rumor. The results show that under the same conditions, the greater are the mechanisms of discernibility and confrontation, the smaller is the influence of the rumor

(4) The earlier release of the truth about the event by the government can significantly control the rumor

(5) It is more effective to publish the truth in advance than after the rumor has appeared. In other words, through machine learning and public opinion monitoring technology in the era of big data, the government or authoritative media can spread some information or preissued warnings before a rumor appears. In this way, the influence of rumors can be reduced, and even the appearance of rumors can almost be stifled

(6) It is more important for the government to increase education and improve the ability of citizens to reveal the rumor than to increase the spread of the truth after the rumor occurs

Data Availability

The generated data used to support the findings of this study have not been made available because the data is randomly generated according to the rules.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this paper.

Acknowledgments

This paper is financially supported by the Beijing Municipal Natural Science Foundation (9202018), Fundamental Research Funds for the Central Universities (CUC21GZ005, CUC2000004, CUC19ZD002, 2018CUCTJ047), and Academic Research Project of China Federation of Radio and Television Association (2020ZGLH015).

References

[1] R. K. Garrett, "Troubling consequences of online political rumoring," *Human Communication Research*, vol. 37, no. 2, pp. 255–274, 2011.

[2] S. Galam, "Modelling rumors: the no plane pentagon french hoax case," *Physica A*, vol. 320, pp. 571–580, 2003.

[3] S. A. Thomas, "Lies, damn lies, and rumors: an analysis of collective efficacy, rumors, and fear in the wake of katrina," *Sociological Spectrum*, vol. 27, no. 6, pp. 679–703, 2007.

[4] N. DiFonzo, N. M. Robinson, J. M. Suls, and C. Rini, "Rumors about cancer: content, sources, coping, transmission, and belief," *Journal of Health Communication*, vol. 17, no. 9, pp. 1099–1115, 2012.

[5] H. J. Oh and H. Lee, "When do people verify and share health rumors on social media? The effects of message importance, health anxiety, and health literacy," *Journal of Health Communication*, vol. 24, no. 11, pp. 837–847, 2019.

[6] G. W. Allport and L. Postman, *The Psychology of Rumor*, American Psychological Association, 1947.

[7] Y. L. Zan, J. L. Wu, P. Li, and Q. Yu, "SICR rumor spreading model in complex networks: counterattack and self- resistance," *Physica A*, vol. 405, pp. 159–170, 2014.

[8] Q. M. Liu, T. Li, and M. Sun, "The analysis of an SEIR rumor propagation model on heterogeneous network," *Physica A*, vol. 469, pp. 372–380, 2017.

[9] D. J. Daley and D. G. Kendall, "Epidemics and Rumours," *Nature*, vol. 204, no. 4963, pp. 1118–1118, 1964.

[10] D. J. Daley and D. G. Kendall, "Stochastic rumours," *IMA Journal of Applied Mathematics*, vol. 1, no. 1, pp. 42–55, 1965.

[11] D. H. Zanette, "Critical behavior of propagation on small-world networks," *Physical Review E*, vol. 64, no. 5, article 050901, 2001.

[12] D. H. Zanette, "Dynamics of rumor propagation on small-world networks," *Physical Review E*, vol. 65, no. 4, article 041908, 2002.

[13] D. J. Watts and S. H. Strogatz, "Collective dynamics of 'small-world' networks," *Nature*, vol. 393, no. 6684, pp. 440–442, 1998.

[14] Y. Moreno, M. Nekovee, and A. Pacheco, "Dynamics of rumor spreading in complex networks," *Physical Review E*, vol. 69, no. 6, article 066130, 2004.

[15] A. L. Barabási and R. Albert, "Emergence of scaling in random networks," *Science*, vol. 286, no. 5439, pp. 509–512, 1999.

[16] J. J. Wang, L. J. Zhao, and R. B. Huang, "SIRaRu rumor spreading model in complex networks," *Physica A*, vol. 398, pp. 43–55, 2014.

[17] J. J. Wang, L. J. Zhao, and R. B. Huang, "2SI2R rumor spreading model in homogeneous networks," *Physica A*, vol. 413, pp. 153–161, 2014.

[18] L. Zhao, J. Wang, Y. Chen, Q. Wang, J. Cheng, and H. Cui, "SIHR rumor spreading model in social networks," *Physica A*, vol. 391, no. 7, pp. 2444–2453, 2012.

[19] L. Zhao, X. Qiu, X. Wang, and J. Wang, "Rumor spreading model considering forgetting and remembering mechanisms in inhomogeneous networks," *Physica A*, vol. 392, no. 4, pp. 987–994, 2013.

[20] R. L. Rosnow, J. L. Esposito, and L. Gibney, "Factors influencing rumor spreading: replication and extension," *Language & Communication*, vol. 8, no. 1, pp. 29–42, 1988.

[21] L. A. Huo and Y. Y. Cheng, "Dynamical analysis of a IWSR rumor spreading model with considering the self- growth mechanism and indiscernible degree," *Physica A*, vol. 536, article 120940, 2019.

[22] L. L. Xia, G. P. Jiang, B. Song, and Y. R. Song, "Rumor spreading model considering hesitating mechanism in complex social networks," *Physica A*, vol. 437, pp. 295–303, 2015.

[23] A. Rizzo, M. Frasca, and M. Porfiri, "Effect of individual behavior on epidemic spreading in activity-driven networks," *Physical Review E*, vol. 90, no. 4, article 042801, 2014.

[24] K. Afassinou, "Analysis of the impact of education rate on the rumor spreading mechanism," *Physica A*, vol. 414, pp. 43–52, 2014.

[25] L. A. Huo, F. Ding, and Y. Y. Cheng, "Dynamic analysis of a SIbInIu, rumor spreading model in complex social network," *Physica A*, vol. 523, pp. 924–932, 2019.

[26] M. Askarizadeh, B. T. Ladani, and M. H. Manshaei, "An evolutionary game model for analysis of rumor propagation and

control in social networks," *Physica A*, vol. 523, pp. 21–39, 2019.

[27] N. Zhang, H. Huang, B. Su, J. Zhao, and B. Zhang, "Dynamic 8-state ICSAR rumor propagation model considering official rumor refutation," *Physica A*, vol. 415, pp. 333–346, 2014.

[28] K. Ji, J. Liu, and G. Xiang, "Anti-rumor dynamics and emergence of the timing threshold on complex network," *Physica A*, vol. 411, pp. 87–94, 2014.

[29] M. Jiang, Q. Gao, and J. Zhuang, "Reciprocal spreading and debunking processes of online misinformation: a new rumor spreading-debunking model with a case study," *Physica A*, vol. 565, article 125572, 2021.

[30] Y. Wang, F. Qing, J.-P. Chai, and Y.-P. Ni, "Spreading dynamics of a 2sih2r, rumor spreading model in the homogeneous network," *Complexity*, vol. 2021, Article ID 6693334, 9 pages, 2021.

[31] A. I. E. Hosni, K. Li, and S. Ahmad, "Analysis of the impact of online social networks addiction on the propagation of rumors," *Physica A*, vol. 542, article 123456, 2020.

[32] D. M. Fan, G. P. Jiang, Y. R. Song, and Y. W. Li, "Novel fake news spreading model with similarity on PSO-based networks," *Physica A*, vol. 549, article 124319, 2020.

[33] M. Yan, X. Lou, and Y. Wang, "Channel noise optimization of polar codes decoding based on a convolutional neural network," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 1434347, 10 pages, 2021.

[34] C. Y. Sang and S. G. Liao, "Modeling and simulation of information dissemination model considering user's awareness behavior in mobile social networks," *Physica A*, vol. 537, article 122639, 2020.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 9667506, 11 pages
https://doi.org/10.1155/2022/9667506

Research Article

A Novel Vehicle Detection Framework Based on Parallel Vision

Ying Zhuo,[1] **Lan Yan**,[2] **Wenbo Zheng**,[2] **Yutian Zhang**,[1] **and Chao Gou**[1]

[1]*Shenzhen Campus of Sun Yat-sen University, Sun Yat-sen University, Shenzhen 518107, China*
[2]*Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China*

Correspondence should be addressed to Chao Gou; gouchao@mail.sysu.edu.cn

Received 15 October 2021; Accepted 17 December 2021; Published 12 January 2022

Academic Editor: Ming Yan

Copyright © 2022 Ying Zhuo 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.

Autonomous driving has become a prevalent research topic in recent years, arousing the attention of many academic universities and commercial companies. As human drivers rely on visual information to discern road conditions and make driving decisions, autonomous driving calls for vision systems such as vehicle detection models. These vision models require a large amount of labeled data while collecting and annotating the real traffic data are time-consuming and costly. Therefore, we present a novel vehicle detection framework based on the parallel vision to tackle the above issue, using the specially designed virtual data to help train the vehicle detection model. We also propose a method to construct large-scale artificial scenes and generate the virtual data for the vision-based autonomous driving schemes. Experimental results verify the effectiveness of our proposed framework, demonstrating that the combination of virtual and real data has better performance for training the vehicle detection model than the only use of real data.

1. Introduction

Recently, with the breakthrough progress of artificial intelligence technology, intelligent vehicles with the advanced driving assistance system (ADAS) are vigorously launched on the market [1, 2]. The ADAS of the intelligent vehicle collects the surrounding data from the sensors like radars and cameras and then performs road object detection and so on. With the detection results and the map data, the control system of the intelligent vehicle proposes autonomous driving schemes. If necessary, the ADAS can even take over the vehicle to avoid accidents.

As the important information sources of intelligent vehicles, vision systems play crucial roles in safe and efficient automatic driving. It can not only obtain the information accurately but also help to conduct the analysis promptly [3]. With the rapid development of intelligent visual perception technology, the deep learning methods for the vision-based intelligent driving models far exceed the traditional methods. These deep learning-based methods require a large amount of labeled data for training. These data should be diverse enough to ensure the model validity under complex conditions. However, the collection and annotation of data are time-consuming and costly, especially under complex driving scenes. What is more, annotating images by hand is often accompanied by subjective judgment, making mistakes in the labeling process. Artificial scenes, which simulate the actual traffic scenes, can be utilized to solve the above problems of real data. They can also help to collect data under various complex situations more easily. What is more, the detailed and accurate annotation information is automatically generated during the collection of virtual data.

With the development of game engines [4, 5] and virtual reality [6–8], the construction of vivid artistic scenes has made great progress. Therefore, parallel vision technology [9–11], which combines real and virtual data to conduct the experiments, obtains better research value and broader application prospects. In the computer vision field, parallel vision [9] is the popularization and application of the ACP (artificial societies, computational experiments, and parallel execution), a theory of complex system modeling and control [12–14]. It firstly simulates and represents complex actual scenes through artificial scenes. Then, computational experiments are used to train, verify, and evaluate various visual models. Finally, the parallel execution of virtual-real interaction is applied to optimize the visual model online,

completing the intelligent perception and understanding of the complex environment. Figure 1 presents the basic framework and architecture of parallel vision [10].

In this work, we propose a novel vehicle detection framework based on the parallel vision theory, using specially designed virtual datasets to help train the vehicle detection model. Concretely, a method to construct artificial scenes and generate virtual data is presented at first. This method can generate a large number of artificial scenes to simulate complex real scenes. These generated scenes meet the requirements of quantity, diversity, scale, and specific occlusion levels. What is more, accurate annotation is automatically annotated during the image generation process. Subsequently, we choose DPM (Deformable Parts Model) [15] and Faster R-CNN [16] as the vehicle detection model. Experiments demonstrate the effectiveness of our framework, which are conducted on the combination of the KITTI [17], PASCAL VOC [18], and MS COCO [19] datasets and the constructed virtual datasets.

Motivated by the above analysis, the main contributions of our paper are threefold. First, a virtual dataset of traffic scenes is generated. By configuring the camera parameters and environmental conditions, the proposed method can flexibly construct artificial scenes with different map data. Second, we apply the parallel vision theory to vehicle detection. Combined with the virtual data, the parallel vision addresses the difficulty of collecting and labeling a large amount of diverse data. Finally, experimental results on the combination of the KITTI, PASCAL VOC, and MS COCO datasets and the constructed virtual dataset verify the effectiveness of the proposed framework.

The remainder of this paper is organized as follows. Some related work is introduced in Section 2. Section 3 presents our method to construct artificial scenes and generate virtual datasets with annotation. Experiments are conducted in Section 4 to validate the utility of our vehicle detection framework based on parallel vision. Section 5 provides the conclusions in the end.

2. Related Work

There are numerous researches about the artificial scene construction and some of them have achieved great results. Qureshi and Terzopoulos [20] first use OpenGL to construct virtual train stations and generate virtual pedestrians based on real railway station scenes. Using 2D virtual image data generated from 3D models in the Google 3D Warehouse, Sun and Saenko [21] train a 2D object detector based on the discriminative decorrelation method. And Hattori et al. [22] use virtual data to train the pedestrian detector under the specific scenario. Chao et al. [23] take the spatiotemporal information of traffic flows as a 2D texture and then formula the generation of new traffic flows as a texture synthesis process. Then, the synthesized vehicle trajectory data is placed to virtual road networks through a cage-based registration scheme so that they can populate virtual road networks with realistic traffic flows. By using the stochastic grammar, Jiang et al. [24] propose a learning-based pipeline to automatically generate and render indoor scenes with the physics-based rendering. With the foreground and background elements of traffic scenes, Yuan et al. [25] construct the traffic scenes by the scene model.

With the Unity3D game engine, Gaidon et al. [26] generate a virtual KITTI dataset based on the real KITTI dataset. The virtual KITTI dataset contains automatically generated annotation information of object detection, semantic segmentation, image depth, and optical flow. It is allowed to set different weather like rainy and foggy days, different periods like morning and dusk, and different camera orientations for the artificial scenes, thus obtaining a more diverse virtual dataset than the real KITTI dataset. Based on the Unity3D game engine, Ros et al. [27] establish a virtual city image dataset SYNTHIA with automatically generated pixel-level semantic annotation, including sidewalks, traffic signs, buildings, trees, street lights, pedestrians, vehicles, and other elements. Recently, generative adversarial networks have also been applied to images synthesis [28–30]. Brock et al. [30] use the trunacation trick to control of the tradeoff between sample variety and fidelity. But these synthetic images lack corresponding annotation information and there are artifacts sometimes. Liao et al. [31] consider that the image generation process should be modeled in 3D space. They generate the abstract 3D representation at first, which is then projected to and refined in the 2D image domain. Therefore, the image synthesis process can be modeled jointly in 3D and 2D spaces.

In this work, we propose a method to construct large-scale artificial scenes flexibly. By changing lighting conditions (from sunrise to sunset) and weather (sunny, cloudy, rainy, foggy, etc.), the proposed method can construct artificial scenes with different map data. And we can obtain the diverse contents of artificial scenes by adjusting the position, height, or orientation of the cameras allows.

3. Artificial Scene Construction and Virtual Dataset Generation

In this section, numerous artificial traffic scenes are constructed at first. We then generate various virtual datasets by tuning different parameters, like the configuration of virtual cameras and the placement of vehicles on the roadsides.

3.1. Artificial Scene Construction. In this section, we introduce our method of artificial scene construction for automatic driving task, as shown in Figure 2. To increase the authenticity of the constructed scenes, we export geographic data from OpenStreetMap, making the geographic attribute of the virtual world correspond to the real world. The geographic data is then imported into the CityEngine software. By editing the CGA (Computer Generated Architecture) rules, we can utilize CityEngine to design a realistic artificial scene, which includes roads, buildings, shops, gardens, trees, and sidewalks. Imported into Unity3D, the artificial scene can be refined by adding the models like vehicles and pedestrians as well as weather systems. And we can adjust the final result of the artificial scene through scripts and coloring programs.

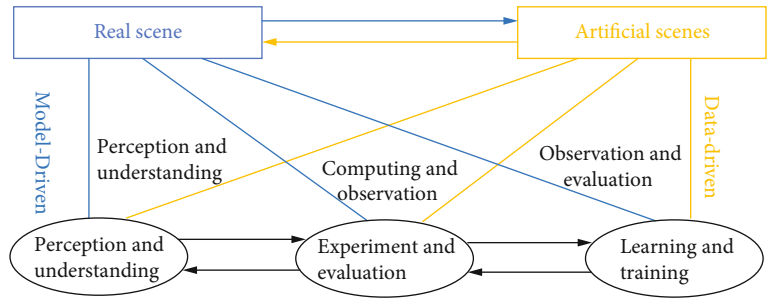


Figure 1: Framework of parallel vision.

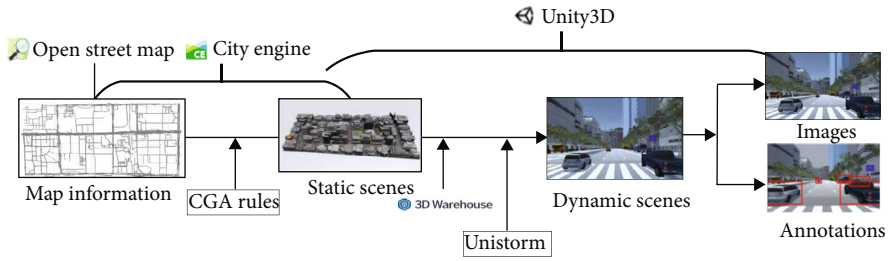


Figure 2: Artificial scene construction pipeline for automatic driving tasks.

3.2. Static Artificial Scene Construction. CityEngine is mainly used to construct a static artificial scene. Firstly, an urban road network is extracted, which is centered on Zhongguancun, Beijing, with a length of 3 kilometers and a width of 2 kilometers. Then, through the OpenStreetMap (OSM), we export the above geographic data as OSM format files, which is subsequently imported into CityEngine. To construct the static city scene in CityEngine, the basic data, texture data, and model component data are needed. Concretely, basic data refers to the vector data needed for modeling and the geographic data needed to construct the scenes. And texture data can help us use rules more easily and make the model more beautiful. And model component data can present complex details and highly reusable models, such as landmarks and tree models. CityEngine is a programmatic modeling application, and we can utilize CGA rules to define how the 3D models are constructed in CityEngine. Since that CGA rules are based on Python, numerous functions are provided to edit CGA rules conveniently. It is critical to extract the common between different objects when we edit CGA rules.

3.3. Dynamic Artificial Scene Construction. Following the construction of the static artificial scene, dynamic models are added dynamic models to construct the dynamic artificial scene in Unity3D, such as vehicles, pedestrians, and the weather system. The static artificial scene in CityEngine is first exported as fbx format models and then imported into Unity3D. Similarly, traffic models like cars, trucks, and buses from Google 3D Warehouse are imported into Unity3D. And we can adjust these traffic models to different sizes and place them in different locations.

Unistorm is applied as the weather system during this process. Unistorm is a powerful Unity3D weather plug-in that can simulate realistic weather such as sunny, partly cloudy, fog, heavy snow, light snow, heavy rain, light rain, and thunderstorm. What is more, Unistorm has an advanced time system and interactive interface, which allows us to adjust the time and weather conveniently.

3.4. Annotation Information Generation. Ground truth annotation information is crucial for the design and evaluation of vision algorithms. Images are traditionally labeled by hand, which is time-consuming and error-prone. Taking semantic segmentation as an example, it usually takes 30 to 60 minutes to label a traffic image containing 10-20 object categories [32]. In addition, manual annotation is subjective. For example, different annotators label different semantic tags for the same image, especially near the object boundary. To solve the above problems, we propose to automatically generate accurate ground truth labels by unity3D. Figure 3 presents some examples of ground truth annotation, including depth, optical flow, object tracking, object detection, instance segmentation, and semantic segmentation.

It is accurate and effective to generate ground truth annotation through Unity3D. The depth ground truth is generated by the built-in depth buffer information, which offers the depth data of the coordinates on the screen. The depth range has a nonlinear distribution from 0 to 1, and 1 means infinitely distant. As for optical flow, the instantaneous speed of a moving object is calculated on the imaging plane at first. Then, according to the changes of pixels in the image sequence, the correspondence between the previous frame and the current frame is found out and the optical flow ground truth is generated. The bounding boxes of object tracking and object detection are generated by MeshFilter and Shaders, the components in Unity3D. The ground truth of semantic segmentation is directly generated by unlit

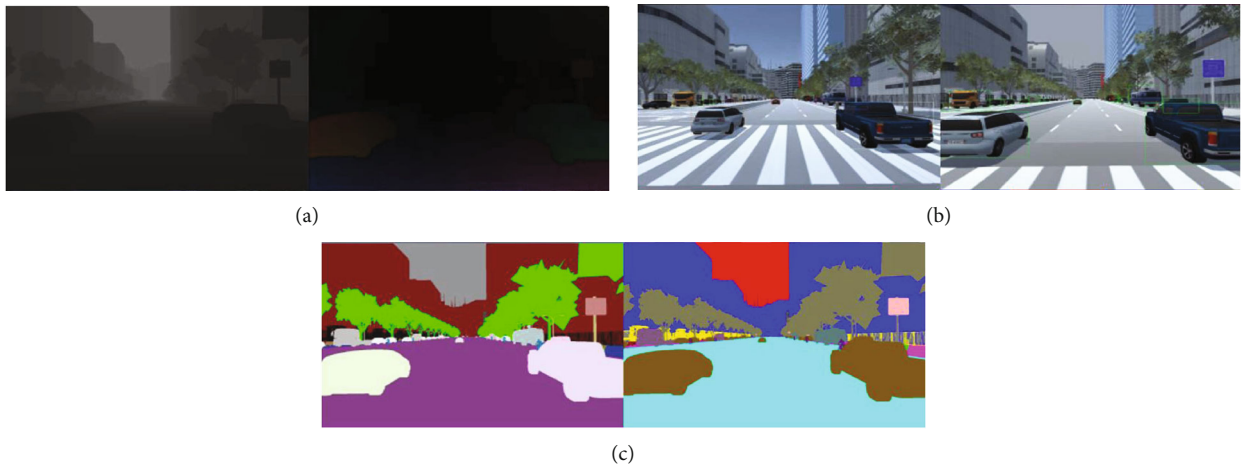


FIGURE 3: Examples of ground truth annotation automatically generated by Unity3D: (a) depth (left) and optical flow (right); (b) object tracking (left) and object detection (right); (c) instance segmentation (left) and semantic segmentation (right).

shades on the material of different objects, and each category corresponds to a unique color. The generation of instance segmentation ground truth is similar. Each instance is assigned a unique color annotation and the color outputted from the modified shader is not affected by light and shadow conditions.

3.5. Virtual Dataset Generation. With the constructed artificial scenes for automatic driving tasks, we generated three different sub-datasets of the ParallelEye [33] dataset, namely, ParallelEye_A, ParallelEye_B, and ParallelEye_C datasets, which contain 5313, 5781, and 4887 images, respectively. The images in ParallelEye_A are the traffic scenes under different weather and shooting angles, used to study the impact of these factors on the vehicle detection models. As for ParallelEye_B, it can help improve the model performance of detecting vehicles at the roadside. There are enough objects with occlusion on each image in ParallelEye_C dataset, which help to study the model performance to detect occluded vehicles. Sample pictures of the three sub-datasets are shown in Figure 4.

For the reason that different weather and shooting angles have a significant impact on the performance of vehicle detection models [34–36], we take weather conditions and shooting angles into consideration at first when generating the virtual datasets. The weather in the artificial scene is adjusted to cloudy at first. And the car camera is set at the height of 2 meters, imitating the camera on the roof. Next, we rotate the camera to five directions, that is, -30°, -15°, 0°, 15°, and 30° relative to the lane deviation. The data is collected once in each direction. These configurations contribute to obvious changes in the appearance of objects. Then, we adjust the camera angle back to 0° and collect data once on rainy and foggy days, respectively. The data collected above consist of the ParallelEye_A dataset. The image resolution of these data is 500×375. There are about 9 to 12 objects on each image, with less occlusion. The size of different objects is small because the road is wide and many vehicles are parked on the side of the road far from the lane where the camera is located.

Since that the virtual scenes in ParallelEye_A dataset simulate the camera on the vehicle driving on the lane, the images in ParallelEye_A dataset present the scenes facing the lane. What is more, the vehicle mentioned above moves on straightly when we generate ParallelEye_A dataset. Therefore, the objects in the images of ParallelEye_A dataset only have a small range of angle changes. To address the above problem, we first adjust the angle of the virtual camera to 90°, so that it faces the side of the road. Then, the camera is set to move up and down when the vehicle is moving. At the same time, we use the script to control vehicles at the roadside to park at numerous angles. Thus, ParallelEye_B dataset is generated. The image resolution of this dataset is 500×375. With less occlusion, there are only 1 or 2 objects on each image, which have large sizes and numerous angles.

Obviously, there are few images with objects occluded in ParallelEye_A and ParallelEye_B dataset and the image resolution is low. Therefore, we further generate ParallelEye_C dataset. In order to increase the number of objects and make more occlusion, we place a lot of vehicles at the adjacent lanes and the roadside. The image resolution of ParallelEye_C dataset has been refined to 1000×3000. What is more, the color of objects varies randomly during the generation of ParallelEye_C dataset. As shown in Figure 5, the color of the same vehicle is different between the adjacent two frame images. Therefore, there are enough objects with occlusion on each image in ParallelEye_C dataset. These objects not only have large sizes but also are diverse enough to ensure the model validity under complex conditions.

4. Experimental Results

In this section, we conduct the experiments to validate the effectiveness of our vehicle detection framework. We first introduce the real dataset used in the experiment. And we choose DPM and Faster R-CNN as the vehicle detection model in this work. Then, experiments are conducted on the combination of real and virtual datasets.

Finally, we analyze and summarize the results of the experiment.

(a) (b)

Figure 4: Example images of three sub-datasets: (a) ParallelEye_A (left) and ParallelEye_B (right); (b) ParallelEye_C.

Figure 5: Two adjacent frame examples in the ParallelEye_C dataset.

4.1. Model Description. Deformable part model [15] is a classic object detection model, called DPM for short. DPM models the object as a combination of several components. A DPM model includes a root filter and several part filters. The root filter focuses on the entire object, and the part filters focus on different components. There are deformation parameters to indicate the deformation cost of different components so that the relative position of each component and the main part is relatively stable. For a certain position, DPM uses a score function to determine whether there is an object.

As for object detection models based on deep learning, they can be divided into two categories: one-stage methods and two-stage methods. The former directly produces the category probability and coordinate position of the object. The latter divides the detection problem into two stages, first generating candidate regions and then classifying the candidate regions. Faster R-CNN [16] model is the most classic object detection model in two-stage methods. After obtaining the feature map of the input image, the Region Proposal Network is used to generate region proposals. This layer uses softmax to determine whether the anchor is foreground or background and then uses border regression to correct the anchor to obtain an accurate proposal. RoI pooling integrates feature maps and proposals to extract proposal feature maps. The proposal feature map is classified into specific categories by the next classifier through softmax. At the same time, the classifier uses L1 Loss to complete bounding box regression and obtain the precise position of the object.

4.2. Data Preparation. The proposed framework is based on the parallel vision theory, and therefore, we should prepare real datasets apart from the virtual dataset constructed above. We use VOC_COCO and KITTI datasets as real datasets in this work.

And the VOC_COCO dataset consists of PASCAL VOC and MS COCO dataset.

PASCAL VOC dataset is provided by PASCAL VOC Challenge, which is a benchmark test of visual object classification, recognition, and detection [18]. Apart from the dataset with standard annotation, PASCAL VOC Challenge also provides the standard evaluation system of object detection algorithms and learning performance. Nowadays, both PASCAL VOC Challenge and PASCAL VOC dataset have become a widely accepted standard in the field of object detection. The proposed framework is applied to vehicle detection under complex driving conditions. Therefore, during the experiment, we focus on the vehicle objects in the traffic scene, that is, the cars, buses, and trucks. For the reason that there are no trunk objects in PASCAL VOC dataset, we add MS COCO dataset [19] including truck objects. To construct the real datasets, we select 1997 images of car and bus objects from PASCAL VOC dataset and 1000 images of truck objects from MS

COCO dataset. After the extraction and fusion of the above data, we normalize the image annotation format of MS COCO dataset to the VOC format, as the annotation format of PASCAL VOC dataset. This real dataset is called VOC_COCO dataset.

Using PASCAL VOC and MS COCO datasets, we can only validate the vehicle detection performance in general scenarios of the proposed framework. To demonstrate the performance of the framework under the autonomous driving scene, KITTI [17] dataset is used to conduct another experiment. KITTI is currently the largest dataset for computer vision algorithm evaluation in automatic driving scenarios throughout the world. This dataset is often used to evaluate the performance of computer vision algorithms such as vehicle detection, in the complex real environment.

4.3. Experiments on VOC_COCO Dataset. In this section, the models are trained with the real data and the combination of real and virtual dataset, respectively, and then tested the models on the VOC_COCO dataset. The VOC_COCO dataset is divided into training set and testing set with the proportion 3 : 1. We first train DPM and Faster R-CNN models on the training set of VOC_COCO. Subsequently, images are selected from the virtual datasets of ParallelEye_A, ParallelEye_B, and ParallelEye_C. We combine this virtual data with the real data from the training set of VOC_COCO with the proportion 3 : 2. Then, the combination of virtual and real datasets is used to train DPM and Faster R-CNN models.

All the models are evaluated on the testing set of VOC_COCO.

When training DPM, we train the model for car, bus, and truck objects separately. The number of components is set as 3 and the maximum number of components is 8. Ensure that the ratio of positive to negative samples is 1 : 3 in the training set and testing set. DPM trained for the car objects on VOC_COCO is shown in Figure 6.

As for the Faster R-CNN model, there is no need to train three models for car, bus, and truck objects separately.

We use the classic VGG16 network structure and set the learning rate to 0.001 with the intersection over union (IoU) set to 0.7. The model is pretrained on ImageNet for initialization. The experimental results are shown in Table 1.

It can be seen from Table 1 that both DPM and Faster R-CNN model trained by the combination of real and virtual data obtain higher average precision (AP) than the models trained by the real data. Especially for the Faster R-CNN model of bus objects, the model based on the combination of real and virtual data increases average precision by 5.1%, compared to the model trained by the real data. The average precision is only improved by 0.6% for the DPM of bus objects. It is because that the AP of the DPM model trained by the real dataset is as high as 71.5%, and the performance is hard to be improved. This experiment verifies the effectiveness of our framework.

In addition, it is noticed that DPM achieves higher AP than the Faster R-CNN model for vehicle detection of bus objects. It may be because that there are fewer bus images in VOC_COCO and these bus objects are easy to be detected. Since DPM is based on HOG features, it focuses on the features related to object detection difficulty, i.e., the clarity and occlusion rate of the edge, while Faster R-CNN is a deep learning algorithm, requiring a large amount of data to train the model.

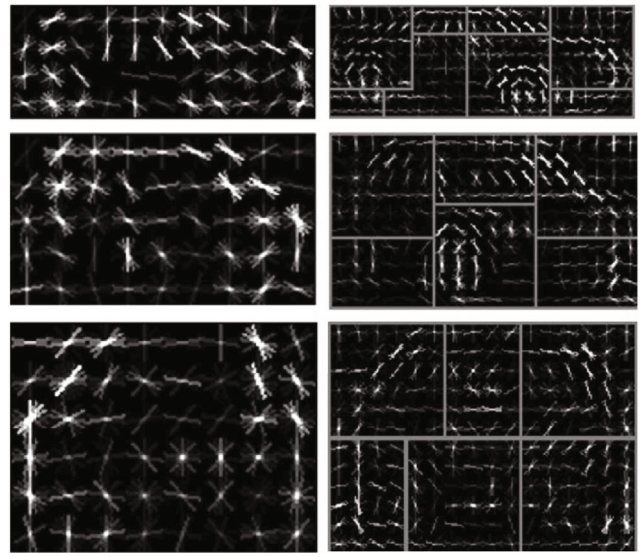


Figure 6: The visual DPM model diagram of car objects trained on VOC_COCO dataset.

The car detection result of the models trained by the VOC_COCO and ParallelEye dataset is shown in Figure 7. It is obvious that DPM performs poorly on occluded objects. While the Faster R-CNN model still has a relatively good performance under such conditions. DPM performs object detection according to the final detection score, and the final score is obtained from the root filter and component filter scores. For partially occluded objects, on the one hand, the features of objects may be affected by the features of the occlusion when the model calculates the object feature. This also affects the score of the filter at the corresponding position, thereby affecting the position with the highest calculated score and resulting in a deviation in the detection position. On the other hand, the occlusion may lower the score of the correct component position, and then, the final score is too low to detect the correct object. The correct object is missed, thereby reducing the detection accuracy. What is more, the object features of DPM are mainly in the edge area of the object because the model is based on HOG features. When there are occlusions in the edge area of the object, DPM has the worst performance of object detection. To sum up, the DPM algorithm cannot deal with the occlusion problem properly. As for the objects without occlusion, there is no significant difference in performance between the two algorithms.

4.4. Experiments on KITTI Dataset. The dataset used in the above experiments is the general dataset, rather than a

Table 1: Experimental results on the VOC_COCO and ParallelEye datasets.

Algorithm	Training dataset	Average precision		
		Car	Bus	Truck
DPM	VOC_COCO	0.482	0.715	0.482
	VOC_COCO and ParallelEye	0.496	0.721	0.499
Faster R-CNN	VOC_COCO	0.542	0.641	0.563
	VOC_COCO and ParallelEye	0.558	0.692	0.594

(a) (b)

Figure 7: Example of the car class detection results of models trained on VOC_COCO and ParallelEye dataset: (a) DPM and (b) Faster R-CNN.

specific dataset for autonomous driving. To validate the effectiveness of our framework in autonomous driving, we use KITTI and ParallelEye datasets to conduct the experiments in this section. Because there are no bus objects and few trunk objects in KITTI dataset, we only use the car data in KITTI and ParallelEye dataset for the experiments. 6684 pictures containing cars are selected from KITTI and we divide them randomly into the training set and testing set with the proportion 1 : 1.

DPM is trained for car objects with the same setting. We set the number of components as 3 and the maximum number of components as 8. The ratio of positive to negative samples is kept as 1 : 3 in both training and testing sets. We first train DPM on the KITTI training set and test the model on the testing set of KITTI. Next, some data is randomly chosen from ParallelEye_A, ParallelEye_B, and ParallelEye_C datasets. These virtual data are then combined with the real data from the training set of KITTI with the proportion 3 : 2. We use these data to train DPM and evaluate the model on the KITTI testing set.

Table 2: Evaluation results of car object detectors based on KITTI and ParallelEye datasets.

Algorithm	Training dataset	Average precision
DPM	KITTI	0.516
	KITTI and ParallelEye	0.527
Faster R-CNN	KITTI	0.741
	KITTI and ParallelEye	0.757

When training Faster R-CNN model, we use the model pretrained on ImageNet for initialization. The learning rate is set as 0.0008 and IOU is 0.7. After the algorithm converges, the model is tested on KITTI testing set. Subsequently, we use 4000 images of cars in the above randomly selected ParallelEye dataset to train Faster RCNN model with the learning rate of 0.001. Then, KITTI dataset is utilized to fine-tune the trained model and we set the learning rate as 0.0002. Finally, the model was evaluated on the

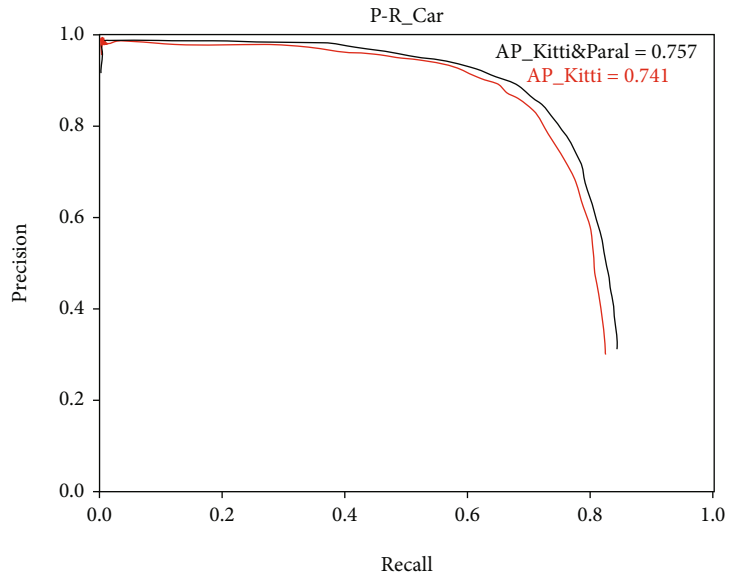


Figure 8: Precision-recall curve of Faster R-CNN on the test set of KITTI dataset. The red line represents model training using only the KITTI dataset, while the black line indicates training using the virtual dataset first and then fine-tuning using the KITTI dataset.

(a) (b)

Figure 9: Example of detection results of a virtual-real training model on a KITTI dataset. (b) Faster R-CNN.

KITTI dataset. The experimental results are shown in Table 2, and the obtained precision-recall (P-R) curve is presented in Figure 8.

According to Table 2, using the combination of virtual and real datasets to train the algorithm achieves better performance than using only the real dataset for visual model training. These experiments all verify the effectiveness of our vehicle detection framework based on the parallel vision theory in automatic driving scenarios. As mentioned before, artificial scenes can help quickly generate the labeled data. It makes up for the shortcomings of real datasets and thereby improves the performance of the model. Figure 9 presents detection results on KITTI testing set of the model trained by the virtual and real dataset.

In addition to the above experiments, we also conducted a comparative experiment on KITTI testing set. We first train detection models for car objects on ParallelEye and VOC_COCO training set separately. Then, both models are evaluated on KITTI testing set. The experimental results are listed in Table 3, and the P-R curve of Faster R-CNN is shown in Figure 10. According to Table 3 and Figure 10, the models trained by ParallelEye achieve better performance under real traffic conditions than the models trained by VOC_COCO. These results further demonstrate the effectiveness of our framework.

Table 3: Evaluation results of car object detectors on KITTI testing set.

Training dataset	Average precision	
	DPM	Faster R-CNN
ParallelEye	0.478	0.574
VOC_COCO	0.419	0.523

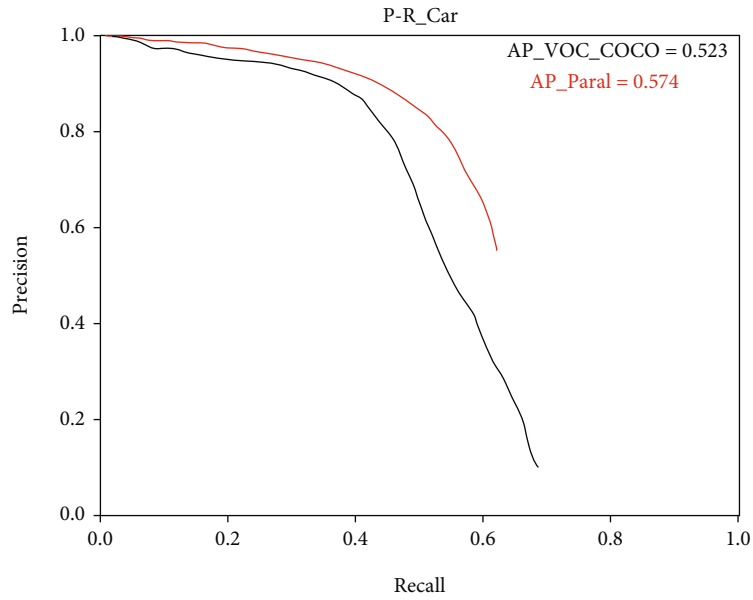


Figure 10: Precision-recall curve of Faster R-CNN model trained on ParallelEye or VOC_COCO dataset.

5. Conclusion

In this paper, we propose a novel vehicle detection framework based on parallel vision, using the specially designed virtual dataset to help train the vehicle detection model. Specially, a method is proposed to construct artificial scenes and generate virtual datasets. This method can not only simulate complex real scenes with diverse environmental changes but also generate ground-truth labels automatically. Extensive experiments have demonstrated that the combination of virtual and real datasets has better performance for training the vehicle detection model than the only use of real datasets, verifying the effectiveness of our proposed framework. Finally, it should be mentioned that we combine real and virtual data with the given proportion for model training, in order to verify the effectiveness of the proposed framework based on parallel vision. Different proportions of virtual and real data in training will be the direction of further development of the proposed framework.

Data Availability

The data used to support the findings of this study were collected from the Pascal VOC, MS COCO, and KITTI public databases as well as ParallelEye dataset. The ParallelEye dataset is also available now.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Authors' Contributions

Chao Gou proposed that we could solve the problem of traffic data shortage based on the parallel vision theory. As supervisor and funding recipient, he oversees and leads the planning and execution of research activities. Ying Zhuo and Lan Yan are the main contributors to this study. They designed the construction pipeline of artificial scenes and generated three different sub-datasets of the ParallelEye dataset. With the help of other authors, they completed vehicle detection experiments based on real and virtual data and analyzed the results. In addition, the initial manuscript is mainly finished by Ying Zhuo, who is also responsible for the visualization and revision based on the review of the manuscript. Wenbo Zheng puts forward some suggestions for vehicle detection and further improvement of virtual data generation. Yutian Zhang has done an in-depth investigation for the research and helped adjust the parameters involved in the virtual data generation. Ying Zhuo and Lan Yan contribute equally to this study.

Acknowledgments

This work is supported in part by the National Key R&D Program of China (2020YFB1600400), in part by the Key Research and Development Program of Guangzhou (202007050002), in part by the National Natural Science Foundation of China (61806198), and in part by the Shenzhen Science and Technology Program (Grant No. RCBS20200714114920272).

References

[1] M. Yan, S. Li, C. A. Chan, Y. Shen, and Y. Yu, "Mobility prediction using a weighted Markov model based on mobile user classification," *Sensors*, vol. 21, no. 5, p. 1740, 2021.

[2] M. Yan, Z. Li, X. Yu, and C. Jin, "An end-to-end deep learning network for 3D object detection from RGB-D data based on Hough voting," *IEEE Access*, vol. 8, pp. 138810–138822, 2020.

[3] A. S. Matveev, H. Teimoori, and A. V. Savkin, "A method for guidance and control of an autonomous vehicle in problems of border patrolling and obstacle avoidance," *Automatica*, vol. 47, no. 3, pp. 515–524, 2011.

[4] W. S. Bainbridge, "The scientific research potential of virtual worlds," *Science*, vol. 317, no. 5837, pp. 472–476, 2007.

[5] Q. Miao, F. Zhu, Y. Lv, C. Cheng, C. Chen, and X. Qiu, "A game-engine-based platform for modeling and computing artificial transportation systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 2, pp. 343–353, 2011.

[6] J. Sewall, J. van den Berg, M. Lin, and D. Manocha, "Virtualized traffic: reconstructing traffic flows from discrete spatio-temporal data," *IEEE Transactions on Visualization and Computer Graphics*, vol. 17, no. 1, pp. 26–37, 2011.

[7] H. Prendinger, K. Gajananan, A. B. Zaki et al., "Tokyo virtual living lab: designing smart cities based on the 3D Internet," *IEEE Internet Computing*, vol. 17, no. 6, pp. 30–38, 2013.

[8] I. Karamouzas and M. Overmars, "Simulating and evaluating the local behavior of small pedestrian groups," *IEEE Transactions on Visualization and Computer Graphics*, vol. 18, no. 3, pp. 394–406, 2012.

[9] K. Wang, C. Gou, and F. Wang, "Parallel vision: an ACP-based approach to intelligent vision computing," *Acta Automatica Sinica*, vol. 42, no. 10, pp. 1490–1500, 2016.

[10] K. Wang, C. Gou, N. Zheng, J. M. Rehg, and F.-Y. Wang, "Parallel vision for perception and understanding of complex scenes: methods, framework, and perspectives," *Artificial Intelligence Review*, vol. 48, no. 3, pp. 299–329, 2017.

[11] K. Wang, Y. Lu, Y. Wang, Z. Xiong, and F. Wang, "Parallel imaging: a new theoretical framework for image generation," *Pattern Recognition and Artificial Intelligence*, vol. 30, no. 7, pp. 577–587, 2017.

[12] W. Fei-Yue, "Parallel system methods for management and control of complex systems," *Control and Decision*, vol. 19, no. 5, pp. 485–489, 2004.

[13] F.-Y. Wang, "Parallel control and management for intelligent transportation systems: concepts, architectures, and applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 630–638, 2010.

[14] F. Wang, "Parallel control: a method for data-driven and computational control," *Acta Automatica Sinica*, vol. 39, no. 4, pp. 293–302, 2013.

[15] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2009.

[16] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 6, pp. 1137–1149, 2017.

[17] A. Geiger, P. Lenz, and R. Urtasun, "Are we Ready for Autonomous Driving? The Kitti Vision Benchmark Suite," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3354–3361, IEEE, Providence, RI, USA, 2012.

[18] M. Everingham, L. Van Gool, C. K. 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.

[19] T.-Y. Lin, M. Maire, S. Belongie et al., "Microsoft coco: common objects in context," in *European Conference on Computer Vision*, pp. 740–755, Springer, 2014.

[20] F. Qureshi and D. Terzopoulos, "Smart camera networks in virtual reality," *Proceedings of the IEEE*, vol. 96, no. 10, pp. 1640–1656, 2008.

[21] B. Sun and K. Saenko, "From virtual to reality: fast adaptation of virtual object detectors to real domains," *BMVC*, vol. 1, p. 3, 2014.

[22] H. Hattori, V. Naresh Boddeti, K. M. Kitani, and T. Kanade, "Learning scene-specific pedestrian detectors without real data," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3819–3827, Boston, MA, USA, 2015.

[23] Q. Chao, Z. Deng, J. Ren, Q. Ye, and X. Jin, "Realistic data-driven traffic flow animation using texture synthesis," *IEEE Transactions on Visualization and Computer Graphics*, vol. 24, no. 2, pp. 1167–1178, 2018.

[24] C. Jiang, S. Qi, Y. Zhu et al., "Configurable 3D scene synthesis and 2D image rendering with per-pixel ground truth using stochastic grammars," *International Journal of Computer Vision*, vol. 126, no. 9, pp. 920–941, 2018.

[25] S. Yuan, Y. Chen, H. Huo, and L. Zhu, "Analysis and synthesis of traffic scenes from road image sequences," *Sensors*, vol. 20, no. 23, p. 6939, 2020.

[26] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig, "Virtual worlds as proxy for multi-object tracking analysis," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 4340–4349, Las Vegas, NV, USA, 2016.

[27] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez, "The synthia dataset: a large collection of synthetic images for semantic segmentation of urban scenes," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3234–3243, Las Vegas, NV, USA, 2016.

[28] I. Goodfellow, J. Pouget-Abadie, M. Mirza et al., "Generative adversarial nets," *Advances in Neural Information Processing Systems*, vol. 27, pp. 2672–2680, 2014.

[29] A. Brock, J. Donahue, and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis," in *2018 IEEE International Conference on Big Data (Big Data),*, Seattle, WA, USA, 2018.

[30] H. Zhang, I. Goodfellow, D. Metaxas, and A. Odena, "Self-attention generative adversarial networks," in *International Conference on Machine Learning*, pp. 7354–7363, PMLR, 2019.

[31] Y. Liao, K. Schwarz, L. Mescheder, and A. Geiger, "Towards unsupervised learning of generative models for 3D controllable image synthesis," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5871–5880, Seattle, WA, USA, 2020.

[32] A. Kundu, Y. Li, F. Dellaert, F. Li, and J. M. Rehg, "Joint semantic segmentation and 3D reconstruction from monocular video,," in *Computer Vision – ECCV 2014*, pp. 703–718, Springer, 2014.

[33] X. Li, K. Wang, Y. Tian, L. Yan, F. Deng, and F.-Y. Wang, "The ParallelEye dataset: a large collection of virtual images for traffic vision research," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 6, pp. 2072–2084, 2018.

[34] K. Wang, W. Huang, B. Tian, and D. Wen, "Measuring driving behaviors from live video," *IEEE Intelligent Systems*, vol. 27, no. 5, pp. 75–80, 2012.

[35] K. Wang, Y. Liu, C. Gou, and F.-Y. Wang, "A multi-view learning approach to foreground detection for traffic surveillance applications," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 6, pp. 4144–4158, 2015.

[36] H. Zhang, K. F. Wang, and F. Y. Wang, "Advances and perspectives on applications of deep learning in visual object detection," *Acta Automatica Sinica*, vol. 43, no. 8, pp. 1289–1305, 2017.

Hindawi
Wireless Communications and Mobile Computing
Volume 2022, Article ID 4867220, 14 pages
https://doi.org/10.1155/2022/4867220

Research Article

Knowledge Graph Construction for Intelligent Media Based on Mobile Internet

Jianhua Dai and Jingxin Xu

Business School, China University of Political Science and Law, Beijing 100088, China

Correspondence should be addressed to Jianhua Dai; daijianhua66@163.com

Received 15 October 2021; Revised 16 November 2021; Accepted 6 December 2021; Published 11 January 2022

Academic Editor: Ming Yan

Copyright © 2022 Jianhua Dai and Jingxin Xu. 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.

Mobile Internet-based intelligent media has become a popular academic topic. This study uses the CiteSpace visualisation tool and Scientific Citation Index Expanded database to comb the existing research in the field of intelligent media from a quantitative perspective. A total of 7248 English papers were published on the topic of "intelligent media" from 2012 to 2021, and 145 highly cited papers refined were analysed. Scientific knowledge graphs were analysed from six dimensions: annual publication quantity, country of publication, institution of publication, author, keywords, and cited references. In the last 10 years, the research literature on intelligent media has been found to increase annually. Presently, the People's Republic of China and the United States of America have a high proportion of documents in this field. Chinese universities and institutions have achieved significantly in terms of the quantity and quality of documents. From the perspective of the whole intelligent media discipline, the high-yield author group has not been formed, and there is minimal cooperation amongst authors. Popular intelligent media topics include film, social media, machine learning, swarming motility, data mining, and artificial intelligence. Subject words of the main research directions are event recognition, fake news, Cable News Network model, reconfigurable intelligent surface, comprehensive survey, microblog message, strain sensor, and traffic event. Combined with popular topics and time zone maps, the future research frontier in the field of smart media is identified.

1. Introduction

Presently, the development of mobile communication, artificial intelligence, intelligent terminal technology, and social media has resulted in some technologies penetrating the media industry, such as Internet of things (IoT), cloud computing, virtual augmented reality, intelligent hardware, and big data. The transformation of new media has led to the evolution and change of media forms, content production, distribution channels, user experience, and other fields. The media space dominated by people in the past has been substantially expanded, and intelligent media has gradually developed. With the support of mobile Internet and big data technology, meeting the personalised needs and customised requirements of users has become a possibility. The emergence of various media forms has resulted in intelligent media becoming an important development direction for media organisations to win against competitors. Additionally, changes in user behaviour have promoted the intelligent development of media.

In the last decade, the number of intelligent media research has been increasing, and the distribution of intelligent media research in different disciplines has been expanding. The analysis of the development trend of intelligent technology, influence of intelligent technology on intelligent media, media market environment, and future development trend of intelligent media has an important reference value for regulators to formulate intelligent media policies and for enterprises to choose the development path of intelligent media. Although the development

of intelligent media has provided substantial convenience, it has also brought numerous social problems. This study conducts the necessary examination and reflection on previous academic research.

2. Literature Review

According to the search of scholars and Scientific Citation Index Expanded (SCIE) databases, research on intelligent media has mainly focused on such aspects as intelligent technology, personalised recommendation system, user and algorithm credibility, diversity of information environment, participation of intelligent technology in Internet information governance, information values, transparency, and algorithm accountability mechanism. This study selects highly cited articles in the aforementioned fields for analysis.

In the intelligent technology research, Pollack surveyed new technologies that incorporate artificial intelligence techniques to support older adults and help them cope with the changes of aging, particularly with cognitive decline [1]. Brooks et al. provided a detailed description of data-assisted approaches to building intelligent technology-enhanced learning systems, which focus on helping instructional experts discover insights into the teaching and learning process, and leveraged these insights as instructional interventions [2]. Marres evaluated an emerging paradigm for testing intelligent technology in society through the analysis of recent street trials of self-driving cars [3]. Jin et al. proposed the method of combining reward network and cross-entropy loss to guide the training of generators and produce high-quality music works [4]. Zhang et al. provided a computational intelligence method for different system levels, described the development direction of intelligent systems, and introduced the basic state-of-the-art concepts of data mining [5]. For intelligent systems, Liu et al. presented tactile sensors with excellent application prospects in numerous fields, such as wearable devices, medical treatment, artificial limbs, and robotics [6]. Intelligent technology has been widely used in the Industry 4.0 era, in which manufacturing will be via the Internet, to achieve internal and external network integration towards the intelligent direction [7]. Industry 4.0 strategy revolutionises manufacturing systems and processes and also leads to the formation of an intelligent supply chain [8].

In the aspect of a personalised recommendation system, numerous studies have discussed users' preference between personalised and nonpersonalised news. Sang et al. developed a recommendation system that enables bidirectional communication between users and systems using a utility range-based product recommendation algorithm to provide markedly dynamic and personalised recommendations [9]. The traditional recommendation system disregards the internal relationship between user preference and time. To address this issue, Cui et al. proposed a novel recommendation model based on time correlation coefficient and an improved K-means with cuckoo search (CSK-means) [10]. When the number of books provided by libraries is relatively high, users experience difficulty selecting the appropriate books from numerous candidate books. For this case, Tian et al. designed a personalised recommendation system for college libraries based on a hybrid recommendation algorithm [11]. According to the characteristics of each user group, Yan et al. proposed an environment prediction algorithm based on mobile user classification, and the algorithm has great application prospects in news recommendation [12]. According to statistics of users in 26 countries, although there are individual differences, users generally believe that the algorithm based on historical consumption behaviour is a better way of news recommendation than manual editing. Such variables as age, trust in news, attention to privacy, and use of mobile news have an impact on this preference [13]. Readers' attitude towards personalised news depends on the diversity of news recommendation systems and degree of care in the field of public news, rather than privacy and other factors [14]. Moreover, users' attitudes towards personalised news are heterogeneous. Young and poorly educated users favour personalised news and have minimal contact with diversified information services provided by public media.

With the help of computer technology, researchers have designed random, manual, and multiple personalised recommendation systems based on the principles of popularity, semantic filtering, and collaborative filtering to measure the impact of recommendation algorithms on content diversity [15]. Accordingly, the diversity of the algorithm recommendation has not been reduced compared with the nonpersonalised recommendation, and content topics recommended by personalised collaborative filtering systems are the most abundant. Nevertheless, the recommendation system continue to face the risk of gradual marginalisation of long-tail content, which should be countered by weighting minority content.

Scholars have determined that the algorithm formulated by dominant organisations can cause the entire industry to follow and reflect the power in the process of controlling the production and circulation of information and also demonstrate dominance by formulating industry norms [16]. Researchers are convinced that apart from emphasising the role of algorithms in constructing reality, people should also realise the function of algorithms as a relationship management mechanism between organisations and individuals [17]. As part of this organisational structure, algorithm systems can automate the management mechanism of leading organisations and also provide a common language or structure to redefine value and affect the legitimacy of other organisations and individuals [18].

In the research of transparency and algorithm accounting mechanisms, fake news remains a difficult problem for Internet users and scholars. The fake news research published by *Science* magazine has observed the proliferation of all verified real and fake news published on Twitter from 2006 to 2017 and found that the dissemination range of false information substantially exceeds that of real news regardless of theme [19]. Additionally, fake news has proven to have a powerful agenda-setting function [20]. Given the flood of false news and overstretched manual verification

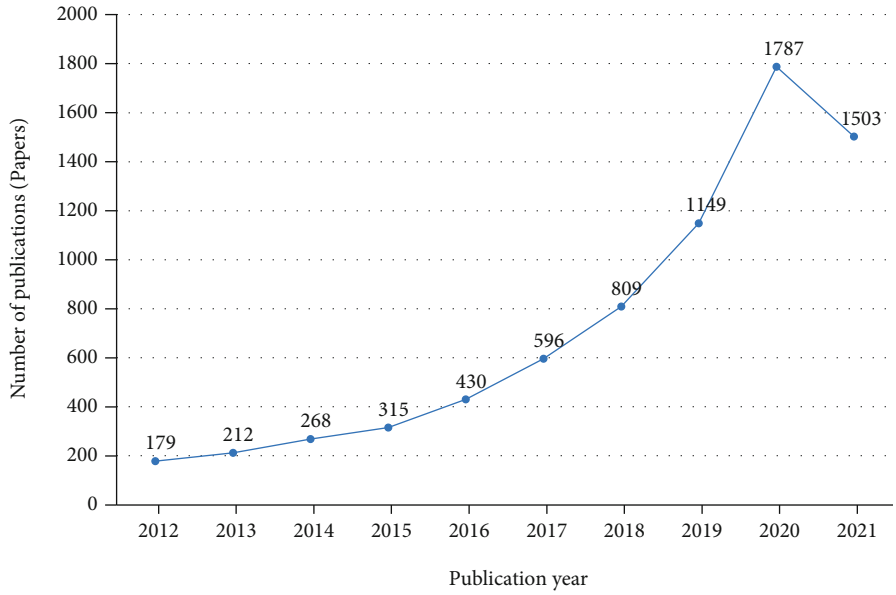


Figure 1: Number of articles published in intelligent media from 2012 to 2021.

in the Internet information environment, intelligent technology has been widely used in identifying and communicating false news [21]. In 2018, Facebook announced changes to its algorithm, prioritising reliable news in response to questions on whether or not its algorithm promotes the social spread of spam [22]. To date, no systematic review has been undertaken of studies conducted in the field of "algorithmic accountability." Wieringa obtained a perspective from critical algorithm research and also provided an interdisciplinary overview using materials from data research to law and from computer science to governance research [23].

3. Materials, Methods, and Tools

3.1. Research Tools. This study is based on the research methods of "cocitation analysis" and "cooccurrence analysis" of bibliometrics and conducts visual analysis using CiteSpace software (version 5.8. R3). Mapping knowledge domain is drawn for the cited reference and citing articles after sorting. CiteSpace, which was developed by Chaomei Chen, is a freely available Java application for visualising and analysing trends and patterns in the scientific literature. The software measures the literature in specific fields to explore the key path and knowledge turning point of discipline evolution. Thereafter, this research draws a series of visual maps, analyses the potential dynamic mechanism of discipline evolution, and explores the frontier of discipline development. The design of the tool is based on the basic assumption that actual scientific knowledge is constantly changing, similar to the gradual obsolescence and emergence of old and new literature, respectively. The so-called scientific domain refers to any boundary range that can be customised at different levels of clustering, including a scientific topic, research field, or discipline. Knowledge domain is represented by scientific literature sample data sources from different fields. This domain can be used to analyse the potential knowledge contained in massive literature data and present its structure, law, and distribution in the form of a knowledge graph.

CiteSpace can effectively help readers acquire an improved understanding of their research fields. It can not show the overall situation of a research field and also highlight some specific important documents in the development process. Specifically, CiteSpace helps answer the following questions: "Which literature is pioneering and iconic in a research field?," "Which literature plays a key role in the development of a research field?," "Which topics occupy the mainstream position in the whole research field?," "How are different research fields related to each other?," and "How did the research frontier based on a certain knowledge base evolve?". CiteSpace provides a new research perspective and a valuable scientific research tool.

This research mainly studies the development status and trend of intelligent media in the recent 10 years by analysing the six dimensions of annual publication quantity, country of publication, institution of publication, author, keywords, and cited references. CiteSpace map has two most direct foundations, namely, (1) disciplinary basis based on citation analysis and (2) technical basis based on information visualisation, thereby intuitively expressing the research results of this study. The current research reveals the main forces, evolution process, research hotspots and topic categories of intelligent media research in the past decade. Through scientific cooperation network, keyword symbiosis network, and keyword clustering analyses, this study reveals the main forces, evolution process, research hotspots, and theme categories of intelligent media research in the recent 10 years. This study uses the results of the knowledge map in predicting the possible development trend of intelligent media in the future.

3.2. Research Object and Data Source. This study selects the core collection of Web of Science (WOS) as a data source.

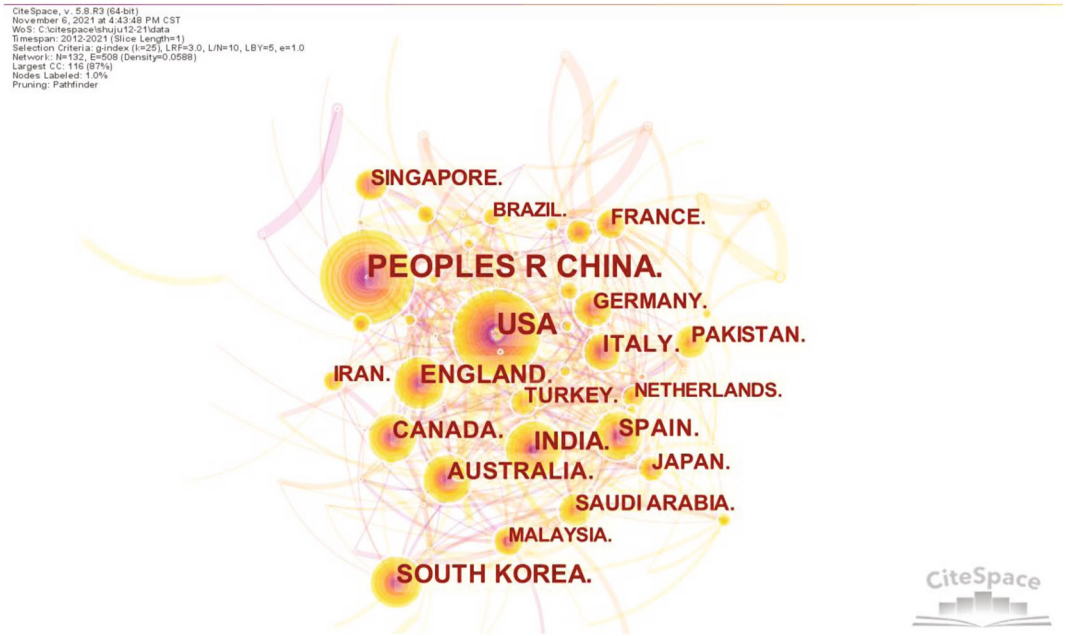


Figure 2: Cooccurrence graph of paper publication from a national perspective.

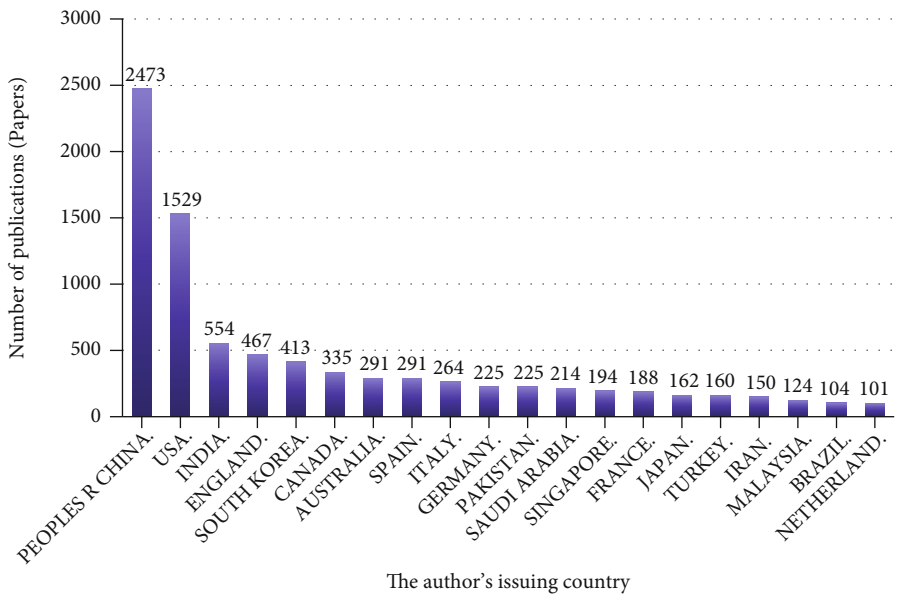


Figure 3: Countries with at least 100 papers.

According to the interpretation and analysis of the research literature in the field of intelligent media mentioned in the literature review, the relevant keywords in intelligent media have been preliminarily determined. This study searches keywords in the WOS database and attempts to unlock the keywords that can reflect the field in the WOS database. "Intelligent media" and "smart media" are used as search words for the trial check, and keywords related to the two fields of "artificial intelligence" and "media" are sorted through trial search. Combined with the interpretation of intelligent media by wireless communications and mobile computing, Li et al. [24] and Qian et al. [25] formulated a retrieval formula for subsequent trial retrieval for the research of key terms in the field of artificial intelligence and common media and enterprises using intelligent technology in social and economic activities. On this basis, the current study repeatedly attempts and adjusts the retrieval formula and browse and compare the retrieval results after each adjustment, excluding the content not related to the subject. Lastly, this study determines the retrieval formula as [TS = ((("adaptive system") OR ("machine intelligen∗") OR ("intelligent network") OR ("intelligent systhesis") OR

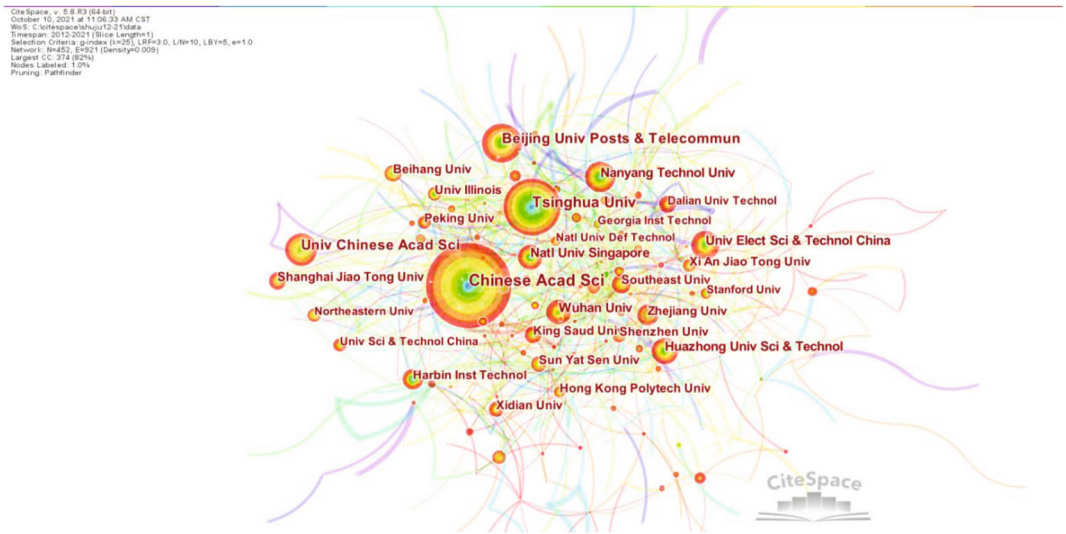


Figure 4: Cooccurrence graph of institutions.

("ambient intelligen ∗") OR ("brain model ∗") OR ("virtual reality") OR ("augmented reality") OR ("cognitive system") OR ("deep learn ∗") OR ("algorithm") OR ("expert system") OR ("formal logic ∗") OR ("heuristic program") OR ("intelligent agent ∗") OR ("local search ∗") OR ("prediction model ∗") OR ("logic program") OR ("reinforcement learning") OR (robot ∗) OR ("image recogn ∗") OR ("speech recogn ∗") OR ("distributed comput ∗") OR ("system Theor ∗") OR ("cloud computing") OR ("fuzzy control ∗") OR ("knowledge engineering") OR ("ubiquitous comput ∗") OR ("intelligent system") OR ("agent-based model ∗") OR ("multi-agent system") OR ("neural network") OR ("problem solv ∗") OR (swarm ∗) OR ("data proces ∗") OR ("artificial intelligence") OR ("text minning") OR ("feature recogn ∗") OR ("pattern analysis") OR ("intelligent creation") OR ("resource allocat ∗") OR ("immersive experience")) AND (((((smart) OR (intell ∗)) AND ((film) OR (television) OR (audio) OR (radio)))) OR ("social media") OR ("mass media") OR ("news media") OR ("short video") OR (Youtube) OR (Blog) OR (facebook) OR (Twitter) OR (QQ) OR (wechat) OR (instagram) OR (TikTok) OR (vimeo) OR (matecafe) OR (dailymotion) OR (liveleak) OR (hulu) OR (viewster) OR (twitch) OR (netflix) OR (vube) OR (weibo) OR (zhihu) OR (bilibili)))]. The data set is limited to "SCIE," time is limited to "2012–2021," language is limited to "English," document types are limited to "article" and "review," retrieval time is October 10, 2021, and a total of 7248 records are obtained. Additionally, 145 highly cited papers are obtained through refining for comparison.

4. Results and Discussion

4.1. Analysis of Annual Publications. Statistics on annual publication quantity can show the research attention of scholars in intelligent media at different periods. This study used the method of metrological visualisation analysis to calculate the annual number of 7248 retrieved articles (see Figure 1). The result indicated that the number of relevant

Table 1: Institutions with at least 40 published papers.

No.	Institution	No. of papers
1	Chinese Acad Sci	204
2	Tsinghua Univ	142
3	Beijing Univ Posts & Telecommun	104
4	Univ Chinese Acad Sci	91
5	Nanyang Technol Univ	85
6	Univ Elect Sci & Technol China	74
7	Huazhong Univ Sci & Technol	73
8	Natl Univ Singapore	70
9	Wuhan Univ	67
10	Zhejiang Univ	64
11	Harbin Inst Technol	61
12	Shanghai Jiao Tong Univ	58
13	Beihang Univ	56
14	Xidian Univ	55
15	King Saud Univ	55
16	Xi An Jiao Tong Univ	54
17	Southeast Univ	52
18	Univ Illinois	51
19	Shenzhen Univ	50
20	Hong Kong Polytech Univ	49
21	Sun Yat Sen Univ	48
22	Peking Univ	47
23	Dalian Univ Technol	46
24	Northeastern Univ	46
25	Georgia Inst Technol	45
26	Natl Univ Def Technol	44
27	Stanford Univ	44
28	Univ Sci & Technol China	43

Figure 5: Cooccurrence graph of institutions with highly cited literature.

studies on intelligent media by international scholars has been increasing annually. In 2012, there were only 179 articles on the topic of "intelligent media," but the number of articles published in this field reached 596 in 2017, 809 in 2018, 1149 in 2019, and 1787 in 2020. Given that 2021 data are only up to October 10, 2021 (i.e., retrieval time of this study), the number of articles published in this field is expected to exceed 2000 in 2021. Note that since 2017, the annual number of papers published in intelligent media has increased rapidly, and the research heat has not decreased. The general trend is to promote the entry of intelligent technology to the media industry.

4.2. Analysis of the Publication of Paper from a National Perspective. The visual analysis of the author's issuing country by CiteSpace software indicates that the node type is country. Figure 2 shows that countries with over 100 starting documents are obtained. The national cooperation perspective indicates 132 nodes, 508 connections between nodes, and network density of 0.0588. Note that the current national cooperation in intelligent media research is relatively close. Figure 2 shows that the larger a country's name and the thicker the node ring, the more the total number of articles published by the country. The number of documents issued can also reflect a country's research strength and level in this field. The thicker the connection between nodes, the closer the cooperation between countries.

The People's Republic of China ranks first in the number of published papers, with 2473 papers. The United States of America (the USA) and India are second and third with 1529 and 554 articles, respectively (see Figure 3). Note that the People's Republic of China and the USA have high proportions of documents in this field. Presently, various countries are closely connected in intelligent media research.

4.3. Analysis of Productive Institutions. The visual analysis of the document issuing organisation by CiteSpace software indicates that the node type is organisation. Organisations with over 40 published papers are obtained (see Figure 4). The cooccurrence graph of institutional cooperation shows 452 nodes, 921 connections between nodes, and network density of 0.009. Note that there is cooperation amongst institutions in intelligent media research, but such cooperation is insufficient. Figure 4 shows that the larger the name of organisations and the thicker the annual rings of the nodes representing the total number of documents issued by organisations reflect the research strength and level of these organisations in this field. The thicker the connection between nodes, the closer the cooperation between representative organisations.

The top three institutions are Chinese Academy of Sciences (CAS), Tsinghua University, and Beijing University of Posts and Telecommunications. A total of 28 institutions have a volume of at least 40 articles, 21 of which are Chinese universities and scientific research institutions and 4 are American universities (see Table 1). This result is consistent with the cooccurrence graph of the national document volume in the second part. Note that China's document volume is dominant in this field, and the cooperation between domestic institutions is relatively close.

A total of 145 highly cited documents obtained from refining were visually analysed based on CiteSpace software (see Figure 5). Amongst the highly cited documents, CAS has the most number of documents and has close cooperation with major universities and institutions locally and overseas.

CAS has the most number of papers, with a total of 12. There are 13 institutions with at least 3 articles, of which 7 are Chinese universities and scientific research institutions

Table 2: Institutions that have published at least three highly cited papers.

No.	Institution	No. of papers
1	Chinese Acad Sci	12
2	Stanford Univ	6
3	Sejong Univ	6
4	Beijing Univ Posts & Telecommun	6
5	Nanyang Technol Univ	6
6	Dalian Univ Technol	5
7	Peking Univ	3
8	Arizona State Univ	3
9	Huawei France R&D	3
10	Univ Houston	3
11	Carleton Univ	3
12	Tsinghua Univ	3
13	Beijing Univ Chem Technol	3

and 3 are American universities (see Table 2). Figure 5 and Table 2 show that CAS ranks first in intelligent media in terms of the number of publications and number of highly cited literature. The overall proportion of Chinese universities is also high. This result shows that China's research in intelligent media is at the forefront and has achieved good results in terms of quantity and quality.

4.4. Author Analysis. The author's cooccurrence analysis can reflect the cooperative relationship between the core authors and researchers in this field. Price law points out that the author's calculation formula is as follows:

$$N_{\min} = 0.749\sqrt{N_{\max}}, \tag{1}$$

where $N_{\max}$ is the number of articles published by the author with the largest number of articles in a certain field. If the number of articles published by an author is not below $N_{\min}$, then the author is called a high-yield author. Table 3 shows that if the value $N_{\max}$ is 21, then $N_{\min} \approx 3.432$ is obtained and the integer is 4. Therefore, if there are at least 4 published papers, then the author is called a high-yield author.

According to the price law, only when the high-yield author group accounts for 50% of the total can the discipline high-yield author group be formed. After calculation, the number of high-yield authors in this field accounts for only 7.19% of the total number of papers, indicating that researchers in intelligent media are relatively scattered and have not formed a high-yield author group. Table 3 shows that the author with the most number of articles (21) is Wei Wang of Sichuan University, followed by Yingchang Liang (with 14) of the University of Electronic Science and Technology. Note that the high-yield authors began to publish more papers in intelligent media in 2016.

The graph shows 529 nodes, 422 connections between nodes, and network density of 0.003. Cooperation of researchers in this field is not close (see Figure 6). Particularly, teams with relatively more papers and outstanding cooperation have two cooperative relationships (1) Jun Zhang and Qingqing Wu and Ming Li, amongst others and (2) Yingchang Liang and Muhammad Ali-Imran and Lei Zhang, amongst others.

Table 3: High-yield authors with at least six papers.

No.	Author	No. of papers	Start year
1	Wei Wang	21	2015
2	Yingchang Liang	14	2019
3	Akshi Kumar	12	2019
4	Jun Zhang	12	2017
5	Jian Wang	12	2019
6	Erik Cambria	11	2015
7	Mugen Peng	10	2019
8	Alagan Anpalagan	9	2013
9	Muhammad Imran	9	2020
10	Wei Huang	9	2018
11	shiqiang yang	8	2013
12	Lei Zhang	7	2019
13	Fadi Alturjman	7	2020
14	Koklim Alvin Yau	7	2014
15	Hua Wang	7	2017
16	M Shamim Hossain	7	2018
17	Tatseng Chua	7	2014
18	Min Chen	7	2014
19	Junaid Qadir	7	2016
20	Chang Liu	7	2019
21	Jing Liu	6	2014
22	Arun Kumar Sangaiah	6	2018
23	Liang Xiao	6	2015
24	Hui Li	6	2020
25	Bin Zhang	6	2017
26	Chao Wang	6	2013
27	Ting Liu	6	2016
28	Ming Li	6	2020
29	Qihui Wu	6	2013
30	Hongfei Lin	6	2019
31	Khan Muhammad	6	2020

4.5. High-Frequency Keyword Analysis

4.5.1. Cooccurrence Analysis of High-Frequency Keywords. Keywords are highly generalised to the literature theme. Keyword analysis of the literature is aimed at revealing the literature theme, which can reflect the research hotspot, research direction, internal relationship, and importance of this research field.

CiteSpace software was used in conducting cooccurrence analysis of the keywords of 7248 retrieved literature and removing words irrelevant to the research field (see Figure 7). Particularly, there are 526 nodes 689 connections

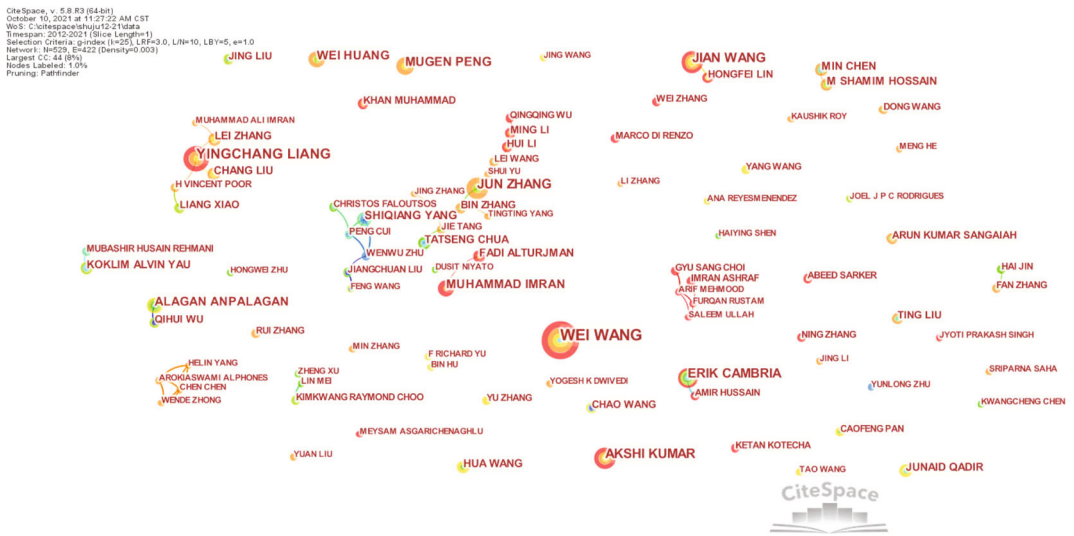


Figure 6: Cooccurrence graph of high-yield authors.

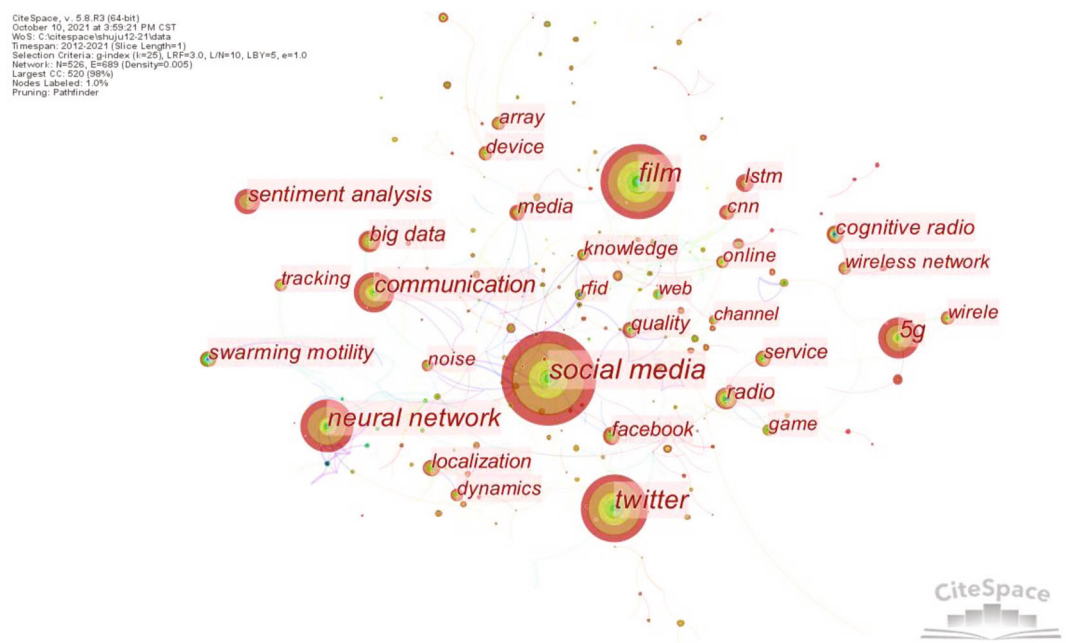


Figure 7: Cooccurrence graph of keywords.

between nodes and network density is 0.005. The larger the node and font size the higher the keyword usage frequency. The thicker the connection the closer the connection between the two keywords. The greater the centrality the more important the keywords. The general belief is that keywords with centrality above 0.1 are important and keywords with high frequency and great centrality can be used as research hotspots.

Keywords with frequency of over 40 are analysed (see Table 4). The frequency and centrality of film and communication are very high, indicating that the two keywords are more important. Related keywords include social media, twitter, neural network, 5G, sentient analysis, radio, and big data.

4.5.2. Keyword Cluster Analysis. Keyword clustering analysis can reflect the most popular research topic in this field.

CiteSpace conducts cluster analysis on high-frequency keywords in intelligent media. It sets the cluster number to 6 and obtains 6 clustering points about intelligent media (see Figure 8): film, social media, machine learning, swarming motility, data mining, and artificial intelligence. Each cluster is composed of closely related keywords. When the cluster S value is above 0.5, clustering is reasonable. When the clustering S value is above 0.7, clustering results are convincing. Given that the S values of the six clusters shown in Figure 8 are above 0.7, the clustering results of this time have a high degree of credibility.

Table 4: High-frequency keywords and their centrality list.

No.	Keyword	Frequency	Centrality	Start year
1	Social media	252	0.02	2013
2	Film	200	0.23	2014
3	Twitter	180	0.02	2015
4	Neural network	145	0.07	2013
5	Communication	115	0.21	2012
6	5G	112	0.03	2015
7	Sentiment analysis	70	0.01	2017
8	Radio	67	0.02	2012
9	Big data	64	0.01	2016
10	Cognitive radio	57	0.02	2012
11	Swarming motility	54	0.02	2012
12	Time	52	0.02	2015
13	Service	52	0.09	2013
14	Facebook	51	0.08	2014
15	Quality	50	0.07	2012
16	LSTM	49	0	2019
17	Localization	49	0.03	2013
18	Media	48	0.03	2014
19	Device	45	0.07	2015
20	Array	43	0.05	2015
21	CNN	42	0.01	2019
22	Dynamics	42	0.05	2013
23	Knowledge	41	0.04	2012

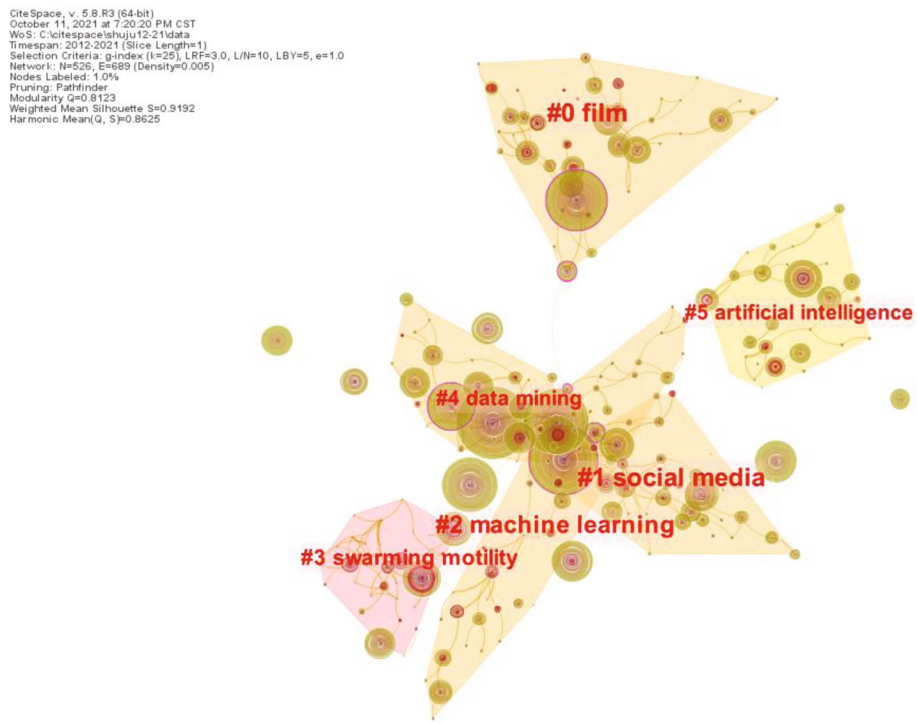


Figure 8: Keyword cluster analysis graph.

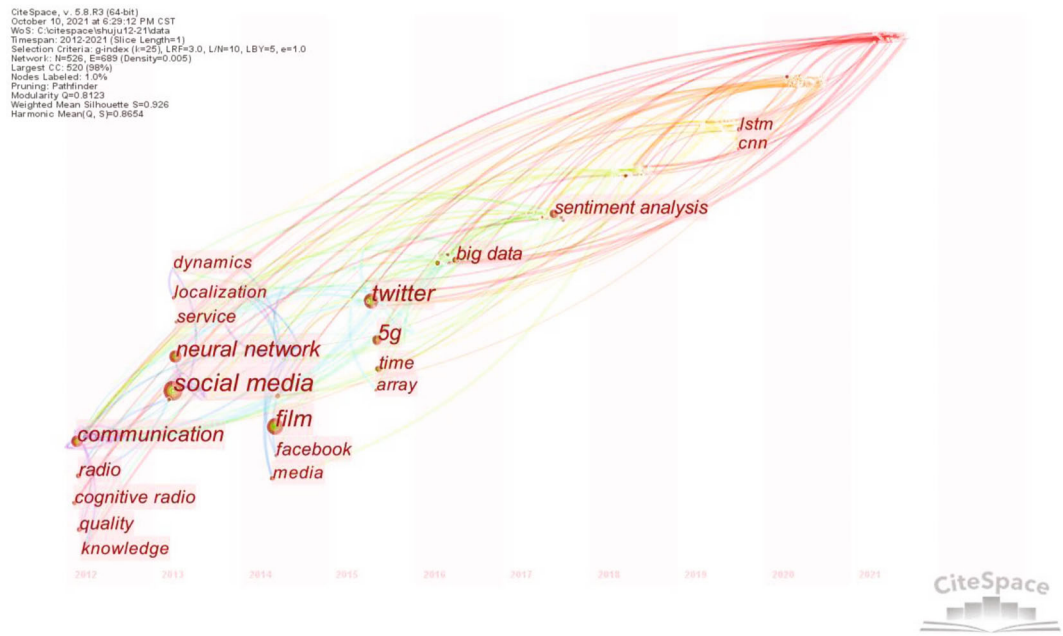


Figure 9: Keyword time zone graph.

Keywords	Year	Strength	Begin	End	2012-2021
Swarming motility	2012	13.25	2012	2017	
Cognitive radio	2012	7.3	2012	2015	
Rfid	2012	6.45	2012	2014	
Context	2012	4.72	2012	2014	
Intelligibility	2012	3.13	2012	2015	
Radio	2012	2.49	2012	2014	
Genetic algorithm	2012	3.47	2013	2015	
Game	2012	3.27	2013	2018	
Video	2012	2.69	2013	2015	
Web	2012	8.49	2014	2016	
Time	2012	6.04	2015	2017	
Complex network	2012	4.72	2016	2017	
Fiber	2012	4.18	2017	2019	
Cognitive radio network	2012	3.16	2017	2018	
Mobile	2012	3.4	2019	2021	

Figure 10: Ranking of mutation keywords.

4.5.3. Keyword Time Zone Analysis. CiteSpace software analyses the time zone distribution of high-frequency keywords (see Figure 9). The topic path map can reflect the cooccurrence relationship of keywords in a specific time range and the development context and research trend of a certain field. Figure 9 shows that the location of each circle represents the earliest publication year of an article in which the keyword is located, the thickness of the growth ring represents the heat of the research, the line between keywords represents the cooccurrence relationship, and the trend of the line represents the development of this research direction. Figure 9 shows a trend in the development of high-frequency keywords in intelligent media in the past decade from 2012 to 2021. The keyword with high frequency in 2013 is social media, which has continued to date.

4.5.4. Keyword Burst Analysis. Emergent words are key terms in which keywords suddenly increase or their use frequency has increased significantly in a certain period. Mutation words can show some cutting-edge research fields of intelligent media research. Figure 10 shows that a total of 15 mutation words are obtained. Amongst them, the strongest is swarming mobility, with mutation intensity of 13.25. The first year of its emergence was 2012 and ended in 2017. Swarming motility has a longer time of existence, which was also the area of interest in the early years and

Table 5: Citation frequency and starting year of highly cited references.

No.	Paper	Frequency	Start year
1	LeCun Y, 2015, NATURE, V521, P436, DOI 10.1038/nature14539	143	2015
2	Devlin J, 2019, P NAACL HLT, V0, P0, DOI 10.18653/v1/N19-1423	120	2019
3	Kingma DP, 2017, ICLR, V0, P0	115	2017
4	Kaiming He, 2016, 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), V0, P770, DOI 10.1109/CVPR.2016.90	92	2016
5	Krizhevsky A, 2017, COMMUN ACM, V60, P84, DOI 10.1145/3065386	81	2017
6	Goodfellow I, 2016, ADAPT COMPUT MACH LE, V0, P1	60	2016
7	He KM, 2016, PROC CVPR IEEE, V0, P770, DOI 10.1109/CVPR.2016.90	57	2016
8	Cambria E, 2016, IEEE INTELL SYST, V31, P102, DOI 10.1109/MIS.2016.31	56	2016
9	Pennington J, 2014, P EMNLP C, V0, P1532, DOI DOI 10.3115/V1/D14-1162	54	2014
10	Bojanowski P, 2017, T ASSOC COMPUT LING, V5, P135	53	2017
11	Russakovsky O, 2015, INT J COMPUT VISION, V115, P211, DOI 10.1007/s11263-015-0816-y	46	2015
12	Vosoughi S, 2018, SCIENCE, V359, P1146, DOI 10.1126/science.aap9559	45	2018
13	Simonyan K, 2014, ARXIV14091556, V0, P0, DOI 10.1109/CVPR.2015.7298594]	45	2014
14	Mnih V, 2015, NATURE, V518, P529, DOI 10.1038/nature14236	45	2015
15	Schmidhuber J, 2015, NEURAL NETWORKS, V61, P85, DOI 10.1016/j.neunet.2014.09.003	43	2015
16	Kai Shu, 2017, ACM SIGKDD Explorations Newsletter, V19, P22, DOI 10.1145/3137597.3137600	42	2017
17	He K, 2016, P IEEE C COMPUTER VI, V0, P770, DOI DOI 10.1109/CVPR.2016.90	42	2016
18	Ma J, 2016, P 25 INT JOINT C ART, V0, P3818	41	2016
19	Wu QQ, 2019, IEEE T WIREL COMMUN, V18, P5394, DOI 10.1109/TWC.2019.2936025	40	2019

Figure 11: Cooccurrence graph of highly cited documents (citation frequency of at least 40).

lasted a long time. With the advent of the 5G era, mobile has been a theme of recent interest since 2019.

4.6. Document Cocitation Analysis. Cocitation analysis of references primarily is aimed at studying key documents and research hotspots in this field. Cocitation clustering reflects the popular topics and research directions of common concern amongst clustered documents. The cocitation time zone graph can reflect the evolution process of this field in the time dimension.

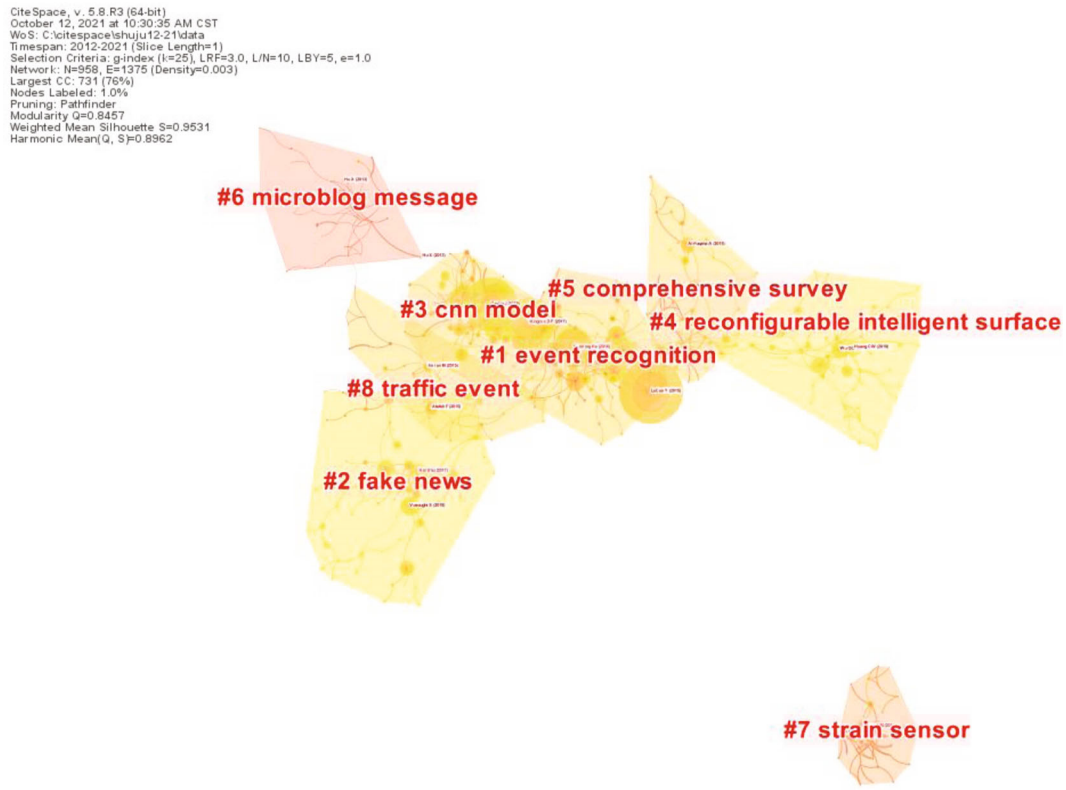


Figure 12: Clustering graph of highly cited documents.

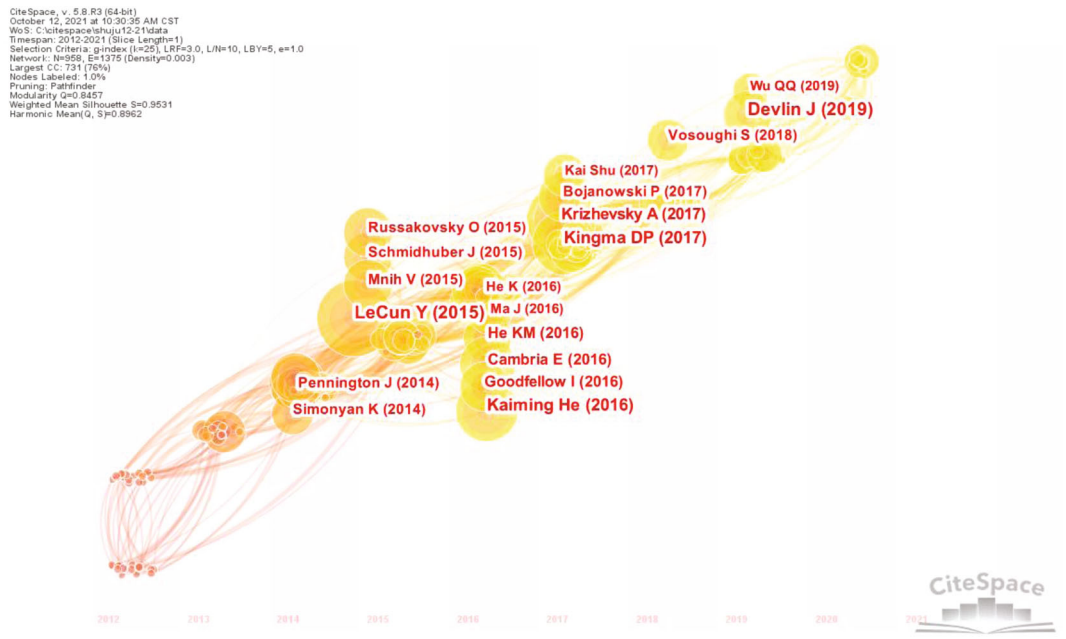


Figure 13: Time zone graph of highly cited documents.

CiteSpace conducted cooccurrence analysis and cluster analysis on literature cocitation (see Table 5 and Figure 11). References with citation frequency of at least 40 are concentrated in 2015–2016 from the time of publication, which is relatively early. In terms of content, it is mainly aimed at artistic intelligence, news, Cable News Network (CNN), and other fields. These cited references lay the foundation for the subsequent study of intelligent media.

Cluster analysis was conducted on the cited literature. The number of clusters in this cluster was set to 9, of which

the No. 0 cluster point was electronic skin, which is inconsistent with the theme of "intelligent media" studied in this research. Therefore, it was removed. Lastly, eight cluster points on intelligent media references were obtained. Subject words of the main research directions in the recent 10 years are event recognition, fake news, CNN model, reconfigurable intelligent surface, comprehensive survey, microblog message, strain sensor, and traffic event (see Figure 12).

According to CiteSpace, highly cited literature was analysed. If there is less literature in a time zone (see Figure 13), then there are less influential achievements in that time zone. By contrast, if there is more literature in a time zone, then numerous influential achievements have accumulated in that time zone. The connection of nodes between time zones indicates the inheritance of the research. Figure 13 shows that in the past 10 years, there were numerous literature in the three-year time zone from 2015 to 2017, which accumulated many influential achievements in this field and laid a certain research foundation for the subsequent research in intelligent media. Given that highly cited literature also involves time delay, papers in recent years have not formed a highly cited effect. With the passage of time, the results accumulated in recent years will continue for follow-up research, and high-quality papers will also be highly cited.

5. Conclusion

This research conducted a bibliometric analysis of SCIE retrieved journals from 2012 to 2021 and provided beneficial insights into document types, distribution of countries, regions and institutions, geographical distribution of authors, co-author network, and content analysis. The following conclusions can be drawn according to the development and analysis of research:

(1) In the past decade, the number of papers in intelligent media has been increasing annually. Since 2017, the speed has accelerated. From the perspective of countries that issue papers, the People's Republic of China and the USA have high proportions of documents in this field. Presently, the research of various countries in intelligent media is closely connected

(2) From the perspective of institutions that issue papers, CAS has published the most number of papers. However, in intelligent media research, cooperation amongst institutions is not close because intelligent media is a research topic that has only emerged in recent years. From the perspective of authors, researchers in intelligent media are relatively scattered and have yet to form a high-yielding discipline writer group

(3) In intelligent media research, a high-frequency word in the last decade is social media, which shows that intelligent media is closely related to social media. Sharing of social media is essential to meet social needs but not necessarily the needs of the target object. Intelligent media uses algorithm push, tap the needs of the audience, and push the information that meets the needs. Social media will eventually evolve into intelligent media

(4) Judging from the increasing number of papers annually, the general trend is to promote the entry of intelligent technology to the media industry. The key is how to integrate with the existing media business forms or develop new media forms and platforms. According to the law of technological innovation, the best organisational form conducive to the promotion of innovative business is to establish new institutions, adopt new technologies and operate new businesses

(5) Presently, the focus should be research on social issues caused by intelligent media. The privacy, addiction, and risk caused by intelligent media are also important factors affecting the development of intelligent media. Only by studying in advance and preparing scientifically can we prevent problems before they happen

The research on intelligent media presents an interdisciplinary trend. The academic field of intelligent media comprises computer and communication scholars, as well as numerous scholars from communication, journalism, management, and philosophy. With the help of their professional knowledge and research perspective, they discuss the operation mechanism of algorithms, legal supervision, ethical risks and other issues. This aspect shows that systematic problems brought by intelligent media to information communication and even the entire society go beyond the scope of a single discipline and require the joint participation of multiple disciplines.

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 no conflicts of interest.

Acknowledgments

We thank Steven Pagel of Philadelphia, USA, for his help in writing this paper. This research was funded by the National Social Science Fund Art Project (17BC059).

References

[1] M. E. Pollack, "Intelligent technology for an aging population: the use of AI to assist elders with cognitive impairment," *AI Magazine*, vol. 26, no. 2, pp. 9–24, 2005.

[2] C. Brooks, J. Greer, and C. Gutwin, "The data-assisted approach to building intelligent technology-enhanced learning environments," in *Learning analytics*, pp. 123–156, Springer, New York, NY, 2014.

[3] N. Marres, "What if nothing happens? Street trials of intelligent cars as experiments in participation," in *TechnoScienceSociety: Technological Reconfigurations of Science and Society*, vol. 30, pp. 111–130, Springer, Cham, 2020.

[4] C. Jin, T. Wang, X. Li et al., "A transformer generative adversarial network for multi-track music generation," *CAAI Transactions on Intelligence Technology*, vol. 2021, pp. 1–12, 2021.

[5] J. Zhang, S. O. Williams, and H. Wang, "Intelligent computing system based on pattern recognition and data mining algorithms," *Sustainable Computing: Informatics and Systems*, vol. 20, pp. 192–202, 2018.

[6] Y. Liu, R. Bao, J. Tao, J. Li, M. Dong, and C. Pan, "Recent progress in tactile sensors and their applications in intelligent systems," *Science Bulletin*, vol. 65, no. 1, pp. 70–88, 2020.

[7] G. J. Cheng, L. T. Liu, X. J. Qiang, and Y. Liu, "Industry 4.0 development and application of intelligent manufacturing," in *International Conference on Information System and Artificial Intelligence (ISAI)*, pp. 407–410, Hong Kong, China, 2016.

[8] Y. Xie, Y. Yin, W. Xue, H. Shi, and D. Chong, "Intelligent supply chain performance measurement in Industry 4.0," *Systems Research and Behavioral Science*, vol. 37, no. 4, pp. 711–718, 2020.

[9] H. C. Sang, S. Kang, and Y. J. Jeon, "Personalized recommendation system based on product specification values," *Expert Systems with Applications*, vol. 31, no. 3, pp. 607–616, 2006.

[10] Z. Cui, X. Xu, F. Xue et al., "Personalized recommendation system based on collaborative filtering for IoT scenarios," *IEEE Transactions on Services Computing*, vol. 13, no. 4, pp. 685–695, 2020.

[11] Y. Tian, B. Zheng, Y. Wang, Y. Zhang, and Q. Wu, "College library personalized recommendation system based on hybrid recommendation algorithm," in *Procedia CIRP*, vol. 83, pp. 490–494, Elsevier, 2019.

[12] M. Yan, H. Yuan, Z. Li, Q. Lin, and J. Li, "Energy savings of wireless communication networks based on mobile user environmental prediction," *Journal of Environmental Protection and Ecology*, vol. 22, no. 1, pp. 206–217, 2021.

[13] N. Thurman, J. Moeller, N. Helberger, and D. Trilling, *My friends, editors, algorithms, and I: examining audience attitudes to news selection*, vol. 7, no. 4, 2019Social Science Electronic Publishing, 2019.

[14] P. Lenhart and D. Herzog, *Combining content-based and collaborative filtering for personalized sports news recommendations*, CBRecSys@ RecSys, 2016.

[15] J. Möller, D. Trilling, N. Helberger, and B. van Es, "Do not blame it on the algorithm: an empirical assessment of multiple recommender systems and their impact on content diversity," *Information, Communication & Society*, vol. 21, no. 7, pp. 959–977, 2018.

[16] J. Huenteler, T. S. Schmidt, J. Ossenbrink, and V. H. Hoffmann, "Technology life-cycles in the energy sector – technological characteristics and the role of deployment for innovation," *Technological Forecasting and Social Change*, vol. 104, pp. 102–121, 2016.

[17] N. Diakopoulos, "Algorithmic accountability," *Digital Journalism*, vol. 3, no. 3, pp. 398–415, 2015.

[18] A. Hanelt, R. Bohnsack, D. Marz, and C. Antunes Marante, "A systematic review of the literature on digital transformation: insights and implications for strategy and organizational change," *Journal of Management Studies*, vol. 58, no. 5, pp. 1159–1197, 2021.

[19] S. Vosoughi, D. Roy, and S. Aral, "The spread of true and false news online," *Science*, vol. 359, no. 6380, pp. 1146–1151, 2018.

[20] C. J. Vargo, L. Guo, and M. A. Amazeen, "The agenda-setting power of fake news: a big data analysis of the online media landscape from 2014 to 2016," *New Media & Society*, vol. 20, no. 5, pp. 2028–2049, 2018.

[21] K. Sharma, F. Qian, H. Jiang, N. Ruchansky, M. Zhang, and Y. Liu, "Combating fake news," *ACM Transactions on Intelligent Systems and Technology*, vol. 10, no. 3, pp. 1–42, 2019.

[22] E. A. Pedersen, L. F. Loft, S. U. Jacobsen, B. Søborg, and J. Bigaard, "Strategic health communication on social media: Insights from a Danish social media campaign to address HPV vaccination hesitancy," *Vaccine*, vol. 38, no. 31, pp. 4909–4915, 2020.

[23] M. Wieringa, "What to account for when accounting for algorithms: a systematic literature review on algorithmic accountability," in *ACM Conference on Fairness, Accountability, and Transparency (FAT)*, pp. 1–18, Barcelona, Spain, 2020.

[24] Y. Li, C. Su, J. Jia, Z. Xu, and R. Tian, "Analysis of development status of world artificial intelligence based on scientific measurement," *Computer Science*, vol. 44, no. 4, pp. 183–187, 2017.

[25] Y. Qian, Y. Liu, and Q. Z. Sheng, "Understanding hierarchical structural evolution in a scientific discipline: a case study of artificial intelligence," *Journal of Informetrics*, vol. 14, no. 3, article 101047, 2020.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 8927822, 13 pages
https://doi.org/10.1155/2021/8927822

Research Article

A Robust Invariant Local Feature Matching Method for Changing Scenes

Di Wang, Hongying Zhang, and Yanhua Shao

School of Information Engineering, Southwest University of Science and Technology, 621010 Mianyang, China

Correspondence should be addressed to Hongying Zhang; zhywyd@163.com

Received 14 October 2021; Accepted 6 December 2021; Published 28 December 2021

Academic Editor: Ming Yan

Copyright © 2021 Di 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.

The precise evaluation of camera position and orientation is a momentous procedure of most machine vision tasks, especially visual localization. Aiming at the shortcomings of local features of dealing with changing scenes and the problem of realizing a robust end-to-end network that worked from feature detection to matching, an invariant local feature matching method for changing scene image pairs is proposed, which is a network that integrates feature detection, descriptor constitution, and feature matching. In the feature point detection and descriptor construction stage, joint training is carried out based on a neural network. In the feature point extraction and descriptor construction stage, joint training is carried out based on a neural network. To obtain local features with solid robustness to viewpoint and illumination changes, the Vector of Locally Aggregated Descriptors based on Neural Network (NetVLAD) module is introduced to compute the degree of correlation of description vectors from one image to another counterpart. Then, to enhance the relationship between relevant local features of image pairs, the attentional graph neural network (AGNN) is introduced, and the Sinkhorn algorithm is used to match them; finally, the local feature matching results between image pairs are output. The experimental results show that, compared with the existed algorithms, the proposed method enhances the robustness of local features of varying sights, performs better in terms of homography estimation, matching precision, and recall, and when meeting the requirements of the visual localization system to the environment, the end-to-end network tasks can be realized.

1. Introduction

The excellent matching performance between local features of changing scene images can ensure the stability of adjacent frame matching in visual localization. Visual localization is also the key technology for the robot to realize autonomous movement. Moreover, visual localization is an essential technology in route planning. Route planning has been applied in many fields [1]. The determination of camera pose has always been the focus in the field of machine vision. It can ensure the better operation of follow-up work, such as tracking and loop closure detection. Feature detection, descriptor constitution, and local feature matching are necessary supporting means to deal with most machine vision problems. At present, standard traditional and improved feature extraction methods [2–5] have been widely used in the visual localization system, the Markov model [6] has also been widely used in the traditional route planning issues. Scale-invariant feature transform (SIFT) [2] constructs the Gaussian pyramid of the input image, queries the extreme points, and determines the feature points' position. Finally, it calculates the domain gradient with the feature points as the circle's center and gives the SIFT feature point direction to obtain the descriptor. Speed-Up Robust Features (SURF) [4] improves the SIFT algorithm, converts the input image into an integral image, transforms the image with Hessian matrix, and finally extracts local feature points through nonmaximum suppression to solve the problems of high computational complexity and time-costs of SIFT. As a feature detection algorithm, the document [3] as well as the descriptor construction algorithm proposed in document [5] is usually used together. Compared with SIFT and SURF, it has a faster speed. Oriented Fast and Rotated Brief (ORB) [7] improves FAST and BRIEF algorithms. Compared with FAST, this algorithm only uses fewer pixel values in feature point extraction and solves its disadvantage of

nondirectionality. Compared with BRIEF, ORB uses a statistical learning method to select the set of feature points, which strengthens the discrimination ability but reduces the computational efficiency. The research on feature point detection and descriptor construction mainly focuses on finding a unique location that can reliably estimate its scale, rotation, and illumination change. Feature points are usually the premise of processing machine vision tasks. However, even if conventional feature detection ways, such as SIFT, SURF, and other algorithms, have been widely used in the field of machine vision, they cannot reflect semantic features and lack strong robustness to scene viewpoint, scale, and weather illumination changes. The introduction of the convolutional neural network can effectively solve different computer tasks [8]. Computers benefit from the introduction of deep learning and can obtain robust models against changes in the natural environment under the training of large amounts of data.

It is very important to describe the attitude transformation between adjacent frames. DeTone et al. [9] applied depth learning to feature point detection and transformation estimation and proposed a paper on using depth neural network to obtain homography matrix. The Learned Invariant Feature Transform (LIFT) algorithm proposed by Trulls et al. [10] uses the spatial deformation combiner to combine the feature extraction algorithm [11], feature direction algorithm [12], and descriptor construction algorithm [13] based on neural network. The LIFT leads a fresh deep network structure into the algorithm, but it suffers the defect of carrying out an end-to-end learning network. DeTone et al. [14] put forward the Superpoint, an algorithm advanced on [15], which uses the Siamese neural network to train the joint model of local features, to complete the feature task by a single neural network. The descriptor constitution phase of the united training network uses the network architecture proposed in [16], to promote the performance of the feature point detector, and Superpoint also puts forward homographic adaptation, which is mainly responsible for operating on a single image and extracting two-dimensional feature points. Compared with the effect of the traditional feature point detector, it has better detection performance in the case of image noise. To improve the generalization and robustness of the extracted feature points, TGD is combined with a depth neural network to detect the confidence score of each extracted feature point to screen the wrongly extracted feature points through the confidence score. The descriptors built by Superpoint cannot adaptively adjust the invariance for changing scenes, to strengthen the robustness of local features to natural change scenes, and Pautrat et al. [17] proposed to use NetVLAD module [18] to integrate and compute the output of the local feature construction stage. NetVLAD is suitable for weakly supervised CNN architecture of location recognition. Its introduction can make the descriptor construction stage have strong compatibility combined with most feature point extraction algorithms. A good feature matching algorithm can reduce much computational burden for the follow-up work of machine vision. To strengthen the linkage of local correlative features from an image pair to another counterpart, Sarlin et al. [19] put forward a matching method being founded on Attention Graph Neural Network (AGNN) [20, 21].

The above algorithms are separately used in three single stages: feature detection, descriptor constitution, and local feature matching. However, for visual localization, it is essential to propose an end-to-end network that integrates phases of local feature detection and matching to facilitate the later realization of good machine vision work, such as visual odometry and loop closure detection. Being founded on this, in this work, a robust invariant local feature matching method for changing scenes is put forward, which unites the invariant local feature matching method of AGNN and the Sinkhorn algorithm [22]. Furthermore, this paper proposes a united network of feature detection and descriptor constitution. Due to the lack of datasets of unique work, self-supervised learning can effectively solve this problem [23]; therefore, this paper uses the method of self-supervised training network in the feature location extraction stage. The NetVLAD module is employed to compute and fuse the correlation of multiple descriptors after homography transformation, which makes the local features output in the later stage have robust illumination invariance and viewpoint invariance. AGNN plays a role in strengthening the relevance of local features in the matching stage, which is enhanced by iterative updating, and the added local features are the output of the above joint training network. Finally, the local features linkage strengthened by AGNN is solved by the Sinkhorn algorithm. Meanwhile, to ensure the robustness of the model to natural scenes viewpoint and illumination varieties, a completed end-to-end network is applied to construct and go together local features.

The main work of this paper includes the following aspects: (1) an end-to-end matching network that integrates local feature constitution and matching is put forward. (2) In the descriptor constitution phase, the NetVLAD module is introduced to enhance the descriptor performance. (3) In the matching stage, AGNN and Sinkhorn algorithms are used to gain the optimal matching matrix. In addition, the algorithm proposed in this article has a better performance than the existing algorithm in the public dataset test.

2. Invariant Local Features Matching Network Architecture

2.1. Network Framework. This work takes the integral channel image pair as input and the direct matching result as the output, which describes a single end-to-end local feature matching method based on a deep neural network. The whole system integrates feature detection, descriptor constitution, and local feature matching.

The fundamental network architecture is indicated in Figure 1, which contains a total of three phases. The initial stage is calibrating the natural scene dataset, and the second stage is the united training network of local features. To begin with, an elementary feature point extractor is achieved, which is trained by a labeled dataset; this dataset is obtained by the joint action of simple shape dataset, real scene dataset, and pretraining feature extractor. Secondly, the united

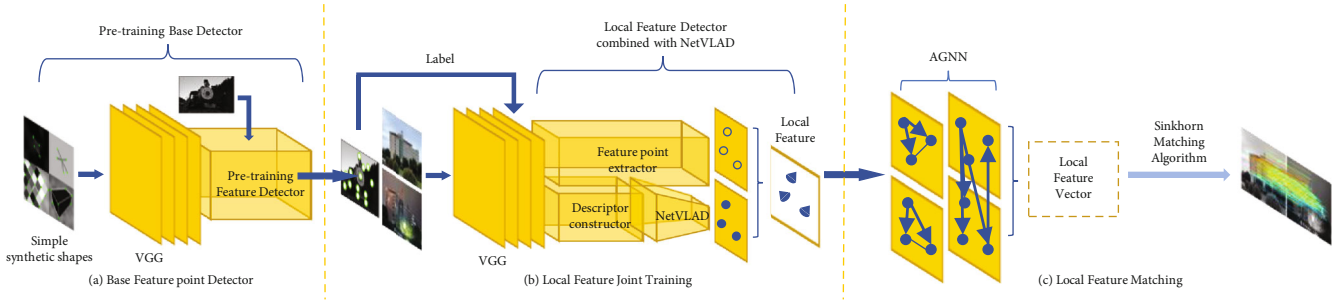


Figure 1: The fundamental network frame of local feature matching. It consists of three parts.

training network outputs the local feature vector composed of the feature position and descriptor vector, and the descriptor vector is integrated and computed by the NetVLAD module in the descriptor constitution stage. Going together of local feature vectors is the third stage of this work. The AGNN layer iteratively updates the local features that output in the second phase and then a fresh descriptor vector with a strong connection is output; eventually, the newly obtained descriptor vector is matched by the Sinkhorn algorithm, which outputs the final matching result of the image pair. As shown in Table 1, compared with the former, the algorithm proposed in this paper uses the deep neural network framework to construct strong, robust local features, enhances the relationship between local features by AGNN, and constructs the process of input to matching with complete channel image pairs. The following will be divided into multiple sections to describe in detail the three parts of the mainframe and the loss function of the network.

2.2. Base Feature Detector. Acquiring tagged datasets is one of the problematic points of leading depth neural networks into machine vision localization tasks. This paper uses a basic feature detector to obtain abundant natural scene image feature point labels. The network structure diagram of this part corresponds to Figure 1(a), and the specific details are shown in Figure 2. The central network framework of this part of the network uses VGG [24], and the convolution kernel of every layer in the network is 3×3. The last layer outputs the feature map of 65 channels that the dimension size is 15×20. An 8×8 area in a simple input image is used as a feature represented by each channel and then the multichannel feature map through the reconstruction upsampling layer that used 1×1 convolution kernel, which makes the network obtain the feature point images that accompany the identical output and input dimension. At the start of the training, firstly, the ordinary shape synthesis generator [15] of the Python file is used to generate a 120×160-dimensional dataset that is the data to be input, which contains ordinary synthetic forms such as polygons, quadrilaterals, stars, and chessboards. After obtaining the base feature point detector, the self-supervised learning means produces the feature point label for the unlabeled natural sight image.

To strengthen the adaptability of the model to the natural scene data, firstly, the input real sight data is handled by multiple homography changes like rotation and scaling, and then the simple graph data marked with corner features are used as the pretext for self-supervised learning. After the above two steps, the model has homography invariance for the input data. Finally, the image data with feature points output in this stage is used as the label data in the next stage. The labeling process of natural scene data is shown in Figure 3. In this stage:

$$x = H^{-1} f_\theta(H(I)). \tag{1}$$

x is the elementary corner vector, $f_\theta(.)$ is the fundamental feature extractor model, H is the homography transmutation matrix, and I indicates the input image. Therefore, the following formula can be used to represent the feature detection model:

$$\widehat{F}(I; \mathrm{f}_\theta) = \frac{1}{N_h} \sum_{i=1}^{N_h} x. \tag{2}$$

The frequency of use of homography transformations is represented by N_h. The joint training network of local features construction is described in detail below.

The original image undergoes a variety of homography matrix changes to obtain various images (N_h in Eq. (2) represents the number of transformations). Then, perform feature point extraction on different images (the extraction process is represented by $f_\theta(.)$ in formula (1)). Finally, the different homography images are superimposed and calculated (the superimposition process is expressed by formula (2)).

2.3. Local Feature United Training Network. Although the descriptors put forward in some prior researchs have invariant characteristics, their invariant characteristics remain in the fixed descriptors, making the descriptors unable to adapt to various change scenarios and causing redundancy of descriptors invariant characteristics. For example, if the descriptor only has the characteristic of illumination invariance, its matching effect on the image pair with changing viewpoint is not as good as the matching effect on the image pair with changing illumination. Similarly, the matching result of descriptors with single viewpoint invariance on a couple of illumination converted images is not as fine as that of a couple of viewpoints converted images. To deal with this problem effectively, this work strengthens the robustness of

Table 1: Related work comparison.

	Full image input	Feature points	Descriptors	Matching	Single network
Ours	√	√	√	√	√
SuperGlue [19]				√	
Superpoint [14]	√	√	√		√
UCN [16]	√		√		√
LIFT [10]		√	√		
TILDE [11]		√			√
DeepDesc [13]			√		√
BRIEF [5]			√		
FAST [3]		√			
SURF [4]		√	√		
SIFT [2]		√	√		
ORB [7]		√	√		

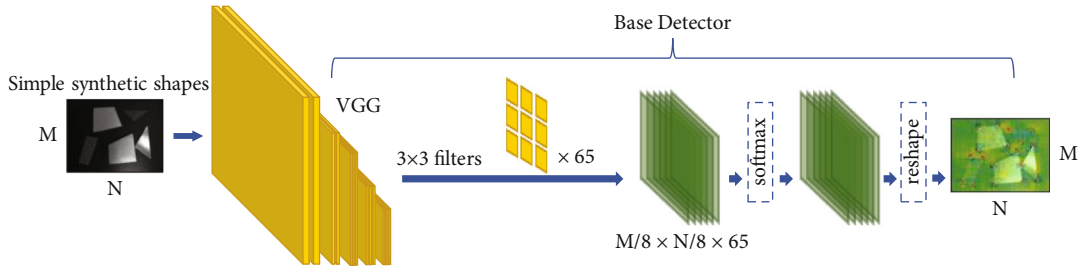


Figure 2: Elementary feature point detection phase.

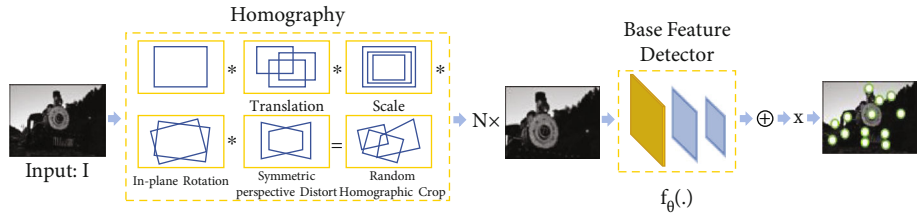


Figure 3: Natural scene data annotation.

descriptors to natural changing scenes. By using the NetVLAD module in the descriptor constitution stage, the diagram is shown in Figure 1(b).

As illustrated in Section 2.1, the natural scene data marked with feature points is used as the label data of the local feature united training network. As indicated in Figure 4, the 8-layer encoder based on VGG is used as the shared network of feature point extractor and descriptor builder to reduce the dimension of input image data. The number of convolution cores in each layer of the shared network is 64-64-64-64-128-128-128-128, and the dimension of convolution cores is 3×3. Every two layers are linked to the max-pooling layer, and the network uses ReLU nonlinear activation function. After the shared network reduces the input data dimension, the input data outputs the position vector p of feature point through the feature point detection model.

In the descriptor construction stage, the NetVLAD module, as indicated in Figure 5, is introduced. The introduction of this module can confirm the descriptor invariant traits for various changing scenes. Its advantages are shown in the test results of the experiment in Section 4.1. Supposing a collection of $H \times W \times D$ feature pictures as the input, which through the preconvolution network trained by multiscene transformation data, it outputs four groups of $H/8 \times W/8 \times 128$-dimensional local characteristic graphs that constitute illumination and viewpoint variations. The four semidense descriptors are, respectively, used as the input x of a NetVLAD module, and finally, the four descriptor vectors output by the NetVLAD module are integrated to obtain the final descriptor vector with strong robustness. Set the number of local descriptor inputs of NetVLAD module to 8, and then cluster center $D \times K$ obtains the elements of image description matrix v through clustering algorithm. Eventually, normalizing the description matrix v is implemented to obtain the descriptor vector represented by d.

2.4. Matching Layer Integrated with AGNN and Sinkhorn Algorithm. A large number of ambient intervention elements

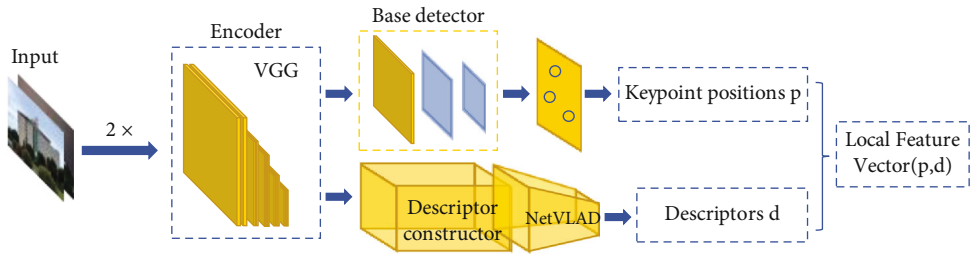


Figure 4: United training network with the insertion of NetVLAD.

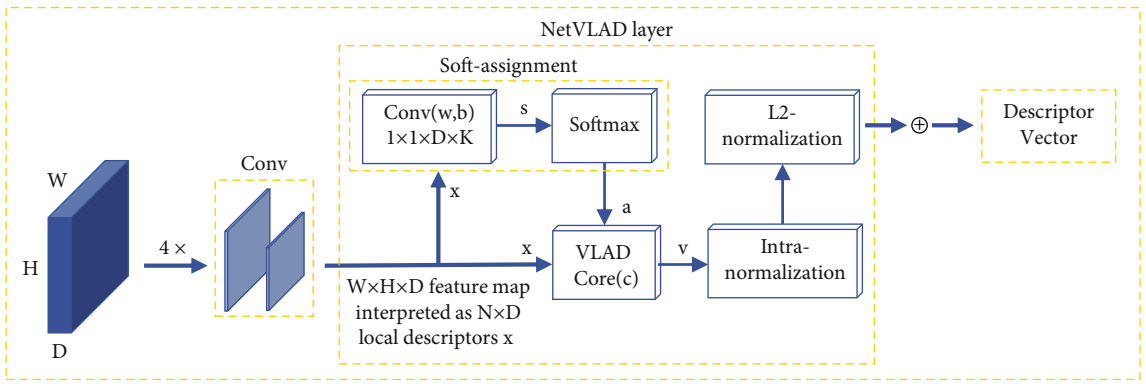


Figure 5: NetVLAD module.

in the real sight will lose the matching information between local features of the image pair, which will affect the subsequent machine vision work; as shown in Figure 1(c), to improve the above challenge, this paper infuses AGNN to achieve the target that strengthen the local feature information of the image itself and image pairs by multiple iterations. Iteration aims to strengthen the connection between local features of an image pair to obtain a better matching effect. As shown in Figure 6, the input of AGNN includes the position vector p and descriptor vector d of local features, and inputs p_i and d_i are mapped into a single vector through the feature point encoder. A five-layer multilayer perceptron encrypts p_i. Then, the local feature vector outputs 256-dimensional data under the action of attention, which is expressed as descriptor vector d, which is integrated to obtain the local feature vector x_i.

$$x_i = d_i + MLP_{\text{enc}}(p_i). \tag{3}$$

In principle, the upper equation links the descriptor and the corresponding feature location, which enables the following attention methodology of encrypting proceeding to ruminate the resemblances between local features entirely. The local feature vector received by the prework transforms to a mapping pattern that contains information about the query, keys, and values, where query and key are calculated by softmax, and then output the weight between adjacent local features, which is multiplied by another element. The upper operation consequences output the refreshed value delta after MLP. Then, to strengthen the linkage between local feature vectors, renewing local features iteratively by using delta.

$$\text{delta} = \text{MLP}\left(\left[{}^{(l)}x_i^{A_{\text{or}}B} \| m_{\varepsilon\longrightarrow i}\right]\right). \tag{4}$$

The refreshed local feature vector can be indicated as

$${}^{(l+1)}x_i^{A_{\text{or}}B} = {}^{(l)}x_i^{A_{\text{or}}B} + \text{delta}. \tag{5}$$

The node info in AGNN is presented by the $m_{\varepsilon\longrightarrow i}$ in the delta treated as the polymerized data. Particular local features are used by AGNN as nodes to accumulate rest nodes surrounding it, indicated as

$$m_{\varepsilon\longrightarrow i} = \sum_{j:(i,j)\in\varepsilon} \alpha_{ij} v_j. \tag{6}$$

The correlation of associated feature points is indicated by attention weight α_{ij}, and the mapping value from local features to graph nodes is indicated by v_j. In Figure 7, to strengthen the matching behavior between local feature vectors of different images, the correlation is weighted by cyclic iteration. To refresh the local features, using AGNN with self and crossattention mechanism is carried on, and the number of iterations is set to 9 in this paper. If A and B are input image pairs, their comparative graphs are Q and S, respectively, and W indicates the similitude among key and query. The association among the local features of the inputs and the node through the AGNN layer is

$$q_i = W^{(L)} x_i^{Q_{\text{or}}S} + b, \tag{7}$$

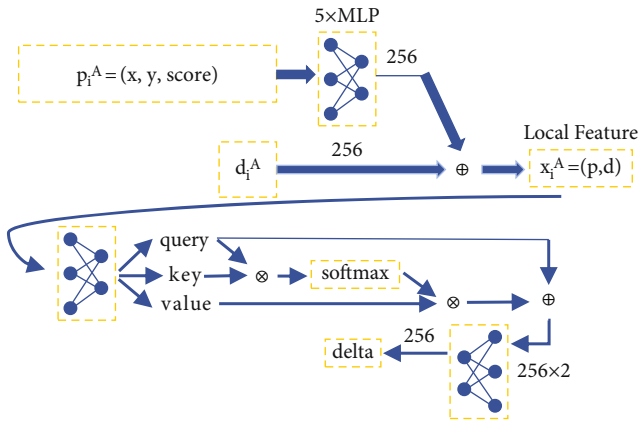


Figure 6: Local feature renewing integrated with attention mechanism.

which obtains nodes through local features handled by the AGNN layer. The ultimate matching descriptor vector of A and B can be indicated as

$$f_i^A = W\bullet^{(L)}x_i^A + b, \forall i \in A, \tag{8}$$

$$f_i^B = W\bullet^{(L)}x_i^B + b, \forall i \in B. \tag{9}$$

In the second stage of Figure 7(b), to achieve the local feature matching degree numerical matrix, it is necessary to calculate the inner product between the local feature vectors of various images. Then, this paper uses the Sinkhorn algorithm to indicate the optimal local feature matching matrix:

$$S_{i,j} = <f_i^A, f_i^B>, \forall (i,j) \in A \times B. \tag{10}$$

To percolate the inaccurate matching points between different image pairs, a channel in the matching score matrix is broadened to reserve undiscovered matching constituents. The broadened matching score matrix is $(M+1)\times(N+1)$ dimensions, which can be indicated as

$$\bar{s}_{i,N+1} = \bar{s}_{j,M+1} = \bar{s}_{M+1,N+1} = z \in \mathbb{R}. \tag{11}$$

To achieve the final image pair matching matrix $P = \bar{P}_{1:M,1:N}$, the matching score matrix $\bar{s}_{M+1,N+1}$ performs $T = 100$ times of alternating iterative calculation including row and column normalization through the Sinkhorn matching algorithm. At the same time, to ensure that the local feature vectors have a better precise matching matrix, the Sinkhorn algorithm draws into the entropic regularization function to legitimately distribute the loss weight in AGNN. The local features in the image pair are all in the form of a matrix, and the Sinkhorn algorithm needs to solve the Wasserstein distance deformation between the two matrices. Finally, outputting the image pair matching result has been realized.

2.5. Loss Function. To facilitate the optimization of network parameter settings, it is necessary to obtain the training degree, performance, and advantages and disadvantages of the network model. The joint loss function indicated in equation (12) is the loss function employed in the whole workflow. The loss function composes a feature detection module section L_{position}, a descriptor construction section $L_{\text{descriptor}}$, and a matching section L_{matching}.

$$\text{LOSS} = L_{\text{position}}(X, Y) + L_{\text{descriptor}}(D) + L_{\text{matching}}. \tag{12}$$

The feature detection section of the loss function is revealed below:

$$L_{\text{position}}(X, Y) = \frac{1}{(H_c W_c)} \sum_{\substack{h=1 \\ w=1}}^{H_c W_c} l_p(x_{hw} : y_{hw}). \tag{13}$$

Crossentropy loss [25] is applied in this function, where X and Y are the detected feature and the tagged feature locations after the homography matrix computation, $H_c W_c$ is the dimension of the inputs following the dimension decrease of the pooling layer, and $l_p(x_{hw} : y_{hw})$ is the logarithmic odds diffusion function.

In our work, the invariant descriptor loss function put forward by paurat et al. [17] is used as the loss function in the descriptor construction stage. Assuming there is a descriptor vector D and, the inputs I^a and I^b, the loss function of local feature descriptor is indicated as

$$L_{\text{descriptor}}(D) = \frac{1}{|D|} \sum_{d \in D} l\left(I_d^a, I_d^b, \text{dist}\right). \tag{14}$$

dist is the two-norm Euclidean distance between the features of the inputs I^a and I^b, and $l(I_d^a, I_d^b, dist)$ is the loss function comprising the local feature descriptor and the Euclidean distance of the image pair. The loss function's establishment helps the model more accurately extract the local descriptors suitable for matching in the next stage of the dense descriptor mapping.

In the process of model iteration, the loss function of the matching section requires refreshing and optimization to ensure the effectiveness and optimality of the algorithm. If the matching network is trained by the natural scene dataset M, the labeled homography transformation is used to estimate the matching of the natural scene image pair. The loss function is the difference of the negative logarithmic minimization of the training image corresponding to the matching matrix P, which is indicated below:

$$L_{\text{matching}} = -\sum_{(i,j)\in M} \log P_{i,j} - \sum_{(i,j)\in A,B} \log P_{i,j}. \tag{15}$$

The real sight dataset is revealed by M, and the input image pair is revealed by (A, B).

3. Training Particulars of Network

This part will introduce the specific details of network training, including the evaluation indexes of model performance

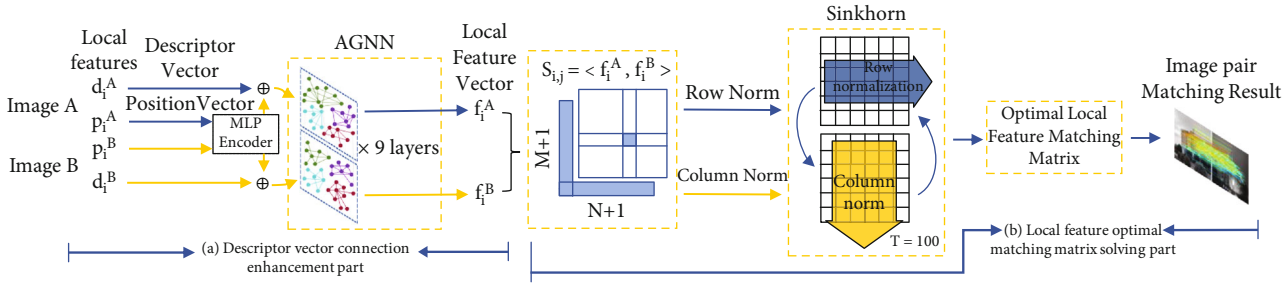


Figure 7: Local feature matching process, which contains two parts (a) and (b).

testing, the training data set used, and the specific parameter settings of the training network.

3.1. Evaluation Metrics

3.1.1. Homography Estimation. Homography estimation (HEstimation) is a critical reference to reflect the pros and cons of matching images and is widely used in machine vision fields such as SLAM. This paper refers to the local feature descriptor evaluation protocol proposed by Mikolajczyk et al. [14, 18, 26] to evaluate the matching performance of the model proposed in this paper and compare the work of the former. The homography matrix describes the mapping relationship between two planes, that is, the transformation relationship between two image feature points on a common plane. Supposing there is an image pair A and B with well-matched feature points p_1 and p_2, which fall on the plane P, and this plane satisfies the equation:

$$n^T P + d = 0. \tag{16}$$

Supposing K is the camera internal parameter matrix, t and R represent the camera motion between adjacent frames, where t is the translation vector, and R is the rotation matrix. The location of the feature point can be expressed as

$$p_2 \simeq K(RP + t) \simeq K\left(RP + t\bullet\left(-\frac{n^T P}{d}\right)\right) \simeq K\left(R - \frac{tn^T}{d}\right)K^{-1}p_1. \tag{17}$$

From the above formula, the transformation $p_2 \simeq Hp_1$ between the image coordinates p_1 and p_2 can be described, and H is the homography matrix.

Assuming that the pixel coordinates of the image pair feature points p_1 and p_2 are (u1, v1, 1) and (u2, v2, 1), H, as a 3×3 matrix, the transformation relationship is as follows:

$$\begin{pmatrix} u_2 \\ v_2 \\ 1 \end{pmatrix} \simeq \begin{pmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{pmatrix} \begin{pmatrix} u_1 \\ v_1 \\ 1 \end{pmatrix}. \tag{18}$$

Conversion relationship $u_2 = h_{11}u_1 + h_{12}v_1 + h_{13}/h_{31}u_1 + h_{32}v_1 + h_{33}$, $v_2 = h_{21}u_1 + h_{22}v_1 + h_{23}/h_{31}u_1 + h_{32}v_1 + h_{33}$, can be obtained, and the matrix form can be obtained after sorting: $A\theta = 0$, where

$$A = \begin{bmatrix} u_1 & v_1 & 1 & 0 & 0 & 0 & -u_2u_1 & -u_2v_1 & -u_2 \\ 0 & 0 & 0 & u_1 & v_1 & 1 & -v_2u_1 & -v_2v_1 & -v_2 \end{bmatrix}, \tag{19}$$

$$\theta = \begin{bmatrix} h_{11} & h_{12} & h_{13} & h_{21} & h_{22} & h_{11} & h_{23} & h_{31} & h_{32} & h_{33} \end{bmatrix}. \tag{20}$$

The homography matrix has 8 degrees of freedom, and eight linear equations are needed to solve it. In summary, it can be concluded that two linear equations can be listed for a pair of matching feature points of a set of image pairs; that is, at least four pairs of matching feature points are required to solve the homography matrix H by the formula $A\theta = 0$.

This article estimates how the homography transformation transforms the four corner points of the first frame to the next frame of an image. Assuming that the four corner points of the first frame are $\hat{c}_{1\cdots4}$ and the four corner points of the second frame are $\hat{c}_{1\cdots4}{}'$, the homography matrix H is obtained by the homography transformation. Setting the threshold ϵ, the homography evaluation formula is expressed as follows:

$$\text{HEstimation} = \frac{1}{4}\sum_{i=1}^{4} \|\hat{c}_i - c_i\|_2 \le \epsilon. \tag{21}$$

3.1.2. Matching Evaluation Metrics. Precision is the average matching accuracy, referring to the percentage of correct matches in all predicted matches. Recall is the ratio of correctly predicted matches to the total number of labeled matches. Assuming that the correct match of the label is 1, the false match is 0; the predicted correct match is 1, and the false match is 0. The true and false positive graphs are shown in Table 2.

Among them, false positives are matching deviations, and false negatives are uncorrelated matches. To improve the reliability of the matching algorithm, the occurrence of FP and FN is usually reduced as much as possible, and the occurrence probability of TP and TN should be as high as possible, enabling the algorithm to screen out

Table 2: The classification of predicted and truth matching results.

	Predicted matching result 1	Predicted matching result 0
Truth matching result 1	True positive(TP)	False positive(FP)
Truth matching result 0	False negative(FN)	True negative(TN)

false matches more accurately. The precision and recall formulas are as follows.

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}}, \tag{22}$$

$$\text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}}, \tag{23}$$

3.2. Datasets. The training aggregation of the whole network applies the COCO14 dataset [27] that contains 80,000 images, and the input dimensionality of every images are 240×320, which is used to train the shared network; and the image dataset put forward in [28] includes diverse viewpoints and illuminations; between them, the training aggregation of feature detection, descriptor constitution, and local feature matching uses COCO14 dataset, which can guarantee that the ultimate model has adaptability in real scenes. The descriptor construction phase primarily uses the Multi-Illumination dataset put forward in [28], it contains 1000 natural scene images, and each image has 25 images under different illumination. The dataset has a total of 25000 images. The constantness of descriptors to environmental changes is ensured by the image features of various viewpoints and illumination in an identical site. Table 3 shows the instance of the dataset applied in this work.

3.3. Implementation. The experimental context of this work consists of hardware devices and system environment settings. The concrete type and version are NVIDIA RTX2080Ti GPU; Intel Core CPU I7-10700 K 3.8GHz eight-core; Ubuntu18.04, Spyder (Python3.6). Moreover, the realization of the deep neural network is carried out on the PyTorch framework.

In the training phase, a simple synthetic dataset is used as the input for the iterative training of the basic feature detector. After that, the basic feature detector is trained by the real scene dataset with a dimension of 240×320, and the output of this part is applied as the label data of the shared network in the local feature united training phase. In the descriptor constitution phase of the united training network, initially, a 128-channel descriptor is output, and then a 512 channel metadescriptor is output, which is composed of four 128 dimensional descriptors. Batch size is set to 32 in the entire training procedure, Adam gradient optimization algorithm is applied, and the initial learning rate is set to 0.001. The number of training epochs was 20. In this paper, the learning rate is set to change with the number of training times. If the model tends to be stable in 10 epochs, the frequency of training is reduced by half, and then the training is discontinued after 20 epochs. Through the above methods, the purpose of saving calculation cost and improving model training efficiency is achieved. After completing training, the input image pairs pass through the local feature constitution layer of the united network, which achieves the local features applied as input in the matching phase. The local features obtained above are iterated 9 times by AGNN, which outputs the updated local feature vector; subsequently, the new local feature vector is internally multiplied to achieve the matching numerical matrix. The matching numerical matrix is then computed by 100 times of alternating row and column normalization of the Sinkhorn algorithm. The ultimate matching consequence was obtained after achieving the optimized matching matrix.

4. Experiments and Discussion

This part will present the concrete experimental consequences. The experiment includes the descriptor and the matching algorithm tests. Compared with the existing algorithm, the experiment effectively verifies the performance of the local feature and matching algorithm.

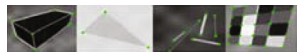
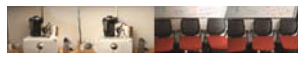
4.1. Descriptor Test Experiment. This work has tested many experiments on self-collected image pairs and open datasets to testify the availability of the local features put forward in this work.

4.1.1. Effect Exhibition of Feature Descriptor. To intuitively describe the performance effects of feature descriptors with invariant environment characteristics, this article first made a set of visual displays as shown in Table 4, comparing SIFT, Superpoint, and the algorithm proposed in this article, and collected two sets of the viewpoint and the rotation of the 1920×1200-dimensional image pair of the experimental building are set to collect up to 200 feature points. The nearest neighbor (NN) matching algorithm and the RANSAC mismatch optimization algorithm [29] are applied. The experimental results are shown in Table 4, and the method proposed in this paper (with invariance) has a better matching performance result than the feature descriptor without environment invariance.

4.1.2. Validation of the Invariance of Illumination and Viewpoint. The purpose of this experiment is to effectively verify that the descriptor constructed by the algorithm in this work has a better behavior consequence in realistic scenarios than the former, and this work applies the dataset HPatches [30], which is exclusively applied to assess descriptors and the model behavior. The 57 scene image data of this dataset all contain environmental transformation factors, containing the HPatches Illum dataset with illumination change and the HPatches View dataset with viewpoint change. Both datasets are applied to evaluate the homography, matching precision, and the recall of the model.

In the experiments displayed in Figures 8(a) and 8(b), the two groups were tested on the illumination and viewpoint change datasets of HPatches, respectively. The test results indicate the influence of the adaptive invariance

Table 3: Datasets examples.

	Dataset diagram
Sythetic Dataset [15]	
COCO2014 [27]	
Multi-Illumination Dataset [28]	

characteristics of descriptors on the matching results. In the experiment, 500 is set as the maximum number of feature points extracted from a single image, and the threshold is tune-up to compute the relevant matching precision. The threshold set in this experiment is the pixel correctness threshold, and its purpose is to compare the pixels contained in the feature points with the threshold to make a matching judgment. Different threshold settings will lead to different matching judgment decisions between the corresponding feature points of the image pair, and the comparison of matching precision under different thresholds can better illustrate the performance advantages of the algorithm.

It can be concluded from Figure 8(a) that descriptors with specified characteristics (Illum invar, rot var) have higher matching precision than other descriptors in experiments on the HPatches Illum dataset. Moreover, the experimental results of Figure 8(b) are just opposite to those of (a). In general, the above two experiments prove the availability of the environmental invariant characters applied in the descriptor.

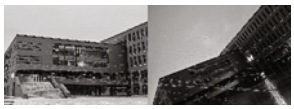
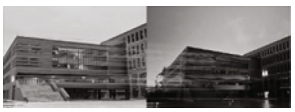
4.1.3. Feature Descriptor Algorithm Comparison. The experiment shown in Figure 8(c) compares our put forward method and other existing algorithms on the HPatches View dataset. It can be seen that our proposed method has higher matching precision. This experiment proves the effectiveness and the progressive nature of our descriptor method in natural scenes. In this experiment, the test dataset dimensionality is preprocessed, and 500 is set as the maximum quantity of extracted feature points of a single test image to guarantee the reliability of comparison between algorithms.

Meanwhile, this paper makes some evaluations on various descriptors on the HPatches dataset (HEstimation, matching precision, and matching recall) to better guarantee that our descriptor has superior performance in matching. The experiments in Tables 5 and 6 separately use HPatches Illum and HPatches View datasets. These two experiments set the same operation method: using the same threshold equal to 3, the nearest neighbor matching algorithm, and the RANSAC mismatching optimization algorithm [29]. As shown in Tables 5 and 6, in these two experiments, the descriptors with adaptability to changing environments put forward in this paper have superior performance in matching results in both HPatches Illum and View datasets. In the experiment in Table 5, the homography estimation is 0.883, the matching recall is 0.664, and the matching precision is 0.664. Besides, in the experiment in Table 6, the homography estimation in this work reaches 0.687, the matching recall is 0.494, and the matching precision is 0.625.

4.1.4. Matching Test Experiment. To make the local feature matching algorithm effectively deal with changing scenes, this paper makes use of the coordination of the AGNN+Sinkhorn matching algorithm and invariant feature descriptors. In this experiment, the combination of various local feature construction methods and various matching methods are used to test the matching outcome on the HPatches View dataset to prove the matching performance of our algorithm. The parameter selection in the local feature extraction phase is identical to the experiment shown in Tables 5 and 6, which sets 500 as the maximum quantity of local features collected for a single image. From Table 7, the various local feature construction algorithms using AGNN + Sinkhorn matching algorithm perform better than the conventional nearest neighbor matching approach in multiple indicators. The end-to-end algorithm integrating local feature formation and matching put forward in this paper has been dramatically increased in various evaluation indexes compared with the rest approaches. The homography estimation is raised by 0.097, with the matching precision is improved by 0.082, and the matching recall is rose by 0.038.

Meanwhile, this paper also gathers image pairs of the library, gymnasium, experimental building, and administrative building. These image pairs contain illumination, rotation, and viewpoint transformations. Before the test, the image pairs are preprocessed in dimension, and the pixel size is set to 1920×1200. The experiment aims to prove the effectiveness and superiority of the AGNN + Sinkhorn matching algorithm. Figure 9 indicates the comparison of visual matching results between the algorithm in this paper and some existing algorithms. In the experiment, 300 is set as the upper limit of feature point extraction of each algorithm on a single image. The first row is the original picture, the second row is the matching result of the ORB algorithm, the third row is the matching result of the SIFT algorithm, the fourth row is the matching result of the Superpoint algorithm, and the fifth row is for the matching results of the algorithm in this paper. In the last two columns of gymnasium and library image pairs with considerable viewpoints and illumination contrast, compared to other existing algorithms, the method put forward in this work can be used for local feature extraction and effective matching of image pairs in the presence of illumination, viewpoints, and abundant foliage occlusions and architectural interference. The color of the matching line in the figure indicates the matching confidence of the corresponding local features between the image pairs. The more the color is toward red, the higher the confidence of matching, and the more the color is toward blue, the lower the confidence of matching. To sum up, under the premise that the maximum extraction times of various local feature extraction algorithms are specified, the local features between image pairs can be more precisely and effectively matched under the algorithm proposed in this article.

Table 4: The performance of descriptor invariance in mutative environments.

	SIFT	SuperPoint	With invariance
With viewpoint, illumination, and rotation			
With viewpoint, illumination, and no rotation			

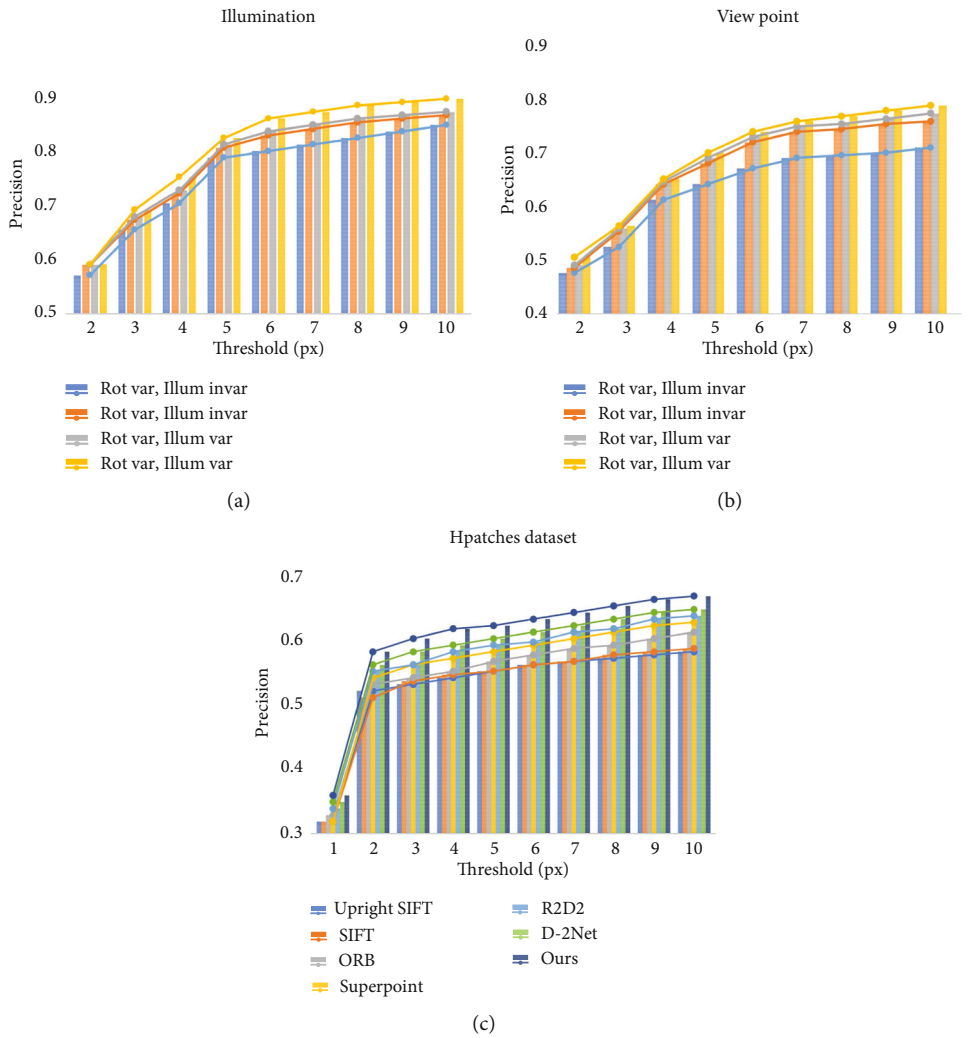


Figure 8: Descriptor experiment.

Table 5: The index comparison of HPatches Illum.

Dataset	Evaluation	SIFT	ORB	Superpoint	R2D2 [31]	D2-net [32]	Our method
HPatches Illum	Precision	0.553	0.564	0.629	0.656	0.651	0.664
	Recall	0.431	0.483	0.564	0.580	0.565	0.664
	HEstimation	0.897	0.884	0.876	0.816	0.818	0.883

Table 6: The index comparison of HPatches View.

Dataset	Evaluation	SIFT	ORB	Superpoint	R2D2 [31]	D2-net [32]	Our method
HPatches View	Precision	0.514	0.572	0.594	0.549	0.563	0.625
	Recall	0.349	0.421	0.445	0.371	0.381	0.494
	HEstimation	0.643	0.687	0.651	0.626	0.552	0.687

Table 7: The experimental contrast results.

Matching algorithm	Local feature	HEstimation	Precision	Recall
AGNN + Sinkhorn	SIFT	0.656	0.625	0.413
AGNN + Sinkhorn	ORB	0.692	0.677	0.579
Nearest neighbor	SIFT	0.643	0.514	0.349
Nearest neighbor	ORB	0.687	0.572	0.421
Nearest neighbor	Superpoint	0.651	0.594	0.445
Nearest neighbor	Des + NetVLAD	0.687	0.625	0.494
SuperGlue [19]		0.668	0.725	0.596
Our proposed method		0.784	0.807	0.634
Promotion		0.097	0.082	0.038

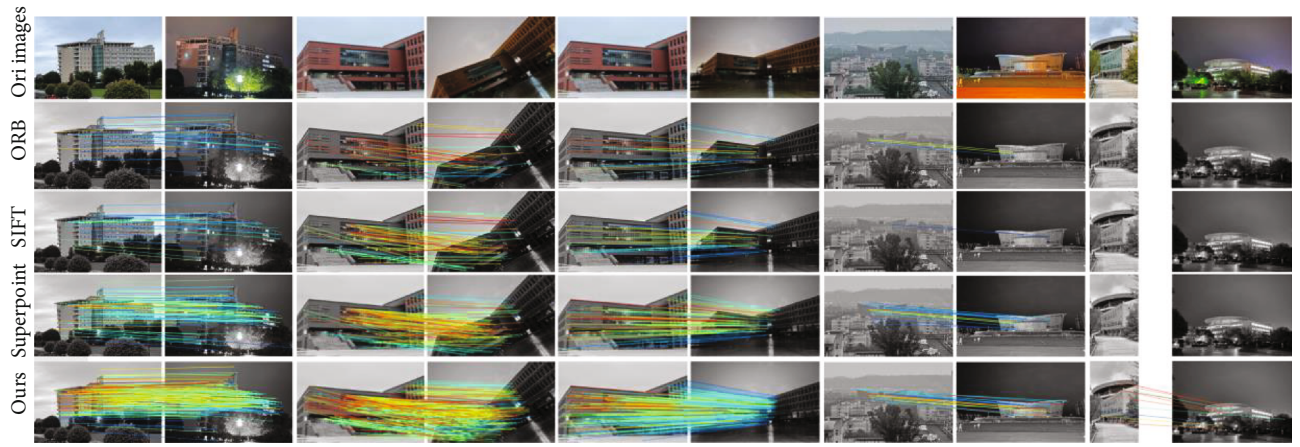


Figure 9: Matching self-collected changing pairs with ours and comparison algorithms.

5. Conclusions

This paper studies a local feature matching method in which the NetVLAD neural network module is implanted in the descriptor constitution phase, and the AGNN and Sinkhron matching algorithm is introduced in the matching stage, which effectively realizes the robustness of the algorithm against scene changes. Compared with typical algorithms, this article has achieved better behavior consequences in homography estimation, matching precision, and matching recall on the public dataset HPatches and self-collected image pairs with significant scene changes. Meanwhile, a single end-to-end network that can match local features more accurately and effectively is achieved. Owing to the strengthening of the invariance of descriptors, even though the matching problem between local features of natural scene images can be well done, the number and complexity of descriptors corresponding to feature points have also expanded compared with traditional algorithms resulting in calculation increase in cost. Finally, the research on the fusion of deep neural networks and local feature matching engineering has received increasing attention. The introduction of deep learning into machine vision-related issues such as visual odometry engineering has also received increasingly investment and research. However, the stability and effectiveness of the system after effective fusion are also nodes and difficulties to be considered, as well as issues that need to be studied in the follow-up.

Data Availability

No data were used to support this study.

Conflicts of Interest

The authors declared that there are no potential conflicts of interest with any financial organizations regarding the material reported in this article.

Acknowledgments

The research of the paper is supported by the Project of National Funding, PRC ([2019], No. 1276).

References

[1] M. Yan, H. Yuan, J. Xu, Y. Yu, and L. Jin, "Task allocation and route planning of multiple UAVs in a marine environment based on an improved particle swarm optimization algorithm," *EURASIP Journal on Advances in Signal Processing*, vol. 2021, no. 1, 2021.

[2] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[3] E. Rosten and T. Drummond, "Machine learning for high-speed corner detection," in *European conference on computer vision*, pp. 430–443, Berlin, Heidelberg, 2006.

[4] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: speeded up robust features," in *European conference on computer vision*, pp. 404–417, Berlin, Heidelberg, 2006.

[5] M. Calonder, V. Lepetit, C. Strecha, and P. Fua, "Brief: binary robust independent elementary features," in *European conference on computer vision*, pp. 778–792, Berlin, Heidelberg, 2010.

[6] M. Yan, S. Li, C. A. Chan, Y. Shen, and Y. Yu, "Mobility prediction using a weighted Markov model based on mobile user classification," *Sensors*, vol. 21, no. 5, p. 2021, 1740.

[7] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: an efficient alternative to SIFT or SURF," in *2011 International conference on computer vision*, pp. 2564–2571, Barcelona, Spain, 2011.

[8] M. Yan, X. Lou, and Y. Wang, "Channel noise optimization of polar codes decoding based on a convolutional neural network," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 1434347, 10 pages, 2021.

[9] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Deep image homography estimation," 2016, https://arxiv.org/abs/1606.03798.

[10] K. M. Yi, E. Trulls, V. Lepetit, and P. Fua, "Lift: learned invariant feature transform," in *European conference on computer vision*, pp. 467–483, Cham, Switzerland, 2016.

[11] Y. Verdie, K. M. Yi, P. Fua, and V. Lepetit, "TILDE: a Temporally Invariant Learned DEtector," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5279–5288, Boston, MA, USA, 2015.

[12] K. M. Yi, Y. Verdie, P. Fua, and V. Lepetit, "Learning to assign orientations to feature points," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 107–116, Las Vegas, Nevada, 2016.

[13] E. Simo-Serra, E. Trulls, L. Ferraz, I. Kokkinos, P. Fua, and F. Moreno-Noguer, "Discriminative learning of deep convolutional feature point descriptors," in *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 118–126, Santiago, Chile, 2015.

[14] D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperPoint: self-supervised interest point detection and description," in *2018 Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 224–236, Salt Lake City, UT, USA, 2018.

[15] D. DeTone, T. Malisiewicz, and A. Rabinovich, "Toward geometric deep slam," 2017, https://arxiv.org/abs/1707.07410.

[16] C. B. Choy, J. Y. Gwak, S. Savarese, and M. Chandraker, "Universal correspondence network," 2016, https://arxiv.org/abs/1606.03558.

[17] R. Pautrat, V. Larsson, M. R. Oswald, and M. Pollefeys, "Online invariance selection for local feature descriptors," in *European Conference on Computer Vision*, pp. 707–724, Cham, Switzerland, 2020.

[18] R. Arandjelović, P. Gronat, A. Torii, T. Pajdla, and J. Sivic, "NetVLAD: CNN architecture for weakly supervised place recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 6, pp. 1437–1451, 2018.

[19] P. -E. Sarlin, D. DeTone, T. Malisiewicz, and A. Rabinovich, "SuperGlue: learning feature matching with graph neural networks," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4937–4946, Seattle, WA, USA, 2020.

[20] J. Zhou, G. Cui, S. Hu et al., "Graph neural networks: a review of methods and applications," *AI Open*, vol. 1, pp. 57–81, 2020.

[21] A. Vaswani, N. Shazeer, N. Parmar et al., "Attention is all you need," in *Proceedings of the 31st International Conference on Neural Information Processing Systems (NIPS'17). Curran Associates Inc.*, pp. 6000–6010, Red Hook, NY, USA, 2017.

[22] M. Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," *Advances in Neural Information Processing Systems*, vol. 26, pp. 2292–2300, 2013.

[23] S. Yang, JingWang, S. Arif, M. Jia, and S. Zhong, "SAL-net: self-supervised attribute learning for object recognition and segmentation," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 2891303, 13 pages, 2021.

[24] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, https://arxiv.org/abs/1409.1556.

[25] C. Jin, T. Wang, X. Li et al., "A transformer generative adversarial network for multi-track music generation," *CAAI Transactions on Intelligence Technology*, vol. 2021, pp. 1–12, 2021.

[26] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[27] T. Y. Lin, M. Maire, S. Belongie et al., "Microsoft coco: common objects in context," in *European conference on computer vision*, pp. 740–755, Cham, Switzerland, 2014.

[28] L. Murmann, M. Gharbi, M. Aittala, and F. Durand, "A dataset of multi-illumination images in the wild," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4080–4089, Seoul, Republic of Korea, 2019.

[29] K. G. Derpanis, "Overview of the RANSAC algorithm," *Image Rochester NY*, vol. 4, no. 1, pp. 2-3, 2010.

[30] V. Balntas, K. Lenc, A. Vedaldi, T. Tuytelaars, J. Matas, and K. Mikolajczyk, "[Formula: see text]-Patches: a benchmark and evaluation of handcrafted and learned local descriptors," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 11, pp. 2825–2841, 2020.

[31] R. Jerome, W. Philippe, D. S. Cesar, and H. Martin, *R2D2: Repeatable and Reliable Detector and Descriptor*, NeurIPS, 2019.

[32] M. Dusmanu, I. Rocco, T. Pajdla et al., "D2-Net: a trainable CNN for joint description and detection of local features," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8084–8093, Long Beach, CA, USA, 2019.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 8121509, 14 pages
https://doi.org/10.1155/2021/8121509

Research Article

A Collaborative Cache Strategy in Satellite-Ground Integrated Network Based on Multiaccess Edge Computing

Li Yang[1], **Xiangguang Kong**[2,3], **Yaowen Qi**[1] and **Chengsheng Pan**[4]

[1]*School of Automation, Nanjing University of Science and Technology, Nanjing 210094, China*
[2]*Key Laboratory of Communication Network and Information Processing, Dalian University, Dalian 116622, China*
[3]*College of Information Engineering, Dalian University, Dalian 116000, China*
[4]*Nanjing University of Information Science and Technology, Nanjing 210044, China*

Correspondence should be addressed to Xiangguang Kong; kxglight@foxmail.com

Received 4 October 2021; Revised 4 November 2021; Accepted 9 November 2021; Published 3 December 2021

Academic Editor: Ming Yan

Copyright © 2021 Li 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.

Multiaccess edge computing (MEC) provides users with a network environment and computing storage capacity at the edge of the network, ensuring a deterministic service with low delivery delay. This paper introduces a new satellite-ground integrated collaborative caching network architecture based on MEC and studies the caching strategy. On the ground side, the edge nodes (ENs) are deployed to the user side to form a hierarchical collaborative cache mode centered on the base station. On the satellite side, we utilize intelligent satellite ENs to precache and multicast the highly popular contents, reducing the initial content delivery delay. Under the constraints of the user demand and storage capacity, we study the deployment and cache scheme of ENs and establish the delivery delay minimization problem. To solve the problem, we propose a content update decision parameter for content cache update and transform the problem into improving the hit rate of ENs. Simulation results show that the proposed MEC network architecture and content caching scheme can increase the caching system hit rate to 64% and reduce the average delay by 32.96% at most.

1. Introduction

The number of mobile devices connected to the global fifth-generation mobile communication system (5G) will soon exceed billions level [1], which is far faster than the growth rate of the 4G era [2]. A wider range of intelligent devices (such as Internet of Things devices, 4K/8K video devices, and augmented reality and virtual reality devices) will continue to access the Internet [3], which will cause a large amount of traffic on the network access side. At the same time, 5G network services will put forward deterministic requirements on time delay and reliability [4]. The existing network architecture and content delivery methods will have difficulty meeting the ubiquitous and efficient network interconnection.

Multiaccess edge computing (MEC) will be an important technology to meet the requirements of 5G deterministic network services [5, 6]. Based on MEC technology, existing studies have carried out a large number of studies on edge collaborative caching [7, 8], collaborative offloading [9, 10], energy optimization [11–14], and other aspects. MEC realizes short-distance and real-time processing of user business and achieves a faster business response, effectively reducing the time delay. With the empowerment of MEC technology, edge computing and caching nodes are deployed on the user side, which not only provides users with faster and high hit rate content service responses [15] but also relieves the pressure on the cloud server and reduces the traffic on the backbone network [16]. At the same time, using the computing and storage capabilities of satellites, the deployment of MEC on satellites is considered to provide low delay and more ubiquitous network application services [17, 18].

However, the computing and storage resources in the edge servers are limited [19]. To provide a better internet experience, considering the content request traffic and content type of the served region, analyzing user request data, and perceiving user request preference information [20], the edge node (EN) deployment scheme and the

collaboration scheme between them should be studied to reduce the average delivery delay and improve the request hit rate of users. For this reason, in [21], aiming to solve the problem of the collaborative cache mode of multicell scenarios on the base station side, the authors proposed a decentralized edge caching strategy via a multiagent framework with multiple actor networks and a single critic network. In [22], the authors proposed an edge network architecture based on software-defined network technology, which uses a neural network model to predict the content of user requests and precache user requests. An edge cooperative caching method based on delay and energy balance is proposed. However, the two above models will bring a large edge computing overhead to the edge server, and in the process of constantly updating model parameters, a large number of computing resources will be consumed.

The authors in [23] proposed a collaborative cache and resource allocation scheme among base station cache nodes and solved the collaborative cache problem with the goal of maximizing resource utilization. In [24], combining the cache of macrobase station and microbase station and considering the energy consumption, the minimization delay is associated with the energy consumption. Different user association strategies are proposed to minimize the delay while reducing the energy consumption. In [25, 26], the authors studied the collaborative caching scheme between users' terminals, but the collaborative sharing of storage and computing resources between terminals involves many beneficial distribution relations. The authors in [27] studied the collaborative caching scheme under the scenario of the Internet of Vehicles based on MEC technology and utilized the storage and computing resources of the Internet of Vehicles to enhance the edge caching and computing power. The authors in [28] studied cooperative cache contents in vehicles, base stations, and edge servers. In order to maximize the efficiency of content caching, reinforcement learning algorithms are used to optimize caching schemes on different devices to improve the cache hit ratio. However, in practical application scenarios, it is challenging to solve actual problems such as link disconnection and task switching caused by the high-speed movement of vehicles at all times. In [29], the authors proposed a double-layer service cache system of cloud-edge collaboration, designed a cache replacement algorithm according to the content popularity, and realized the efficient execution of tasks through the cooperation between edge nodes. However, the definition of prevalence is not sufficiently considered.

With the continuous evolution of 5G and the development of 5G-advanced and 6G network research, it has become the focus of research to use satellite and high-altitude communication platform capabilities to realize the ubiquitous global connectivity of the sky, ground, and sea [30, 31]. Therefore, to further reduce the end-to-end delay of users' content delivery [32, 33], it is necessary to fully explore the communication, storage, and computing capabilities of satellites and make use of satellite characteristics to meet more application scenarios of ground networks [34, 35]. In content storage and distribution, the satellite characteristics of wide-area connections will give more possibilities for network content delivery, which is expected to improve the quality of 5G-enhanced mobile broadband services [36]. For this reason, on the basis of studying the coordinated content distribution scheme of the ground dimension, satellite communication is used to expand the network spatial dimension.

Therefore, deploying MEC on satellite can provide lower delay and more ubiquitous network services [17, 37]. In the research of satellite-ground integrated cooperative networks based on MEC, the caching and computing nodes were deployed on the Iridium constellation in [38]. According to the characteristics of four intersatellite links for each satellite, a cooperative satellite-ground integrated network (SGIN) architecture with dual computing and caching capabilities on the satellite is proposed. In [39], the authors proposed a satellite-ground collaborative computing offloading model for edge computing at the user end, satellite end, and remote end and utilized dynamic network virtualization technology to integrate computing resources under the coverage of low-earth orbit satellites. However, this collaborative network model is difficult to apply to efficient edge caching applications. To solve the large delay caused by the long-distance transmission between the satellite and the ground, in [40], the authors added a fog computing edge node (EN) layer between the satellite and the ground base station to address different user services with different delay requirements. However, the deployment strategy of ENs below the base station is not considered, which cannot meet the higher delay requirements of 5G services. The authors in [41] proposed a time-domain separation satellite multicast and unicast cooperative transmission scheme. Multicast can satisfy the request of popular contents in most areas, while unicast can satisfy the request of unpopular contents in local areas. However, the storage resources of satellites are limited, and the caching of unpopular contents will cause a waste of resources and affect the efficient transmission of highly popular contents.

In the above studies on the ground collaborative caching scheme, the problem is mainly focused on the ENs on the base station side or below the base station. Above the base station, it directly communicates with the source server, without considering that there will still be a long communication distance between the base station and the source server, and there is still room for optimization of the delay. Besides, the author mainly focused on the formulation of cooperative caching strategy for local nodes and did not give sufficient consideration to the multilevel cooperative mode of caching nodes. In the study of satellite-ground cooperative cache, the main research is the collaboration between satellites, and the satellite-ground cooperative cache strategy between the satellite cache nodes and the ground cache nodes is not fully considered. Although these studies endowed satellite nodes with computing and storage capabilities, they failed to develop other possible potentials of satellite ENs, which restricted application scenarios of the satellite.

Therefore, this paper takes the base station as the center on the ground side, makes full use of the advantages of satellite communication in the space dimension, and proposes

an edge node cache architecture of the satellite-ground integrated collaborative network. The main contributions of this paper are summarized as follows:

(1) We first propose a novel satellite-ground integrated collaborative cache network architecture with a network hierarchical cache mode centered on the base station and a large-scale precache mode of the cache node cluster through satellites

(2) We then propose a collaborative cache and content delivery scheme and build a communication model and a computing model to minimize the average delivery delay of the system. We prove that the problem model is nondeterministic polynomial hard (NP-hard)

(3) We study the content update and replacement strategy of cache nodes to solve the intractable NP-hard problem. To maximize the hit rate of the cache node, we design a content update decision parameter for content cache update

(4) We study the content caching and content distribution strategy of the collaborative network architecture, and we design a multilevel node content cocaching algorithm MNC^{3}A based on the content update decision parameter to hierarchically and coordinately cache network content and minimize the delivery delay. The time computing complexity of the algorithm is finally proven to be $O(n)$

The paper is organized as follows. Section 1 discusses the related works on the integrated network architecture and caching strategy. Section 2 presents the SGIN architecture and MEC server deployment scheme. In Section 3, we present the system model including communication mode and problem formulation. Section 4 introduces the algorithm to solve the caching problem. In Section 5, we compare the performance of our algorithms in different scenarios. Section 6 concludes the paper.

2. SGIN Architecture Empowered by MEC

2.1. Satellite Network Communication Mode. With wide coverage and large connection characteristics, satellite networks can ignore terrain and regional interference and improve the user experience quality under specific application scenarios. This paper explores and utilizes the characteristics of satellite network communication from two aspects. First, the wide coverage of satellites is used to expand ground communication. User terminals can access the Internet through ground stations and satellite relays, as shown in Figure 1(a). When the ground network is unable to provide users with services or the service experience provided by the ground network is poor due to factors such as geographical location, communication distance, and natural disasters, the users will access the remote source server through satellite relay to obtain contents. Second, satellite communication assists the ground network by increasing the diversity of content caches, as shown in Figure 1(b). Due to the advantage of the satellite's large connection, the gateway communicates directly with the satellite, precaches regional hot contents through the satellite, and regularly broadcasts these contents to all gateways in the region covered by the satellite, realizing that hot contents in region A are cached in region B in advance. This saves the time that users in region B have to go to the source server to request contents when there is no precache, thus reducing the average delivery delay of the users.

2.2. Edge Node Deployment of SGIN. In this section, we introduce the hierarchical EN deployment design of SGIN based on MEC technology. This design makes full use of the advantages of computing resources, storage resources, wide coverage, and large connection characteristics of the satellite and solves the problems of insufficient coverage of ground resources, alleviating the load of ground links and reducing the delay of the first request and response of contents. As shown in Figure 2, under the catalysis of MEC technology, the traditional network cache nodes are sunk and split to form a hierarchical collaborative architecture including the user side EN (UEN) and base side EN (BEN), center EN (CEN) and internal EN (IEN) in the ground network, and satellite EN (SEN) in the satellite network. In the ground network, the microbase station on the user side is equipped with the computing storage edge server UEN, and nearby UENs can communicate with each other to exchange resources. All UEN contents are coordinated and managed by BEN. The IEN cluster is deployed on the metropolitan area network (MAN) side and is responsible for coordinating the caching of hot contents within certain areas. The CEN also takes the role of the autonomous gateway, which is responsible for the connectivity with the BEN and SEN and manages and dispatches the contents cached by the IEN.

Define the EN set $M = \{1, 2, 3, \cdots, m, \cdots, M\}$ and IEN set $\tilde{M} = \{1, 2, 3, \cdots, \tilde{m}, \cdots, \tilde{M}\}$. Network contents are cached, and tasks are processed through collaboration between multilevel MEC servers to improve user experience, ensure the requirements of deterministic timeliness, and increase the flexibility and stability of the network.

3. System Model

3.1. Related Variables. Define the content set $N = \{1, 2, 3, \cdots, n, \cdots, N\}$ and the content storage size set $S = \{s_1, s_2, \cdots, s_n, \cdots, s_N\}$. Define the ground cache edge node set $M = \{1, 2, 3, \cdots, m, \cdots, M\}$ and the UEN and IEN content cache 0-1 matrix $T^h = [T_1^h, T_2^h, \cdots, T_m^h, \cdots]$, where $T_m^h = [t_{1m}^h, t_{2m}^h, \cdots, t_{nm}^h, \cdots]$ and $h \in \{u, i\}$ represent the UEN and IEN, respectively. T^u is managed by BEN and T^i by CEN. Define SvEN, $v \in \{U, B, C, I, S\}$ represents the cache node storage capacity. That is, SvEN can be equal to SUEN, SBEN, SCEN, SIEN, and SSEN, which represents the storage capacity of UEN, BEN, CEN, IEN, and SEN, respectively. SNUM_m represents the cached content number of cache node m.

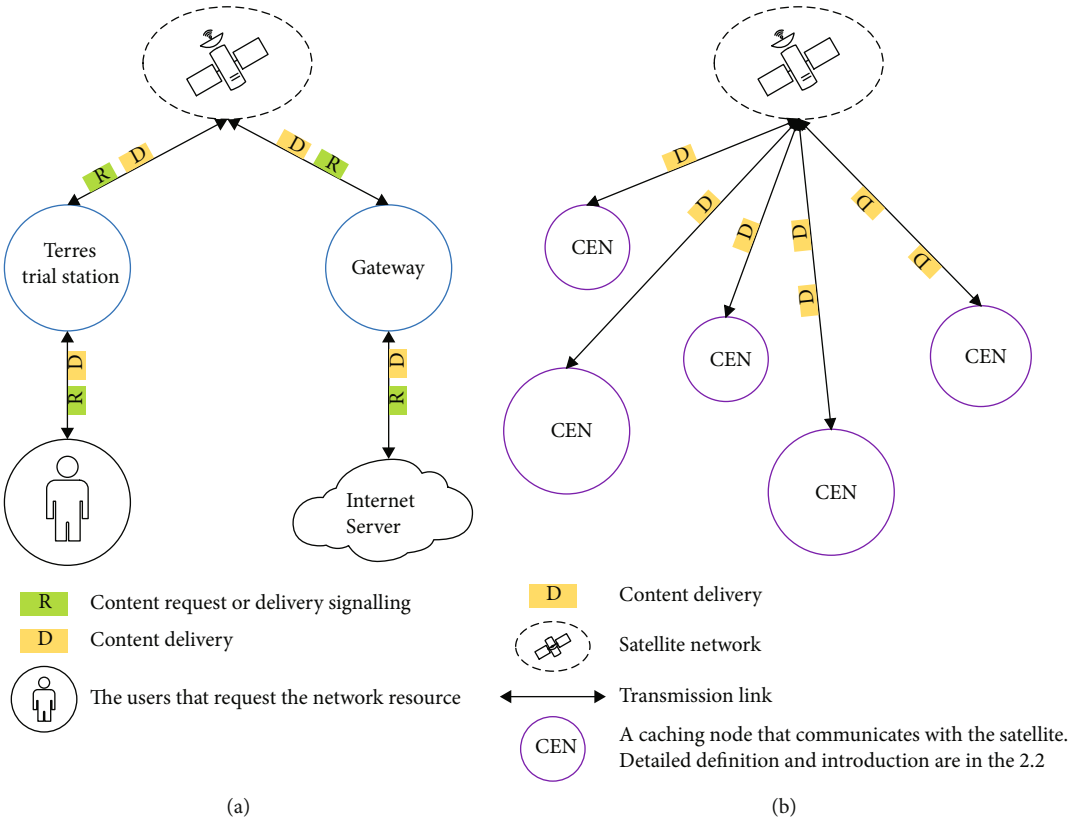


Figure 1: Two communication modes of a satellite network.

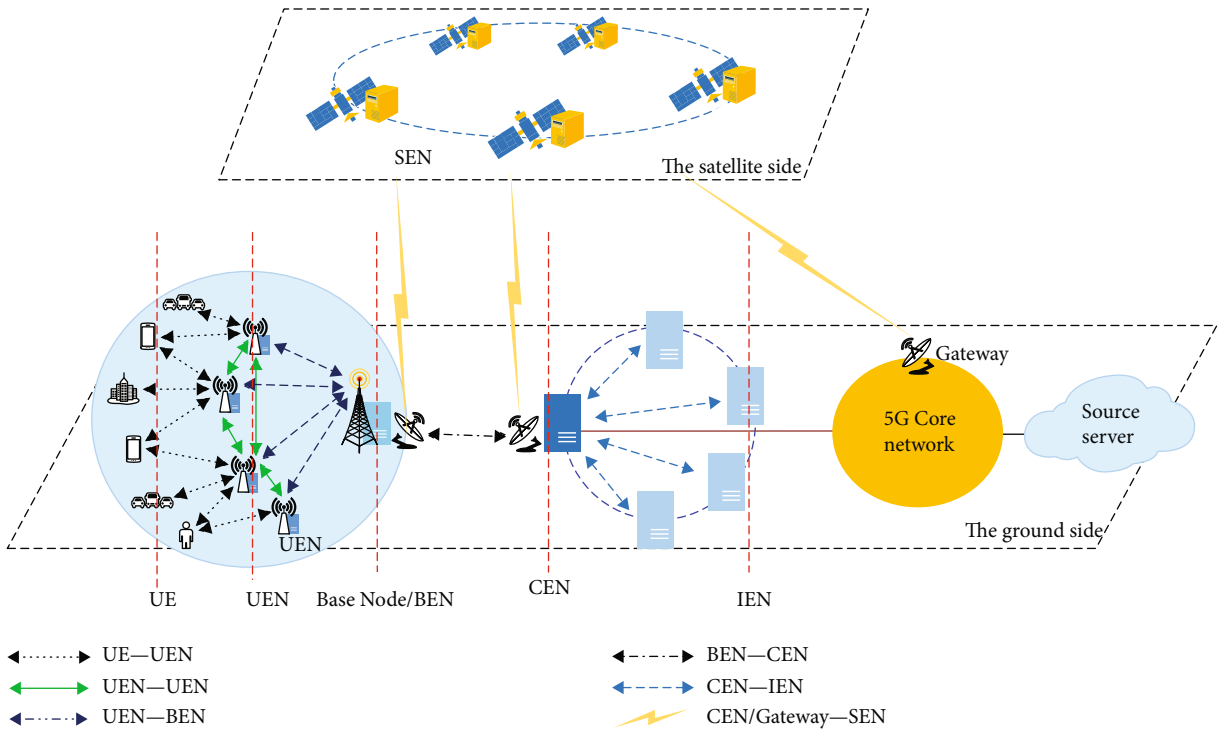


Figure 2: Architecture and edge node deployment of SGIN.

t_{nm}^{h} is defined by the following equation:

$$t_{nm}^{h} = \begin{cases} 1, & \text{if source } n \text{ in EN } m, \\ 0, & \text{others}, \end{cases} \tag{1}$$

when $t_{nm}^{u} = 1$, $t_{nm}^{i} = 0$.

Define the vector PM to record the cache priority of the cache node:

$$\mathrm{PM} = (\mathrm{PM}_1, \mathrm{PM}_2, \cdots, \mathrm{PM}_m, \cdots, \mathrm{PM}_M), \tag{2}$$

where PM_m represents the cache priority of node m.

The delivery delay should include transmission delay, propagation delay, processing delay, and queuing delay. The paper is aimed at verifying the content delivery delay minimization and edge node hit ratio maximization in multilayer converged network architecture. Processing delay and queuing delay depend on the node's computing and storage capacity and the current network load and network scheduling algorithm, which is beyond the scope of our research. We assume that the network status is good and the node's computing and storage capacity is enough. These two terms are set as constants that do not affect the final delivery delay, so they are not included in the final delay calculation model. Therefore, only transmission delay and propagation delay are considered [10]. For the calculation of propagation delay, the wireless transmission bandwidth between node m and user u is defined as d_m^u. The bandwidth between node $m1$ and node $m2$ is $l_{m1}^{m2}, \forall m \in M, \forall m1 \in M, \forall m2 \in M, m1 \neq m2$. Define B as the transmission bandwidth of the wired link and define D as the bandwidth of the satellite ground communication link.

3.2. Computation Model

3.2.1. Requested Content in UEN. The delivery delay in this case is

$$l_{(1)} = t_n^u \Delta_1, \tag{3}$$

where $\Delta_1 = s_n / d_m^u$, $t_n^u \epsilon \{0, 1\}$ indicates whether content n is in the UEN nearest to the user and can be delivered directly; if yes, $t_n^u = 1$.

3.2.2. Requested Content in BEN. In this case, BEN delivers the content to the user directly, and the delivery delay in this case is

$$l_{(2)} = (1 - t_n^u)(1 - t_{nm}^u)\Delta_1. \tag{4}$$

3.2.3. Requested Content in Another UEN Connected to the Current UEN Directly. One or more of the UEN directly connected to the current UEN may cache the content requested by the user. In this case, BEN notifies all another UEN (namely, UEN-m) to send the content to the current UEN. The user selects the content reached first and discards the content reached later. Compared with $l_{(1)}$, the content delivery delay in this case increases the propagation delay from BEN to UEN-m and from UEN-m to UEN. The delivery delay in this case is

$$l_{(3)} = (1 - t_n^u) t_{nm}^u (2\Delta_2 + \Delta_1), \tag{5}$$

where $\Delta_2 = s_n / l_{m1}^{m2}$.

3.2.4. Requested Content in IEN. Compared with $l_{(2)}$, the content delivery delay in this case increases the propagation delay from the CEN to the BEN. The delivery delay in this case is

$$l_{(4)} = (1 - t_n^u)(1 - t_{nm}^u) t_{nm}^i (\Delta_1 + \Delta_3), \tag{6}$$

where $\Delta_3 = s_n / B$.

3.2.5. Requested Content in SEN or Source Server. There are two cases when a user request passes through the satellite: if the content is cached in SEN, it is delivered directly; otherwise, it needs to be relayed by the satellite to the source server for the request. The delay in this case can be expressed as

$$\Delta = 2^c \Delta_4, \tag{7}$$

where $\Delta_4 = s_n / D, c \in \{0, 1\}$ represents the two cases of cached content and no cached content in SEN.

Users will select the response that arrives first from the two content acquisition methods of the ground backhaul link and satellite backhaul link and discard the response that arrives later, that is,

$$\Delta_5 = (\min(\Delta, (\Delta_1 + 2\Delta_3))). \tag{8}$$

The average delay in this case is

$$l_{(5)} = (1 - t_n^u)(1 - t_{nm}^u)\left(1 - t_{nm}^i\right)\Delta_5. \tag{9}$$

3.3. Problem Formulation. Based on the above analysis, we propose the problem model of content delivery delay minimization in the hierarchical cache mode of SGIN. In Sections 3.2.1–3.2.5, content delivery delay in each case is listed, respectively, but the delay in the process of a content request can only be one of the above cases. Therefore, the time delays from 3.1 to 3.5 are added to make only one case meet the requirements through parameter control [42]. Therefore, the objective function of the delay calculation is established as

$$\begin{aligned} L(x) &= \sum_{n=1}^{N}\sum_{m=1}^{M} l_{(1)} + l_{(2)} + l_{(3)} + l_{(4)} + l_{(5)} = \sum_{n=1}^{N}\sum_{M=1}^{M}(1 - t_n^u) \\ &\quad \cdot \left(t_{nm}^u \frac{2s_n}{l_{m1}^{m2}} + (1 - t_{nm}^u)(\Delta_5 + t_{nm}^i \left(\frac{s_n}{d_m^u} + \frac{s_n}{B} - \Delta_5 \right) \right) + \frac{s_n}{d_m^u} \\ &= \sum_{n=1}^{N}\sum_{m=1}^{M}(1 - t_n^u)\left(2t_{nm}^u \Delta_2 + (1 - t_{nm}^u)(\Delta_5 + t_{nm}^i(\Delta_1 + \Delta_3 - \Delta_5)\right) + \Delta_1 \\ &= \min\left\{\omega \cdot T_{n,m} \mid n \in N, m \in M\right\}, \end{aligned} \tag{10}$$

Input: content requested by the user. L, F.
Output: FoHR_n, FoRR_n, ToL_n, P_n, PB.
1: Initialize: $\text{FoHR}_n = 0$, $\text{FoRR}_n = 0$, $\text{ToL}_n = 0$.
2: for each requested $n \in N$
3: $\text{FoRR}_n + = 1$;
4: $\text{ToL}_n = L$;
5: for each $\bar{n} \in N \&\& \bar{n}! = n$
6: $\text{ToL}_{\bar{n}} - = 1$;
7: if ($\text{ToL}_{\bar{n}} == 0$)
8: $P_n = \tau \text{FoHR}_n + (1 - \tau)\text{FoRR}_n$;
9: if ($P_n < F$)
10: $\text{PB} = P_n\%100/100$;
11: Que.push($\bar{n}$) by $(1 - PB) \| \text{Que.delete}(\bar{n})$ by PB;
12: else
13: $\text{ToL}_{\bar{n}} = L$;
14: $\text{FoHR}_{\bar{n}} = \text{FoHR}_{\bar{n}} + \text{FoRR}_{\bar{n}}$;
15: $\text{FoRR}_{\bar{n}} = 0$;
16: end
17: end
20: end for
21: end for

Algorithm 1: Content update algorithm LF.

Input: content request tasks that obey Zipf distribution.
Output: content response and caching strategy.
1: for each requested $n \in N$
2:/∗requesting n ∗/
3: if ($t_n^u == 1$)
4: executing Algorithm 1 and response to user directly;
5: else if ($t_n^u == 0 \&\& t_{nm}^u == 0$)
6: executing Algorithm 1 and response to user via BEN;
7: else if ($t_n^u == 0 \&\& t_{nm}^u == 1$)
8: executing Algorithm 1 and response to user via UEN-m;
9: else if ($t_n^u == 0 \&\& t_{nm}^u == 0 \&\& t_{nm}^u == 1$)
10: executing Algorithm 1 and response to user via IEN;
11: Else
12: Requesting from source server;
13:/∗caching n ∗/
14: while ($\sum_{i=1}^{\text{SNUM}} s_i + s_n > \text{SvEN}$) $v \in \{U, B, C, I, S\}$
15: Que.delete (Minimum_Memory_Size (i));
16: end while
17: caching n in IEN with mini (PM_m);
18: executing Algorithm 1;
19: caching n in vEN $v \in \{U, B, C, S\}$;
20: executing Algorithm 1;
21: end for
22: for each $n \in N$ in IEN
23: find several most popular content and transmit to BEN and SEN;
24: end for
25: broadcast content to CEN via satellite and caching in IEN;
26: executing Algorithm 1;

Algorithm 2: MNC^{3}A.

where $T_{n,m}$ is the delivery delay of content n cached in node m, ω is the indicator parameter, and $\omega \in \{0, 1\}$. If $\omega = 1$, the content is already cached in the node; otherwise, it is not cached. $\omega \bullet T_{m,n}$ means that the content requested by the user can only be delivered by the only edge node with the smallest delay. The average delay optimization problem

of the system is formulated as follows:

$$P1 : \min_{x} L(x), \tag{11}$$

$$s.t. \sum_{n=1}^{N} t_{nm}^{v} s_n \leq \text{SvEN}, \forall n \in N, \forall m \in M, t_{nm}^{v} \in (0,1), \tag{12}$$

where (11) is the target problem to be optimized and (12) is the caching storage constraint and ENs caching indicator parameter.

By reducing problem $P1$ to the generalized assignment problem, since the generalized assignment problem is an NP-hard problem, it can be proven that problem $P1$ is an NP-hard problem [43]. $P1$ is a problem model based on delay minimization. From the perspective of cache nodes, $P1$ is the problem of content delivery delay minimization. The content delivery of the cache node depends on what content the cache node caches. Therefore, the node cache optimization problem is a subproblem of the content delivery optimization problem. Caching N contents on M nodes and minimizing the time delay under this caching scheme are a classic assignment problem, so optimization problem $P1$ is a NP-hard problem.

4. Algorithm Design

4.1. Problem Description. When the delay of system content delivery decreases, it means that the content requested by the users is gradually cached in the edge cache node close to the user; that is, the cache node close to the user has a higher hit rate. Therefore, solving problem $P1$ is simplified to solving the problem of maximizing the cache hit rate of the user's edge node.

Considering the contents that are accessed cyclically and alternately, for the content update strategy of the cache node, the historical request number of content n is defined as FoHR_n, the recent request number is FoRR_n, and the lifetime is ToL_n to lay down the content update decision, where the update rules of FoHR_n and FoRR_n are as follows:

$$\begin{aligned} \text{FoHR}_n &= \text{FoHR}_n + \text{FoRR}_n, \\ \text{FoRR}_n &= 0. \end{aligned} \tag{13}$$

The update rules of FoHR_n and the relationship among FoHR_n, FoRR_n, and ToL_n are described as follows.

For each cache or update, FoRR_n of the requested content n is incremented by 1, the lifetime ToL_n is set to $L(L \in R^+)$, and the lifetime of all other content is reduced by 1. When the lifetime of a certain content becomes zero, the content update decision variable P_n is considered:

$$P_n = \tau \text{FoHR}_n + (1-\tau)\text{FoRR}_n. \tag{14}$$

$\tau \in (0,0.5)$ is the weighted parameter used to control the influence of the historical and recent request numbers on the updating decision of contents. The value range of τ indicates that a single content update decision is greatly influenced by its recent request number. At the same time, FoHR_n and FoRR_n are update. If P_n is less than a threshold F, the content is retained with PB probability and deleted with 1 − PB probability. Otherwise, the content lifetime ToL is reset to L. The PB definition is as follows.

Table 1: The simulation parameters.

Parameter	Value
Content number, N	30000
Propagation rate, c (m/s)	3×10^8
LEO height (km)	700
β	0.2
γ	0.5
μ	0.3

$$\text{PB} = \frac{P_n \%100}{100}. \tag{15}$$

When content n is retained with a probability of 1 − PB, the content is placed in the waiting queue, Que (in this paper, the waiting queue is defined as Que for holding inactive content). If the content is requested again, the values of FoHR_n, FoRR_n, and ToL_n are reset. When the cache node is unable to continue caching contents due to insufficient memory, the inactive content is replaced from e, in the order of content storage size from small to large.

Through the above analysis, the NP-hard problem $P1$ that is difficult to solve is transformed into a problem $P2$ that maximizes the cache hit rate of the underlying cache node UEN with the content update decision determined by L and F.

$$P2 : \max_{L,F} f(L,F) = \frac{\sum_{i=1}^{U} \partial_i}{\text{UT}}, \tag{16}$$

$$s.t. \sum_{n=1}^{N} t_{nm}^{u} s_n \leq \text{SUEN}, \forall n \in N, \tag{17}$$

where $f(L,F)$ is the UEN hit rate expression under the content update decision affected by the content request number and content lifetime. UT is the total number of content requests, and $\partial_i \in \{0,1\}$ indicates whether the requested content is in the UEN. If the request is hit, $\partial_i = 1$; otherwise, $\partial_i = 0$. (17) is the cache node storage resource constraint. The optimization problem $P2$ can be regarded as a binary optimization problem concerning variables L and F.

According to the above analysis, the content update algorithm, namely, LF, is described as Algorithm 1.

4.2. Multilevel Node Content Cocaching Algorithm. The establishment of a cache strategy should consider the joint constraints of the cache nodes' own storage capacity, busy degree, load balance, and content update decision to minimize the average delay of the system and maximize the hit rate. To ensure the load balance of the IEN server, considering the total number of existing contents of ENs in the

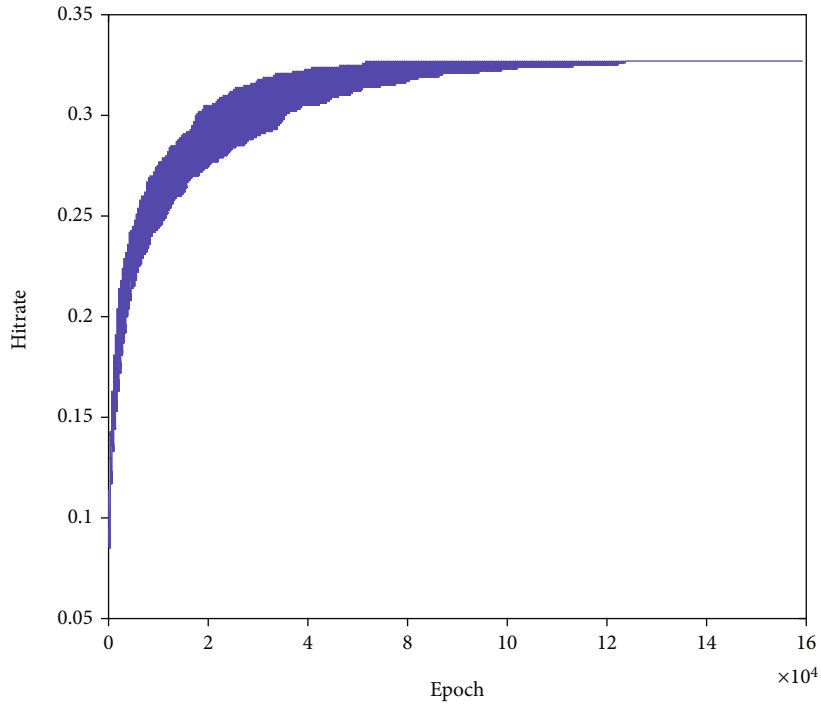


Figure 3: Process to solve L and F.

cluster, the storage resources occupied by these contents and the current busyness of the EN (the number of service requests), the storage priority PM_m of cache node M is defined as follows:

$$\mathrm{PM}_m = \beta \mathrm{SNUM}_m + \gamma \sum_{n=1}^{\mathrm{SNUM}} s_n + \mu \sum_{n=1}^{\mathrm{SNUM}} \mathrm{FoRR}_n, \quad (18)$$

where $\beta + \gamma + \mu = 1, \beta \in (0, 1), \gamma \in (0, 1)$, and $\mu \in (0, 1)$. In this paper, we stipulate that the lower the PM value is, the higher the priority of the cache node is selected for content caching.

When the content update occurs in UEN and IEN, the t_{nm}^h corresponding position should be set to 0 after the update. At the same time, in order to ensure that contents in BEN server are the most popular in the region, it is necessary to check contents cached by BEN regularly and replace the low popular contents in BEN with the high popular contents in IEN through CEN. For SEN, considering the hot contents from local areas to ubiquitous areas with radioactive transmission characteristics, highly popular contents in certain areas for a certain period of time may also be popular in other areas, so highly popular contents in IEN are chosen to be transmitted to SEN by CEN. SEN will broadcast to all CEN clusters in the covered area on a regular basis to realize advanced caching and active pushing of highly popular contents. It reduces the initial request delay of contents and then reduces the average delivery delay of the system as a whole.

Based on the above analysis, we propose the content caching and delivery algorithm MNC^{3}A of multilevel collaborative caching nodes. The implementation of the algorithm is described as Algorithm 2.

4.3. Time Complexity Analysis. The time complexity of Algorithm 2 is on the $O(n)$ level. The proof is as follows.

Since Algorithm 2 executes Algorithm 1 internally multiple times, we first analyze the time complexity of Algorithm 1. In a single time slice of content caching and update, FORR, FOHR, ToL, and other parameters of all other n contents need to be updated. Therefore, the time complexity of Algorithm 1 is $O(n)$ in each content request process. Meanwhile, in each content delivery and caching process, Algorithm 2 needs to execute Algorithm 1 multiple times, so the time complexity of Algorithm 2 is $O(kn), k \in R^+$. In conclusion, the time complexity of Algorithm 2 is approximately on the $O(n)$ level.

5. Simulation Analysis

In this paper, we use MATLAB tools for simulation analysis. Based on the proposed edge network architecture and collaborative caching scheme of SGIN, first, we solve the value of L and F. Second, the benefits of the content update algorithm LF are verified by comparing with the traditional algorithms least recently used (LRU) [44] and least frequently used (LFU) [45]. Second, we determine the best way to deploy edge cache nodes under different traffic conditions in different regions and verify the benefits of deploying hierarchical cache nodes. Then, the MNC^{3}A algorithm proposed in this paper is compared with other verified algorithms in the literature. Finally, the effect of reducing the average delay by satellite precaching is verified. The relevant parameters of the simulation are described in Table 1, and the network contents follow a Zipf distribution.

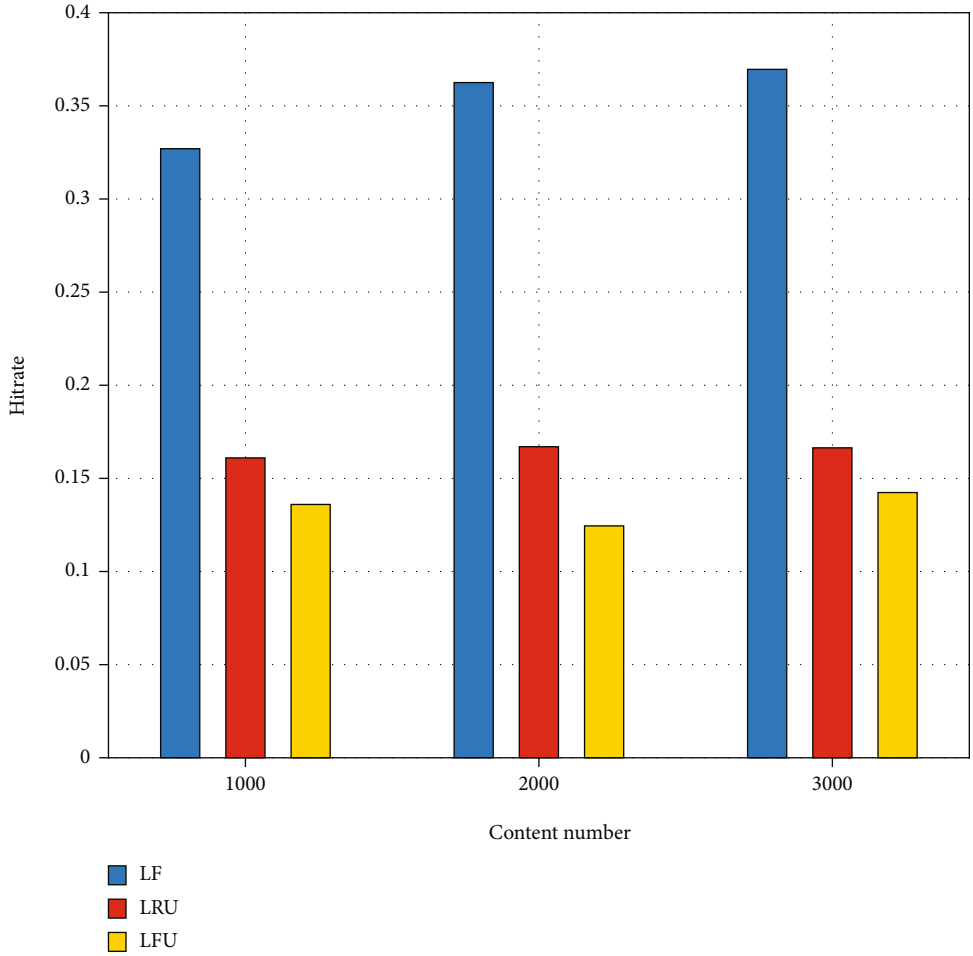


Figure 4: Hit rate of different algorithms for a single cache node.

5.1. Solving Parameters L and F. In this section, we will solve the key parameters L and F proposed in Section 4, which determine the content update strategy.

As shown in Figure 3, we find that the curve shows a gradual upward trend and then fluctuates around a maximum value. Finally, it converges to the maximum and the maximum cache hit rate of a single node are 0.327. Therefore, there are many pairs of optimal L and F values. We choose the point $(149,0.1,0.327)$, that is, $L = 149$ and $F = 0.1$. The values will be used in the following simulations.

5.2. Content Update Algorithm Hit Rate Comparison. This section compares the hit rate of the LF content update algorithm proposed in Section 4 with that of the LFU algorithm and LRU algorithm. We compare the hit rate of a single cache node with 1000, 2000, and 3000 contents that obey the Zipf distribution. Figure 4 shows that the proposed cache update algorithm has a higher hit rate performance than the traditional algorithm.

5.3. The Effect of the Number of IEN on the Delay. Regardless of the connections between users and UEN, BEN, between BEN and CEN, and between CEN and SEN, this section only discusses the scenarios in which CEN requests contents from the source server and the collaborative relationship between CEN and IEN. The goal is to find the most appropriate number of IENs for a particular scenario. Each set of data is the average of 20 simulations. As shown in Figure 5, with the increase in the number of IENs, the average delay curves of Figures 5(a)–5(d) all show a continuous decline at first and then fluctuate around the lowest point after falling to a certain extent. This means that when the number of IENs is too small, the system cannot cache too much content, so that it has to request contents from the source server frequently, which will consume considerable time in the continuous content update process. As the number of IENs continues to increase, the delay decreases and tends to a constant value. With different numbers of tasks, the fluctuation starts from different starting points. The more tasks there are, the longer the curve continues to decline. When the number of tasks is $N = 10000$, the delay starts to fluctuate when the number of IENs is 3, while when the number of tasks is $N = 30000$, the delay curve starts to fluctuate when the number of IENs is 9. It can be concluded that the number of IENs should be reasonably planned according to the number of tasks in certain geographical regions to avoid problems such as insufficient storage resources and unconspicuous delay reduction caused by too little IEN and

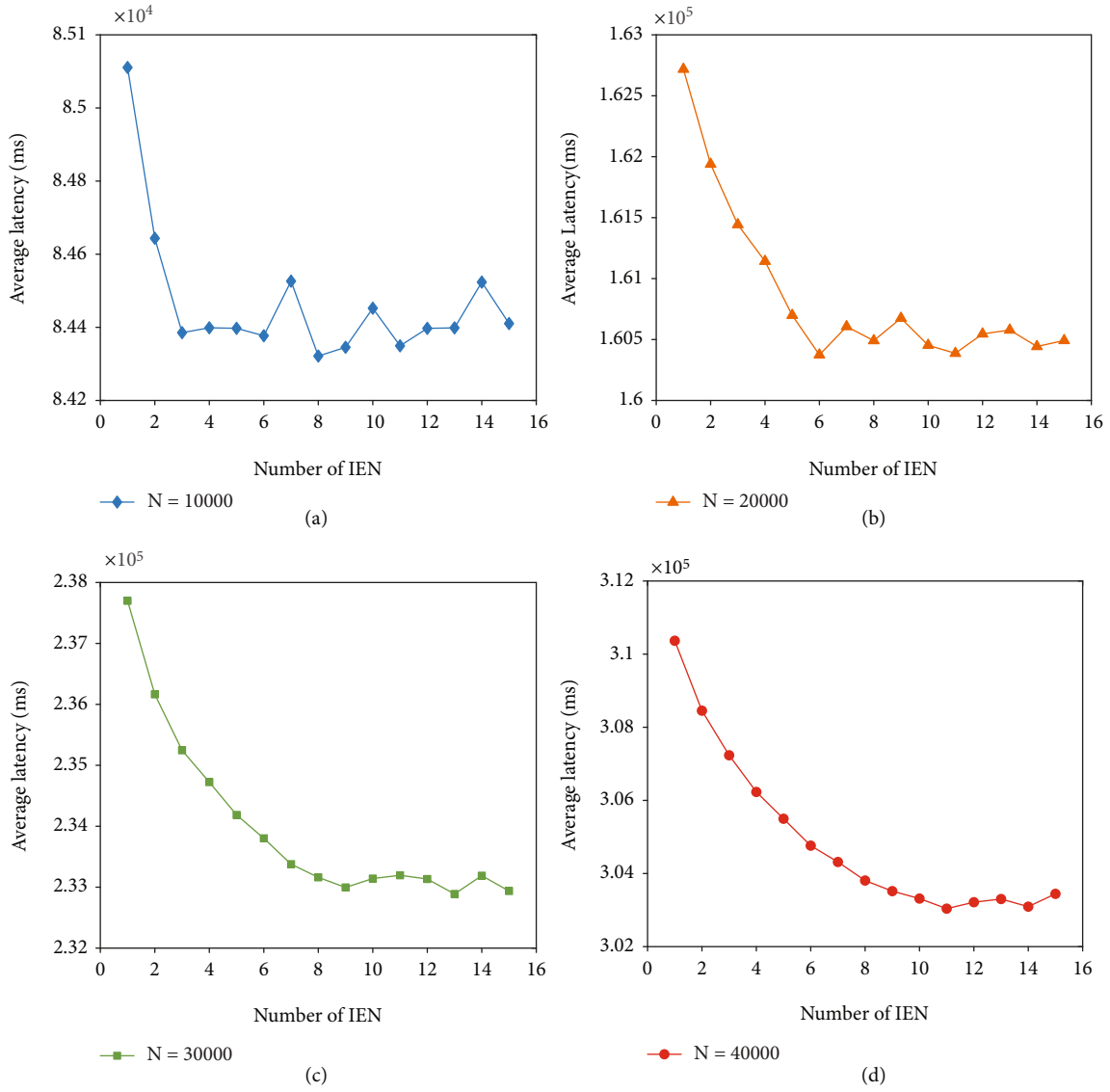


Figure 5: The effect of the number of IENs on the delay under different content numbers.

problems such as increased construction cost, management difficulties, and resource consumption caused by too much IEN.

5.4. The Effect of Cache Node on Delay on the MAN Side. In this simulation section, the benefits of deploying cache nodes on the MAN side are verified, as shown in Figure 6. Obviously, with the center of the base station and two layers cooperative cache mode, the delay performance is superior to the single base station cooperative caching mode. As the number of user content requests increases, the two modes of the delay gap gradually increase because in the process of users constantly requesting network content, IEN is rich in contents, and the user request preferences are gradually met in the area. The average delivery delay is further reduced.

5.5. Comparison of Hit Rates of Different Caching Algorithms. This section compares the cache hit rate between the cache strategy proposed in this article, the UCC cache scheme, and the cLRU(m) cache scheme proposed in [23], as shown in Figure 7.

Under the proposed collaborative caching strategy MN C^3A, when the base station directly obtains content from the source server, the cache hit rate is 45%, which is better than 35% for cRLU(m). However, in the deployment of the CEN+IEN collaborative architecture on the base station, the cache hit rate of MNC^3A finally reaches approximately 64%, which is better than those of 55% for UCC and 35% for cRLU. The average delivery time under a high hit rate is significantly reduced, and the user's QoS is improved, which proves the benefits of the hierarchical collaborative caching strategy proposed in this paper.

5.6. The Effect of Satellite Precache on Delay. The simulation in this section will study the impact of satellite precaching of hot content in a certain area on the average delivery delay in other areas. In the multiple IEN and CEN cluster areas

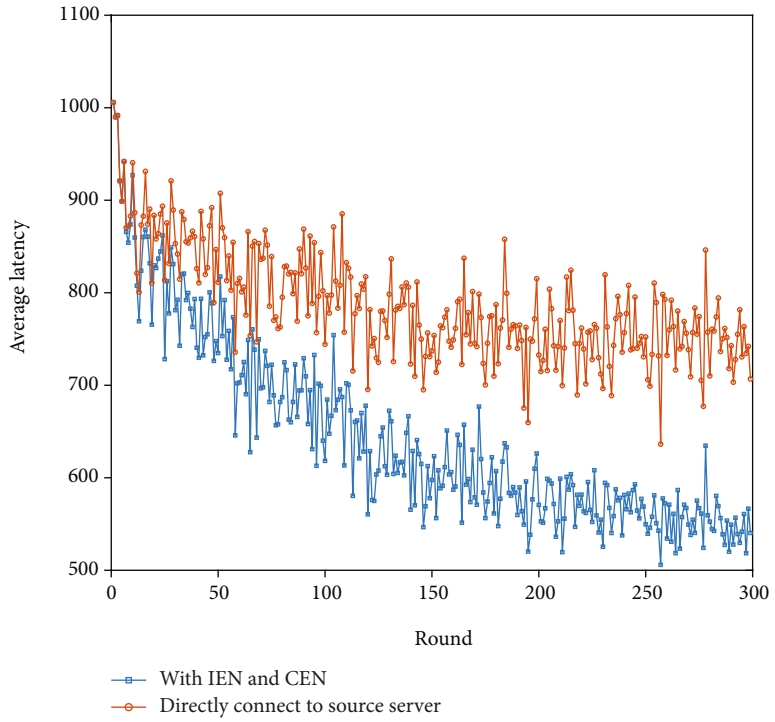


Figure 6: Impact of IEN and CEN multilevel collaboration on delay.

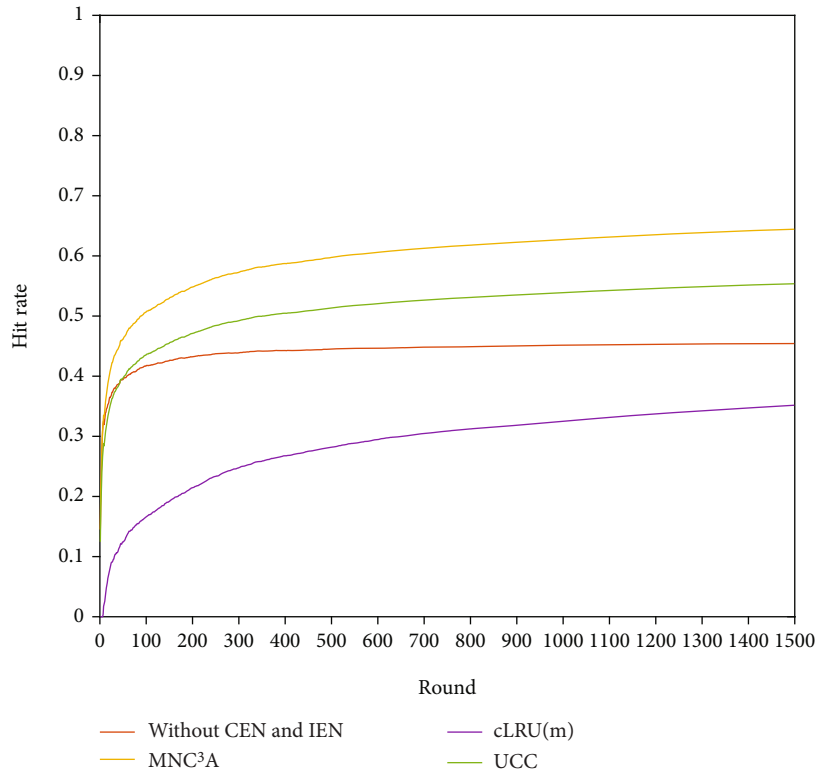


Figure 7: Hit rate comparison of different algorithms.

under satellite coverage, the most popular contents in area A are regularly cached by SEN and broadcast to other areas except A through satellites to achieve the precaching and active pushing of contents. By reducing the user's first request delay for a certain content, the average network delay can be further reduced. Figure 8 is a simulation

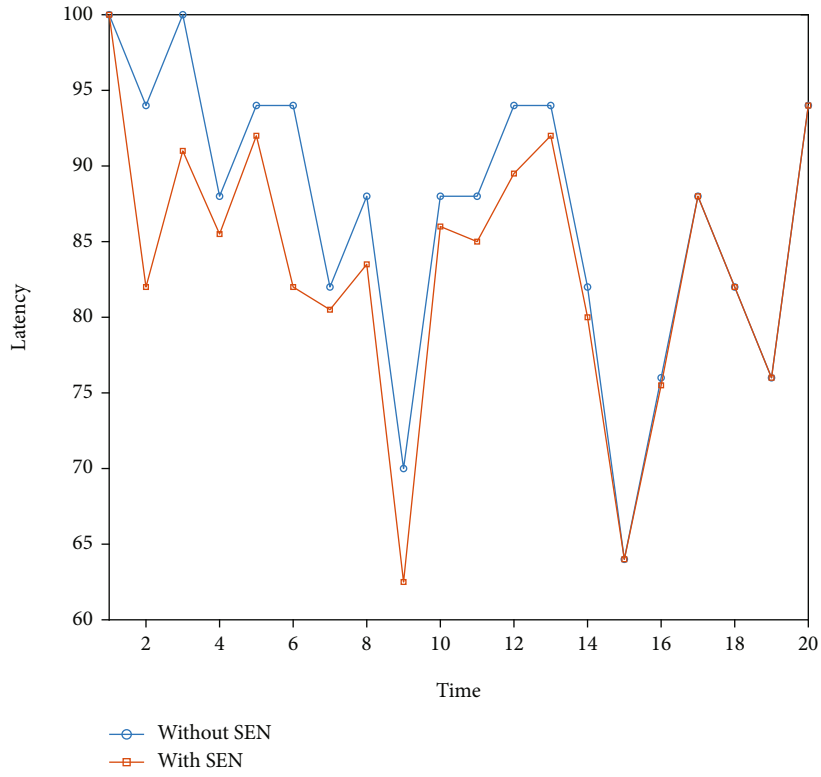


Figure 8: The effect of satellite on the delay.

comparison between satellite precaching and broadcast contents and only through terrestrial communication. As shown in Figure 8, in the first half of the time, the average delay of users in the satellite precaching mode is significantly lower than that in the case of nonsatellite precaching. This is because the satellite precaches the hot content when it is idle and broadcasts it to other areas under the satellite's coverage, saving users the time to go to the source server to request contents when there is no precaching. In the second half of Figure 8, when the content of satellite precaching is gradually cached in the IEN following the user's request, the average delay approaches gradually or even coincides.

6. Conclusion

Based on future network development trends, this article proposes a satellite-ground integrated network cache architecture based on multiaccess edge computing technology. On this basis, we establish a network content delivery delay minimization problem model and transform the problem into solving the problem of maximizing the hit rate of edge nodes. We study the collaborative cache strategy of multilevel cache nodes by finding the key decision parameters that determine the caching content update and replacement strategy. Simulation analysis shows that the proposed satellite-ground integrated edge network caching architecture and content caching strategy have a higher hit rate and significantly reduce the average delivery delay. In future work, the following aspect is still worth further study. The cache architecture studied in this paper is a static network architecture under a single time chip, but both 5G terminals and satellite nodes have strong mobility. It may be possible to consider the impact of nodes mobility on the entire cache system, and we can formulate the caching strategy under the dynamic network architecture.

Data Availability

The simulation data (media data subject to ZIPF distribution) used to support the findings of this study are included within the article. More details are available from the authors upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grants U21B2003 and 61722105.

References

[1] G. Forecast, *Cisco visual networking index: global mobile data traffic forecast update, 2017–2022. Update*, vol. 2017, p. 2022, 2019.

[2] M. Yan, C. A. Chan, W. Li et al., "Network energy consumption assessment of conventional mobile services and over-the-top instant messaging applications," *IEEE Journal on*

Selected Areas in Communications, vol. 34, no. 12, pp. 3168–3180, 2016.

[3] M. Yan, C. A. Chan, W. Li, L. Lei, A. F. Gygax, and C. L. I, "Assessing the energy consumption of proactive mobile edge caching in wireless networks," *IEEE Access*, vol. 7, pp. 104394–104404, 2019.

[4] T. Huang, S. Wang, Y. D. Huang, Y. Deng, J. Liu, and Y. J. Liu, "Survey of the deterministic network," *Journal on Communications*, vol. 40, pp. 160–176, 2019.

[5] J. Andrews, S. Buzzi, W. Choi et al., "What will 5G be?," *IEEE Journal on Selected Areas in Communications*, vol. 32, no. 6, pp. 1065–1082, 2014.

[6] Y. C. Hu, M. Patel, D. Sabella, N. Sprecher, and V. Young, "Mobile edge computing: a key technology to-wards 5g," *ETSI*, vol. 11, no. 11, pp. 1–16, 2015.

[7] R. Chen, H. Lu, and P. Ma, "User-centric cooperative MEC service offloading," in *2021 IEEE Wireless Communications and Networking Conference (WCNC)*, pp. 1–6, Nanjing, China, March 2021.

[8] T. M. Ayenew, D. Xenakis, N. Passas, and L. Merakos, "Cooperative content caching in MEC-enabled heterogeneous cellular networks," *IEEE Access*, vol. 9, pp. 98883–98903, 2021.

[9] R. Xie, Q. Tang, Q. Wang, X. Liu, F. R. Yu, and T. Huang, "Satellite-terrestrial integrated edge com-puting networks: architecture, challenges, and open issues," *IEEE Network*, vol. 34, no. 3, pp. 224–231, 2020.

[10] S. Zhong, S. Guo, H. Yu, and Q. Wang, "Cooperative service caching and computation offloading in multi-access edge computing," *Computer Networks*, vol. 189, p. 107916, 2021.

[11] C. Li, P. Liu, C. Zou, F. Sun, J. M. Cioffi, and L. Yang, "Spectral-efficient cellular communications with coexistent one- and two-hop transmissions," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 8, pp. 6765–6772, 2016.

[12] C. Li, Y. Li, K. Song, and L. Yang, "Energy efficient design for multiuser downlink energy and uplink information transfer in 5G," *Science China Information Sciences*, vol. 59, no. 2, pp. 1–8, 2016.

[13] C. Li, S. Zhang, P. Liu, F. Sun, J. M. Cioffi, and L. Yang, "Overhearing protocol design exploiting intercell interference in cooperative green networks," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 1, pp. 441–446, 2016.

[14] M. Yan, H. Yuan, Z. Li, Q. Lin, and J. Li, "Energy savings of wireless communication networks based on mobile user environmental prediction," *Journal of Environmental Protection and Ecology*, vol. 22, no. 1, pp. 206–217, 2021.

[15] X. Chen, L. He, S. Xu, S. Hu, Q. Li, and G. Liu, "Hit ratio driven mobile edge caching scheme for video on demand services," in *2019 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1702–1707, Shanghai, China, July 2019.

[16] J. Sung, M. Kim, K. Lim, and J. K. K. Rhee, "Efficient cache placement strategy in two-tier wireless content delivery network," *IEEE Transactions on Multimedia*, vol. 18, no. 6, pp. 1163–1174, 2016.

[17] J. Wei, J. Han, and S. Cao, "Satellite IoT edge intelligent computing: a research on architecture," *Electronics*, vol. 8, no. 11, p. 1247, 2019.

[18] Z. Zhang, W. Zhang, and F. H. Tseng, "Satellite mobile edge computing: Improving QoS of high-speed satellite-terrestrial networks using edge computing techniques," *IEEE Network*, vol. 33, no. 1, pp. 70–76, 2019.

[19] Y. Hao, M. Chen, L. Hu, M. S. Hossain, and A. Ghoneim, "Energy efficient task caching and offloading for mobile edge computing," *IEEE Access*, vol. 6, pp. 11365–11373, 2018.

[20] M. Yan, S. Li, C. A. Chan, Y. Shen, and Y. Yu, "Mobility prediction using a weighted Markov model based on mobile user classification," *Sensors*, vol. 21, no. 5, p. 1740, 2021.

[21] C. Zhong, M. C. Gursoy, and S. Velipasalar, "Deep reinforcement learning-based edge caching in wireless networks," *IEEE Transactions on Cognitive Communi-cations and Networking*, vol. 6, no. 1, pp. 48–61, 2020.

[22] C. Li, Y. Zhang, Q. Sun, and Y. Luo, "Collaborative caching strategy based on optimization of latency and energy consumption in MEC," *Knowledge-Based Systems*, vol. 233, p. 107523, 2021.

[23] A. Ndikumana, S. Ullah, T. LeAnh, N. H. Tran, and C. S. Hong, "Collaborative cache allocation and computation offloading in mobile edge computing," in *2017 19th Asia-Pacific Network Operations and Management Symposium (APNOMS)*, pp. 366–369, Seoul, Korea (South), September 2017.

[24] G. Vallero, M. Deruyck, M. Meo, and W. Joseph, "Base station switching and edge caching optimisation in high energy-efficiency wireless access network," *Computer Networks*, vol. 192, p. 108100, 2021.

[25] C. Ge and K. Xu, "A hierarchical cooperative caching strategy for mobile content delivery network," *Chinese Journal of Computers*, vol. 41, pp. 2769–2786, 2018.

[26] X. Huang, L. He, X. Chen, G. Liu, and F. Li, "A more refined mobile edge cache replacement scheme for adaptive video streaming with mutual cooperation in multi-MEC servers," in *2020 IEEE International Conference on Multimedia and Expo (ICME)*, pp. 1–6, London, UK, July 2020.

[27] W. Li, B. Zhang, and X. Wang, "V2X collaborative caching and resource allocation in MEC-based IoV," *Journal of Communications*, vol. 42, pp. 1–14, 2021.

[28] L. Zhao, H. Li, N. Lin, M. Lin, C. Fan, and J. Shi, "Intelligent content caching strategy in autonomous driving toward 6G," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–11, 2021.

[29] X. Wu, Y. Wang, S. Yuan, and P. Xie, "A service caching method for mobile edge computing," *Journal of Physics: Conference Series*, vol. 2005, no. 1, p. 012060, 2021.

[30] M. Yan, H. Yuan, J. Xu, Y. Yu, and L. Jin, "Task allocation and route planning of multiple UAVs in a marine environment based on an improved particle swarm optimization algorithm," *EURASIP Journal on Advances in Signal Processing*, vol. 2021, no. 1, 23 pages, 2021.

[31] Z. Qu, G. Zhang, H. Cao, and J. Xie, "LEO satellite constellation for internet of things," *IEEE Access*, vol. 5, pp. 18391–18401, 2017.

[32] O. Kodheli, A. Guidotti, and A. Vanelli-Coralli, "Integration of satellites in 5G through LEO constellations," in *GLOBECOM 2017 - 2017 IEEE Global Communications Conference*, pp. 1–6, Singapore, December 2017.

[33] L. Boero, R. Bruschi, F. Davoli, M. Marchese, and F. Patrone, "Satellite networking integration in the 5G eco-system: research trends and open challenges," *IEEE Network*, vol. 32, no. 5, pp. 9–15, 2018.

[34] A. Guidotti, A. Vanelli-Coralli, M. Conti et al., "Architectures and key technical challenges for 5G systems incorporating

satellites," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2624–2639, 2019.

[35] J. Zhang, X. Zhang, P. Wang, L. Liu, and Y. Wang, "Double-edge intelligent integrated satellite terrestrial net-works," *China Communications*, vol. 17, no. 9, pp. 128–146, 2020.

[36] T. de Cola and I. Bisio, "QoS optimisation of eMBB services in converged 5G-satellite networks," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 10, pp. 12098–12110, 2020.

[37] S. Kota and G. Giambene, "6G Integrated Non-Terrestrial Networks: Emerging Technologies and Challenges," *2021 IEEE International Conference on Communications Workshops (ICC Workshops)*, pp. 1–6, 2021.

[38] Y. Wang, J. Zhang, X. Zhang, P. Wang, and L. Liu, "A computation offloading strategy in satellite terrestrial networks with double edge computing," in *2018 IEEE International Conference on Communication Systems (ICCS)*, pp. 450–455, Chengdu, China, December 2018.

[39] Z. Zhang, W. Zhang, and F. H. Tseng, "Satellite mobile edge computing: improving QoS of high-speed satel-lite-terrestrial networks using edge computing techniques," *IEEE Network*, vol. 33, no. 1, pp. 70–76, 2019.

[40] C. Suzhi, W. Junyong, H. Hao et al., "Space edge cloud enabling network slicing for 5G satellite network," in *2019 15th International Wireless Communications & Mobile Computing Conference (IWCMC)*, pp. 787–792, Tangier, Morocco, June 2019.

[41] L. Liu, J. Zhang, X. Zhang, P. Wang, Y. Wang, and L. Ouyang, "Design and analysis of cooperative multicast-unicast transmission scheme in hybrid satellite-terrestrial networks," in *2018 IEEE International Conference on Communication Systems (ICCS)*, pp. 309–314, Chengdu, China, December 2018.

[42] Y. M. Saputra, D. T. Hoang, D. N. Nguyen, and E. Dutkiewicz, "A novel mobile edge network architecture with joint caching-delivering and horizontal cooperation," *IEEE Transactions on Mobile Computing*, vol. 20, no. 1, pp. 19–31, 2021.

[43] S. Luo, X. Chen, and Z. Zhou, "F3C: fog-enabled joint computation, communication and caching resource sharing for energy-efficient IoT data stream pro-cessing," in *2019 IEEE 39th International Conference on Distributed Computing Systems (ICDCS)*, pp. 1019–1028, Dallas, TX, USA, July 2019.

[44] F. Kurniawan, L. Yovita, and T. Wibowo, "Modified-LRU algorithm for caching on named data network," in *2019 International Conference on Electrical Engineering and Informatics (ICEEI)*, pp. 438–443, Bandung, Indonesia, July 2019.

[45] P. Aimtongkham, C. So-In, and S. Sanguanpong, "A novel web caching scheme using hybrid least frequently used and support vector machine," in *2016 13th International Joint Conference on Computer Science and Software Engineering (JCSSE)*, pp. 1–6, Khon Kaen, Thailand, July 2016.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 5098992, 10 pages
https://doi.org/10.1155/2021/5098992

Research Article

"Avatar to Person" (ATP) Virtual Human Social Ability Enhanced System for Disabled People

Zichun Guo,[1] **Zihao Wang**,[2] **and Xueguang Jin**[3]

[1]*College of Information Science and Technology, Beijing University of Chemical Technology, Beijing 100029, China*
[2]*Edinburgh College of Art, The University of Edinburgh, Edinburgh EH17, UK*
[3]*College of Cyberspace Security, Changzhou College of Information Technology, Changzhou 213164, China*

Correspondence should be addressed to Xueguang Jin; jinxueguang@ccit.js.cn

Received 24 September 2021; Revised 29 October 2021; Accepted 13 November 2021; Published 3 December 2021

Academic Editor: Ming Yan

Copyright © 2021 Zichun Guo 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.

How to make communication more effective has been underlined unprecedentedly in the artificial intelligence (AI) era. Nowadays, with the improvement of affective computing and big data, people have generally adapted to construct social networks relying on social robots and smartphones. Although the technologies above have been widely discussed and used, researches on disabled people in the social field are still very limited. In particular, facial disabled people, deaf-mutes, and autistic patients are still meeting great difficulty when interacting with strangers using online video technology. This project creates a virtual human social system called "Avatar to Person" (ATP) based on artificial intelligence and three-dimensional (3D) simulation technology, with which disabled people can complete tasks such as "virtual face repair" and "simulated voice generation," in order to conduct face-to-face video communication freely and confidently. The system has been proven effective in the enhancement of the sense of online social participation for people with disabilities through user tests. ATP is certain to be a unique area of inquiry and design for disabled people that is categorically different from other types of human-robot interaction.

1. Introduction

Communication is the main component of interpersonal relationships, which defines the existence of human beings and serves as a vital connection between individuals. It is also one of the most important bases for promoting the development of human civilization. Communication can be divided into verbal and nonverbal forms, and its essence is to exchange information between the receiver and the sender. As the components of communication, verbal and nonverbal forms are separately regarded as spontaneous and disguised spontaneous communication, the former manifested as the intentional communication from the motivation-emotional state, and the latter as the instinctive intentional strategic operation [1]. In general, people focus more on language but lack attention to nonverbal ways of communication which psychologists believe can effectively convey instinctive emotions and that can be an indirect result of the main and noncommunicative functions in the communication process [2]. Verbal forms in this study refer to oral and written language, while nonverbal forms refer to facial expressions, body postures, and other bodily motions used to communicate with others.

Communication is a natural ability for most people. Nevertheless, congenital or acquired physical deformity people with impairments suffered from exacerbation of their communicative difficulties [3]. Psychological as well as physical problems make it more difficult for them to participate in societies, and so they are viewed as a distinct group [4]. Normalizing communication between impaired and healthy people is a tough process. Face-to-face communication allows people to have more intimate expressions during social interactions, but for most people with facial disabilities, this form of communication is unachievable [5].

Most of the previous studies focus on two aspects, the use of "social robots" and "smartphones" for communication. Robots were used to improve social skills and were generally used in education [6], autistic children [7], and elderly

care [8]. These researches mainly focus on the use of robots to complete the dialogue and interaction between humans and machines and then gradually improve the technical iterations of emotional computing and perception [9]. Although the interaction between humans and robots has been meaningfully explored with children, adults, and the elderly, the research on the interaction between people with disabilities and robots is still very limited [10]. This kind of interaction is not the same as normal cases when all the social interaction between humans and robots is conducted through emotional computing.

Apart from social robots, the popularity of smartphones has also greatly benefited people with disabilities, such as reducing their reliance on others, increasing their job competitiveness, encouraging them to participate more actively in public debates, and also building previously impossible communication channels for them and their peers [11]. In particular, we-media social media platforms have had a significant impact on the social and professional lives of individuals with disabilities since the product launch. The rise of blogs also provided disabled people with a new venue to share news, criticism, opinions, and personal experiences [12]. Disability culture on the internet grew in fruitful ways with the development of blogs. Moreover, disability is linked to both the built environment and the physical body [13]. Written language, devoid of emotional components, serves only as a regular way for exchanging information, while face-to-face video connection has become more emotive as we-media platforms have grown into varieties of social forms like live video streaming.

However, none of the social robots for social simulation nor self-media for text or voice communication could be accepted by certain groups of disabled people. Those with disfigurements and autism, for example, are less inclined to use the live video streaming platform to showcase their abilities like storytelling and singing. To hide their flaws, they prefer typing chat with strangers to face-to-face communication. Similarly, for deaf and dumb people, voice communication is also not available. In general, these specific groups include three categories of people: (1) individuals with facial injuries or peripheral facial palsy who are reluctant or afraid to reveal their handicapped faces in front of the camera [14]. The face is an essential aspect of humans' personality and body image. The psychological factor is one of the main factors that influence the healing and rehabilitation of face trauma [15]; (2) deaf-mutes lack sound judgment due to their congenital or acquired deafness and cannot pronounce correctly either; (3) people with cerebral palsy and autism suffering from brain lesions and psychological trauma which prevent their facial muscles from moving as they want are unable to convey information properly through facial expressions. The three categories of people are illustrated in Figure 1.

Based on the specific context, the model of virtual avatar social system called the Avatar to Person (ATP) is proposed as a video social means to bridge the gap between virtual people and actual people. Powered by artificial intelligence, the system is aimed at improving the experience for people with disabilities in online video meetings. All the communication ways, including text, voice, facial expressions, and body language, are supported by the avatar, while three main functions, including facial expression augmentation, text-to-facial expression, and facial repairment, are supported by the virtual avatar social system.

This paper will firstly describe present impediments to communication for disabled people; secondly discuss the technological possibility, user psychology, and user experience; and thirdly evaluate the system's possible future of real-time usage in ATP applications.

2. Materials and Methods

2.1. Research and Design across Disciplines. Communication, artificial intelligence, and animation design are integrated in the process of the model creation and development of the ATP model. For example, owing to the rapid development of deep learning technology and its outstanding performance in the field of artificial intelligence, facial restoration technology has made tremendous progress in feature recognition, mapping, and modelling of pain intensity from face pictures [16]; affective computing is a collection of approaches for extracting effect from data in many modalities and at various granularity scales, aimed at developing a system that can recognize, interpret, process, and simulate human emotions and provides the possibility of emotion design of virtual avatars [17]; and animation technology could be utilized to build virtual avatars with different styles to satisfy various requirements of people with disabilities.

In this project, the model design of ATP is based on the theoretical context of the traditional Person to Person (PTP) transmission mechanism put forward by Sudweeks et al., which can be divided into three ways: one-way communication, two-way communication, and interactive communication (as shown in Figure 2) [18].

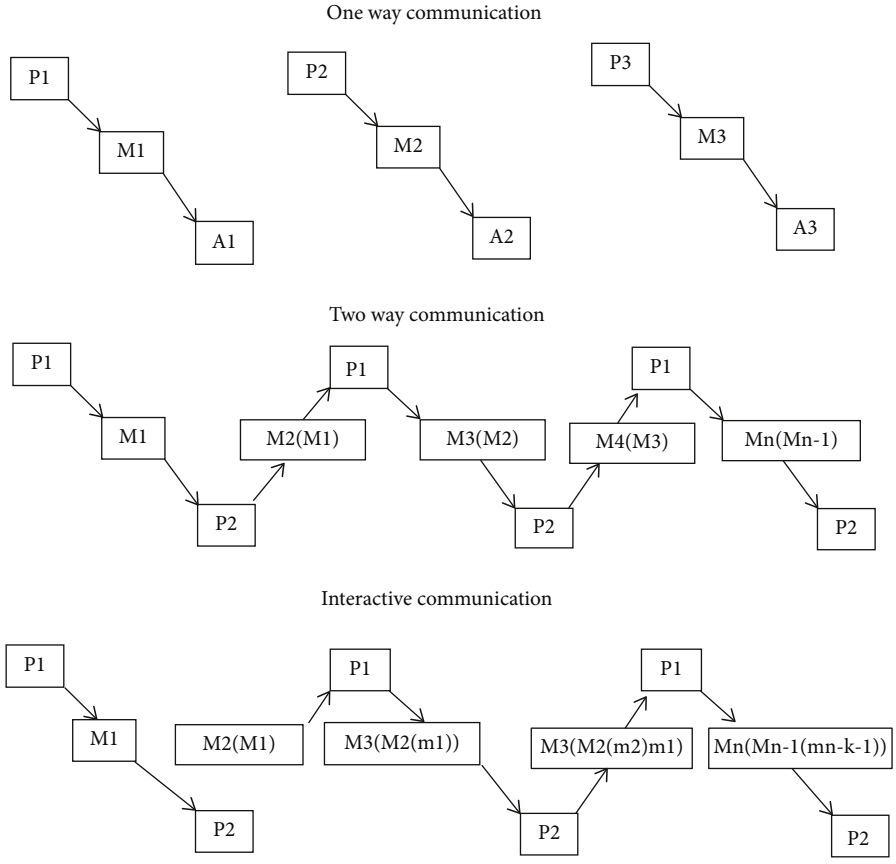
The ATP model, as shown in Figure 3, refers to the social interaction between a real human and a virtual avatar, in which users may transform their physical appearance into a virtualized avatar.

2.2. Content Design. The project focuses on the three categories of handicapped individuals with difficulty in face-to-face online communication. They are people with autism, deaf and dumb people, and people with facial disabilities.

2.2.1. Disabled People with Autism. Autism often occurs at a young age, and the symptoms mainly include communication disorders, language communication disorders, and stereotyped repetitive behaviours [19]. Compared with normal children, autistic children are always more flat or neutral in their expression, which implies the difficulties they have in sharing their affection and making their affective signals to be understood [20]. The current *Diagnostic and Statistical Manual of Mental Disorders* (*DSM*) diagnostic criteria for autism have included items concerned with problems in perceiving and processing emotions: "marked impairments in the use of multiple nonverbal behaviours, such as … facial expression …" and "lack social or emotional reciprocity" [21].

Facial trauma Deaf-mutes people Autism

Figure 1: "Certain handicapped people" may not be willing to use a social robot to perform a social simulation or use self-media for voice communication.



P: Person
M: Message (numbered by temporal sequence)
A: Audience

Figure 2: The Person to Person (PTP) transmission mechanism.

The capacity to identify emotions from facial expressions and then to give proper feedback is such an essential tool in successful social interaction [22]. The inability to recognize facial expressions hinders the social interaction of people with autism. Therefore, it is necessary to take some measures to interfere in autistic children's facial expression recognition, and the way to augment patients' poor facial expressions has become one of the most urgent problems that need to be solved by virtual avatars.

Owing to the emergence of artificial intelligence, autistic individuals may get a helping hand. Face recognition and sentiment analysis powered by artificial intelligence can augment users' facial expressions in online face-to-face meetings, helping autistic individuals recognize and

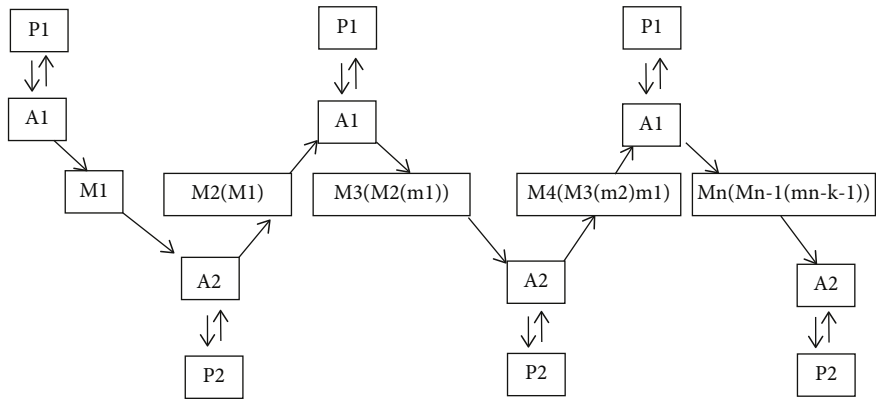


Figure 3: The PTP communication pattern.

comprehend the facial expression of the person speaking to them. The desired result can be accomplished in two ways. One is facial expression recognition augmentation, which applies AI technology to the process of facial expression recognition of autistic people and then uses affective computing to extract expression labels to distinguish their facial expressions, while the other is expression intervention augmentation, which refers to performing emotion recognition by extracting autistic patients' weak physical expressions and then augmenting them in the virtual humanization process.

2.2.2. Disabled People with Deafness. Although deaf and dumb people can communicate with writing, they cannot still mingle with the social world due to the lack of face-to-face communication abilities. There are more than 450 million deaf and dumb people all over the world, which takes up 5% of the world's population. Despite the great number, there is quite little research to bridge the communication barrier between deaf and dumb people and the social world [23]. Therefore, an efficient system must be set up to help them communicate and interact freely.

As for this project, virtual avatars are introduced to help deaf and dumb people overcome the difficulty of pronunciation through text-to-speech and facial expression animation. The mouthpiece translation technology is applied to help deaf and dumb individuals improve their capacity to express themselves clearly. The text input is converted into the information in voice and mouth type, and the user's facial expressions are complemented with the avatar. Furthermore, for congenital deaf-mutes, the system can provide AI voices with various voice styles, while for acquired deaf-mutes, the system will conduct AI training on users' previous voice samples to simulate and generate their inborn voice.

2.2.3. People with Facial Disabilities. Facial disability results in deficits that are necessary for both physical functions like eating and drinking and social functions and mental health like conveying conversational signals and intimate human information [24]. Macgregor emphasized that social interaction during brief encounters in daily life causes people with facial disabilities continuous stress, affliction, and anxiety [25]. Facial disability makes them feel strange and fearful when showing their faces and leads to their inner feelings of inferiority, thus causing problems like social isolation and addiction [26].

Given the psychological barrier that people with facial impairments experience, we propose virtual avatars to restore their faces during online video communications, in which way their willingness to engage themselves in online meetings with strangers would be increased. In order to create virtual avatars, a facial repairing system, an intervention technique for facially disabled people, could be used. The 3D avatar is built by AI based on a database of millions of thousands of samples of predisabled pictures. Users can then use the real-time face capture technology to synchronize their facial expressions with those of the 3D virtual avatars in the video communication process, in which way the barriers between people with facial disabilities and the social world can be bridged.

2.3. System Design

2.3.1. System Overview. Figure 4 depicts the framework of the proposed system. Firstly, the system creates high-fidelity animation to reflect the emotion and look of handicapped individuals properly. Secondly, the system retargets personal characters through various speech styles and animation synced with the vocal input, in which way the AI voice output is shown in accordance with the emotional expression. Finally, the system renders quickly, with the goal of realizing real-time animation.

2.3.2. Mental Cognitive System. The mental cognitive system model used in the paper was designed by Baron-Cohen [27]. 412 emotions divided into 24 emotional categories and 6 developmental levels (from 4 to maturity) are considered. In the phase of the face detection stage, facial expression

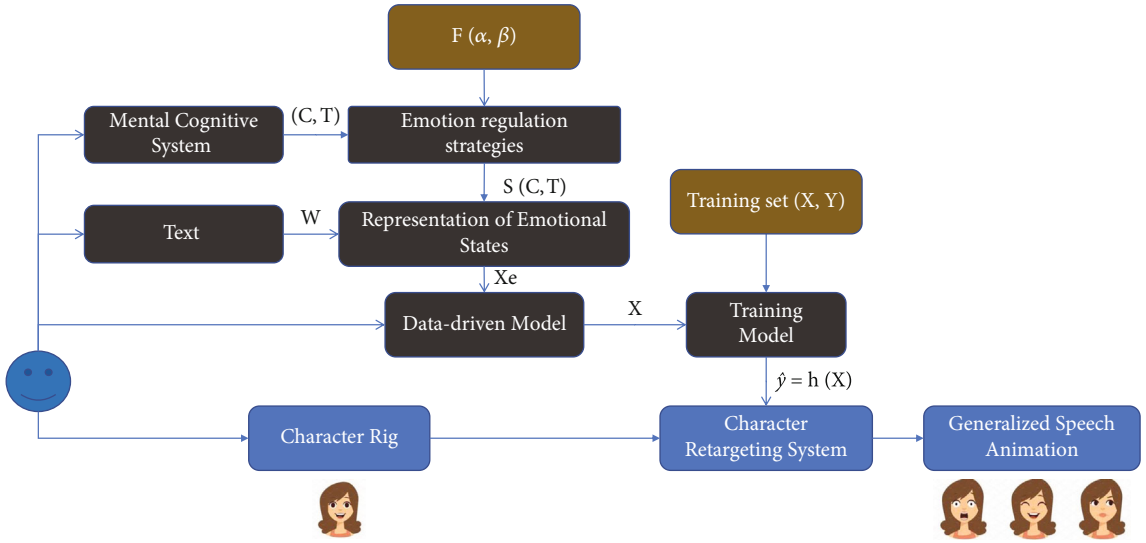


Figure 4: The framework of the ATP social system.

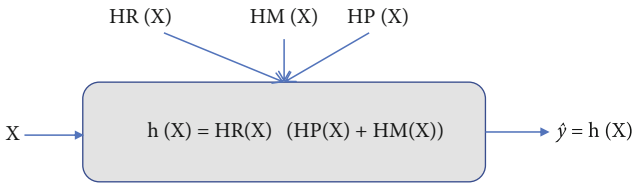


Figure 5: The training model.

classification C and recognition template T are taken into account.

2.3.3. Emotion Regulation Strategy. Emotion regulation is an influential factor in many problems that blind people suffer from [28]. Interventions that target emotion regulation strategies would be useful [29]. For blind people, cognitive reappraisal and expressive suppression are less applied. Emotion regulation strategy distribution function $F(\alpha, \beta)$ is defined as

$$N(\mu_1, \mu_2, \sigma_1^2, \sigma_2^2, \rho), \tag{1}$$

where μ_1 and σ_1^2 are the normal distribution factors of cognitive reappraisal and μ_2 and σ_2^2 are the normal distribution factors of expressive suppression. These parameters can be adjusted by the system.

For emotion regulation response generation, given a query $q(C, T)$, an emotion category $e(C, T)$, and emotion regulation strategy $F(\alpha, \beta)$, the goal is to generate a response $s(C, T)$ that is not only meaningful but also in accordance with the desired normal people emotion.

The Emotional Chatting Machine (ECM) addresses the emotion factor with three new mechanisms: Emotion Category Embedding, Internal Memory, and External Memory [30]. Specifically, (1) Emotion Category Embedding models the high-level abstraction of emotion expression by embedding emotion categories and concatenates the corresponding embeddings to the input at each decoding step. (2) Internal Memory captures the change of implicit internal emotion states with reading and writing gates. (3) External Memory applies an external emotion vocabulary to express emotions explicitly and finally assigns different generation probabilities to emotion and generic words.

The training sample (q, s) is defined as

$$\begin{aligned} q(C, T) &= (q(C_1, T_1, q(C_2, T_2) \cdots q(C_m, T_m)), s(C, T) \\ &= (s(C_1, T_1, q(C_2, T_2) \cdots s(C_m, T_m)))). \end{aligned} \tag{2}$$

The emotion regulation strategy distribution function $F(\alpha, \beta)$ is calculated through a training sample.

The loss function of the training sample (q, s) is defined as

$$-\sum_{t=1}^{m} p_t \log(o_t) - \sum_{t=1}^{m} q_t \log(\alpha_t) + \left\| M^I_{s(C,T),F(\alpha,\beta),m} \right\|, \tag{3}$$

where o_t and p_t are the predicted token distribution and gold distribution, α_t is the probability of choosing an emotion word or a generic word, $q_t \epsilon \{0, 1\}$ is the true choice between them in $s(C, T)$, and $M^I_{s(C,T),F(\alpha,\beta),m}$ is the internal emotion state at the last step m with emotion regulation strategy distribution function $F(\alpha, \beta)$. The first term is the cross-entropy loss, the second is used to supervise the probability of selecting an emotion or a generic word, and the last is used to ensure that the internal emotional state has been expressed completely once the generation is finished.

2.3.4. Emotional State Representation. Auxiliary emotional data derived from facial, vocal, textual, or mixed data is fed into the deep learning network and training samples, along with a tiny quantity of supplementary emotional data. The deep learning network has sufficient data to infer the optimal result. All relevant aspects of the animation should be encoded following the probability distribution of the given training samples, in accordance with the additional

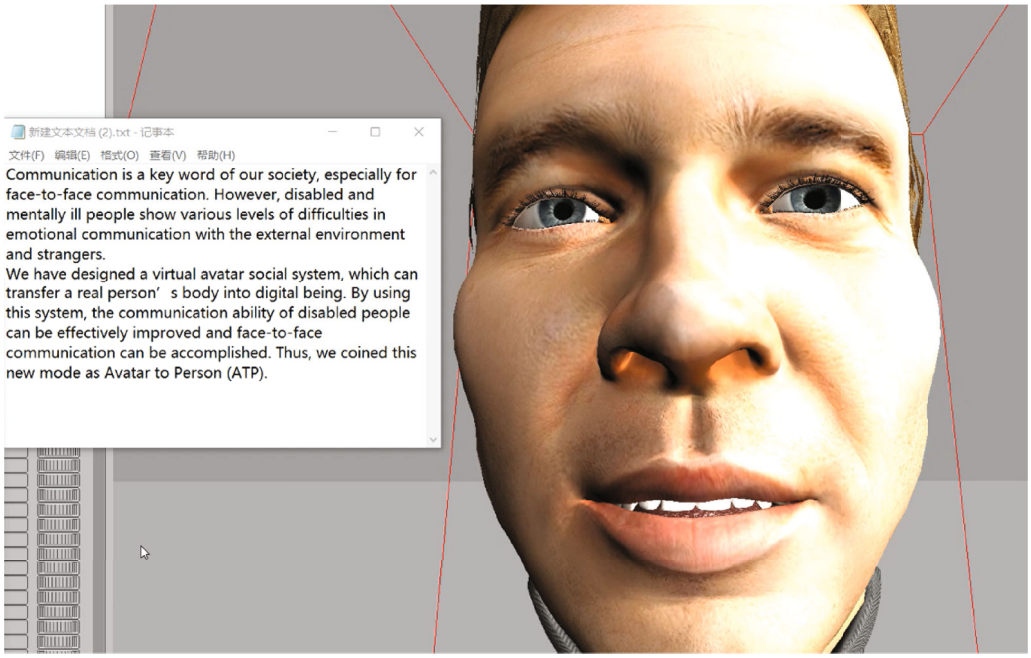


Figure 6: Output facial expressions corresponding to the deaf people's inputting text to the ATP system.

Table 1: Lessons for experiment.

Lesson	Lesson components	Reciprocity components
Having a respectful attitude	(i) Talk about feelings when upset (ii) Use request words and tone rather than demand words (iii) Compromise or accommodate others (iv) Ask permission to touch something not yours	Request Compromise
Getting to know someone new	(i) Introduce yourself (ii) Talking about some interesting thing (iii) Asking question	Ask Respond Share information Introduce oneself
Compromising	(i) Ask what another person wants (ii) Tell what you want (iii) Offer to do some of what you and another person want	Ask Share information

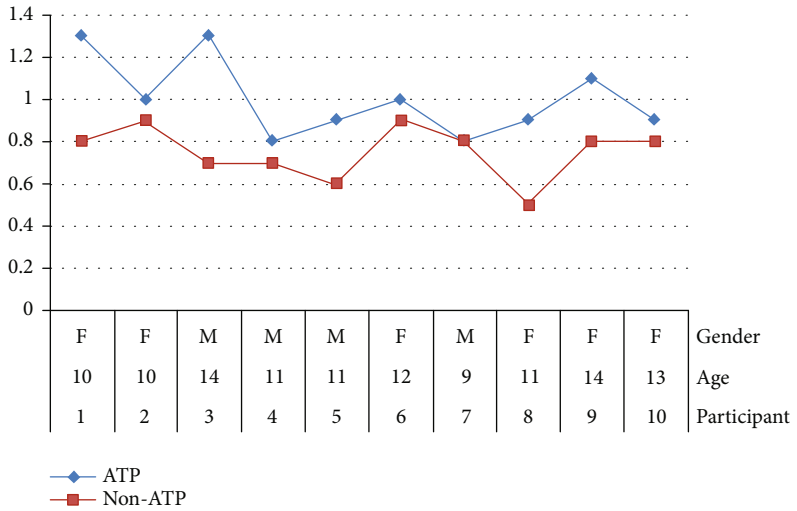


Figure 7: Data for ATP and non-ATP.

Table 2: Test result for deaf-mutes.

Participant	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
ATP	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S	S
Non-ATP	F	S	F	S	S	F	F	F	F	S	S	F	F	F	F	F	S	F

S means information transfer completed successfully; F means information transfer failure.

emotional data. These training examples are based on a variety of variables, including facial expressions, speaking styles, and collaborative pronunciation patterns, rather than the audio itself. As a result, the emotional state of the actor is sampled. On a given vocal cord, auxiliary emotional data input is highly beneficial for reasoning to blend and match different emotional states.

Commercial software such as Baidu AI, manual tagging, and training sample classification can be applied to achieve the emotional state of disabled people [31].

2.3.5. Data-Driven Model. In the data-driven model, some deep neural networks (the grey part of the system) are modelled for learning effectively from appropriate training sets with a large variety of speech patterns. The input audio, facial, and test data are trained. It builds the formant analysis network to extract feature sequences and emotional expression labels at the same time. Thus, short-term features related to the disabled people's facial animation are extracted by the convolution layer, and the features include emphasis, specific phonemes, and intonation. And the temporal evolution of facial features is generated by another convolutional neural network. There is an audio window related to the abstract feature vectors of facial posture. Ambiguities between different facial expressions and speech styles are eliminated by the emotional expression as another input. E-dimensional vectors are used to express emotional states, which are directly connected to the output of each layer in the neural network so that the subsequent layers can change their behaviour accordingly. With the model of Active Appearance Model (AAM), the mouth features are shaped with 34 component vertices and face features are shaped with 108 vertices. Most importantly, the learning network executes the same operation on every point on the same timeline. As a result, the same training samples with different timeline offsets are generated.

With small learning faces, the specific effect of the speaker is controlled, and the details of the captured animation are focused. Cost-effective collection of the appropriate training set captures a wide range of speech patterns. This method meets the request of retarget ability and edit ability.

2.3.6. System of Character Retargeting. The speech animation with specific style settings for disabled people is generated by the character retargeting system (blue part of the system), and X denotes a sequence of inputs of the animation. The true prediction animation sequence Y is constructed by a predictor that predicts any input X. Y is corresponding to the specific reference facial model. The training set (X, Y) is collected from the reference facial data set. In general, h is nonlinear which is learned by the deep learning network to obtain complex nonlinear mappings from X to Y. After getting the mapping h, with the retargeting function, the parameters obtained from the reference face model can be performed on any parameterization precalculation character model. Thus, automatic and quick retargeting can be done.

The operation process of the data is as follows: (1) Phoneme sequence X is generated by input audio (e.g., speech recognition software). (2) $h(X)$ is used to predict the animation parameter Y corresponding to the reference mouth and face model associated with input X. (3) Y in the reference animation model is retargeted to the target character model. Thus, the accompanying audio speech animation in real time is generated by the model. The Hidden Markov Model Toolkit (HTK) or artificial translating is used for speech-to-text translating.

2.3.7. Data Training Model. The loss function is defined to optimize and train the deep learning network as is shown in Figure 5 [32]. It has three different constraints. The position function ensures that each output vertex position is roughly correct. The motion function ensures that the vertex follows the correct motion type in the animation. The regularization function ensures that the emotion short-term changes are well controlled according to the correct animation position and direction of motion.

For a given training sample X, we define the loss function $h(X)$ as

$$h(X) = HR(X)(HP(X) + HM(X)),$$

$$HP(X) + HM(X) = \frac{1}{3D}\sum_{i=1}^{3D}\left(\left(Y^{(i)}(X) - Y\wedge^{(i)}(X)\right)^2 + \left(M\left[Y^{(i)}(X) - M\left[Y\wedge^{(i)}(X) - M\left[Y\wedge^{(i)}(X)\right]\right]\right)^2,$$

$$HR(X) = \frac{1}{E}\sum_{i=1}^{E}\left(M\left|E^{(i)}(X)\right|\right)^2. \quad (4)$$

$HP(X)$ represents the position factor, and $HM(X)$ represents the motion factor. $Y(i)$ represents the ith scalar component of $Y = (Y(1), Y(2), \cdots, Y(3V))$, in which V is the total number of output vertices related to the character face. Thus, for each vertex, the total number of the character's output components is $3V$ with the 3D position.

$HP(X)$ ensures that the deep learning network output is fully synchronized without promoting the quality of animations. To promote the quality of animation, it was optimized from the perspective of vertex motion. Therefore, the motion item $HM(X)$ was quoted to qualify the difference between frames in the operation M [33]. In the simulation

procedure, the data driving model network correctly reflects the emotion expected and the regularization term ensures $HR(X)$ is running at the same finite differencing operator as above.

The Adam optimization method was adopted to balance the previous three-loss terms [34]. The weight of the network is updated by Adam with the gradient of the loss function and is normalized based on the long-term estimation. Normalization applies to the overall loss function, but not to individual terms. Output facial expression corresponding to the deaf people's inputting text to the ATP system is shown in Figure 6.

3. Results and Discussion

All three groups of impaired people, disabled people with autism, deaf and facial abilities should have participated in the ATP intervention study. Unfortunately, only autistic individuals and deaf-mutes are currently willing to participate in the experiment. We met great difficulty in conducting user testing among people with facial trauma. In this case, the results are confined to evaluating the efficacy of ATP in groups of autistic individuals and deaf-mutes.

3.1. Assessment Design for Autism. The metrical model of the experiment is based on the structured learning program proposed by Baker and Myles in the book [35] *Social Skills Training for Children and Adolescents with Asperger Syndrome and Social Communication Problems*, which includes 70 specific social skills that are most likely to cause autism spectrum disorders and communication problems. The metrical model is on the basis of the three lessons in the book from the communication aspect (see Table 1).

3.1.1. Participants. Ten autistic individuals (mean age = 11.5; SD = 2.65; ranging from 9-14), composed of 6 females and 4 males, were invited to test the effectiveness of ATP on disabled people with autism. The physical age of all ten participants had been confirmed that is matched for verbal mental age.

3.1.2. Procedure. 30 questions are designed to evaluate three lessons, each of which contains ten social issues. The scores for these questions represent the level of communication skills. A full score is three points ($a = 3$), while a score of 1.8 is considered typical.

For the test, 10 participants, ranging in age from 9 to 14, are invited. Each participant was tested four times, and each time lasts 30 minutes. It took four days to complete the entire test. For each test, the participant was asked to communicate with strangers through real-time video, and all the tests were performed by the same speech-language pathologist in connection with customary medical visits to the clinic.

The experiment set both an experiment group and a no-treatment group. During the four-day testing process, 10 participants needed to complete three Baker and Myles tests without ATP in the first two days, as the no-treatment group, and then in the next two days, they were asked to complete the same test under the intervention of ATP, as the experiment group. The result of the experiment is shown in Figure 7.

According to the result, the scores of the ATP group are significantly higher than those of the non-ATP group, which indicates that the ATP system can effectively improve the video social ability of autistic children.

3.2. Evaluation Design for Deaf-Mutes. As for the test with deaf and dumb people, 18 participants were required to convey specific information, the meaning of "panda," to normal people through video social networking software. The whole communication process was only allowed to conduct on the video software. Written language was prohibited.

The same as the form of tests on autistic individuals, two groups were set for comparison. In the ATP group, participants entered text into the ATP system, and the avatar generated a spoken animation based on the text and presented it to the recipient; in the non-ATP group, participants were free to transmit nontext information in any way they wanted, such as body language and facial expressions. Table 2 displays the outcome.

The test results show that all participants in the ATP group can successfully communicate information through online video socialization, while few participants in the non-ATP group succeeded in the transmission of the information, which can verify the effectiveness of ATP.

4. Conclusions

Most of the previous studies about social skills assisting communication always focus on children and ageing people with "social robots" and "smartphones," while studies on disabled people have just started to attract the attention of researchers. There is currently a lack of longitudinal research on developing online social tools for disabled people with ATP due to the nonexistence of flexible and effective ATP providing consistent and regular communication. On the contrary, this study showed that ATP would be beneficial for the improvement of online social skills in autistics and deaf-mutes. The ATP social system provides a platform for autistic individuals, deaf and dumb people, and people with facial disabilities, to help them realize free expression and evaluation through online video socialization, and has been verified effective in groups of autistic individuals and deaf-mutes but needs further tests on disabled groups with facial disabilities.

It is the first step for us in applying digital being research to work for impaired individuals. In the future, based on the existing ATP social system, we will continue our journey to develop frameworks including (1) avatar for people social system design strategies in the community and (2) user-generated content for handicapped individuals by ATP. Furthermore, we will go on the exploration of future trends and research areas in public health systems.

Data Availability

The data and model used to support the findings of this study are restricted by the College of Information Science

and Technology of Beijing University of Chemical Technology. Data are available from the corresponding author for researchers who meet the criteria for access to confidential data.

Acknowledgments

This research was supported by the National Science Foundation for Postdoctoral Scientists of China (Grant No. 2021M700355), by the Changzhou Sci&Tech Program (Grant No. CE20212025), by the General Project of Higher Education Reform Research in Jiangsu Province (Grant No. 2021JSJG521), by the Innovation Application Centre of Changzhou College of Information Technology (Grant No. CCIT2021STIT010202), by the Natural Science Foundation of the Jiangsu Higher Education Institutions of China (Grant No. 20KJB140013), and by the Natural Science Foundation of Jiangsu Province, youth project (Grant No. BK20200190).

Disclosure

This research is based on our previous teamwork, *Avatar Social System Improve Perceptions of Disabled People's Social Ability*, presented in 2019 IEEE/ACIS 18th International Conference on Computer and Information Science (ICIS). Rui Hao has also made contribution to the research.

Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this article.

References

[1] R. Buck and C. VanLear, "Verbal and nonverbal communication: distinguishing symbolic, spontaneous, and pseudo-spontaneous nonverbal behavior," *Journal of Communication*, vol. 52, no. 3, pp. 522–541, 2002.

[2] R. Zajonc, S. Murphy, and M. Inglehart, "Feeling and facial efference: implications of the vascular theory of emotion," *Psychological Review*, vol. 96, no. 3, pp. 395–416, 1989.

[3] I. K. Zola, "Communication barriers between 'the able-bodied' and 'the handicapped'," *Archives of Physical Medicine and Rehabilitation*, vol. 62, no. 8, pp. 355–359, 1981.

[4] A. Sheldon, *Disabled people and communication systems in the twenty first century, [Ph.D. thesis]*, The University of Leeds, 2001.

[5] M. Argyle, M. Cook, and D. Cramer, "Gaze and mutual gaze," *British Journal of Psychiatry*, vol. 165, no. 6, pp. 848–850, 1994.

[6] A. Sharkey, "Should we welcome robot teachers?," *Ethics and Information Technology*, vol. 18, no. 4, pp. 283–297, 2016.

[7] S. Costa, F. Soares, A. Pereira, C. Santos, and A. Hiolle, "A pilot study using imitation and storytelling scenarios as activities for labelling emotions by children with autism using a humanoid robot," in *4th International Conference on Development and Learning and on Epigenetic Robotics*, pp. 299–304, Genoa, Italy, 2014.

[8] M. Coeckelbergh, ""How I learned to love the robot": capabilities, information technologies, and elderly care," in *The capability approach, technology and design*, pp. 77–86, Springer, Dordrecht, 2012.

[9] J. Peter, R. Kühne, A. Barco, C. de Jong, and C. van Straten, "Asking today the crucial questions of tomorrow: social robots and the internet of toys," in *The Internet of Toys*, pp. 25–46, Palgrave Macmillan, Cham, 2019.

[10] E. Björling, E. Rose, A. Davidson, R. Ren, and D. Wong, "Can we keep him forever? Teens' engagement and desire for emotional connection with a social robot," *International Journal of Social Robotics*, vol. 12, no. 1, pp. 65–77, 2020.

[11] A. Martinez-Cabrera, "Web more accessible to those with disabilities," *San Francisco Chronicle*, vol. 1, 2010.

[12] R. Anderson, B. Stenger, V. Wan, and R. Cipolla, "Expressive visual text-to-speech using active appearance models," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3382–3389, Portland, USA, 2013.

[13] J. Hunsinger and T. Senft, *The Social Media Handbook*, Routledge, 2013.

[14] T. Movérare, A. Lohmander, M. Hultcrantz, and L. Sjögreen, "Peripheral facial palsy: speech, communication and oral motor function," *European Annals of Otorhinolaryngology, Head and Neck Diseases*, vol. 134, no. 1, pp. 27–31, 2017.

[15] A. De Sousa, "Psychological issues in acquired facial trauma," *Indian Journal of Plastic Surgery*, vol. 43, no. 2, p. 200, 2010.

[16] G. Bargshady, X. Zhou, R. Deo, J. Soar, F. Whittaker, and H. Wang, "Enhanced deep learning algorithm development to detect pain intensity from facial expression images," *Expert Systems with Applications*, vol. 149, article 113305, 2020.

[17] E. Yadegaridehkordi, N. Noor, M. Ayub, H. Affal, and N. Hussin, "Affective computing in education: a systematic review and future research," *Computers & Education*, vol. 142, article 103649, 2019.

[18] F. Sudweeks, M. McLaughlin, and S. Rafaeli, *Network and Netplay*, AAAI Press, Menlo Park, CA, 1998.

[19] S. Rogers, "Interventions that facilitate socialization in children with autism," *Journal of Autism and Developmental Disorders*, vol. 30, no. 5, pp. 399–409, 2000.

[20] N. Yirmiya, C. Kasari, M. Sigman, and P. Mundy, "Facial expressions of affect in autistic, mentally retarded and normal children," *Journal of Child Psychology and Psychiatry*, vol. 30, no. 5, pp. 725–735, 1989.

[21] American Psychiatric Association, *Diagnostic and statistical manual of mental disorders*, American Psychiatric Association, Washington, DC, 1980.

[22] M. Harms, A. Martin, and G. Wallace, "Facial emotion recognition in autism spectrum disorders: a review of behavioral and neuroimaging studies," *Neuropsychology Review*, vol. 20, no. 3, pp. 290–322, 2010.

[23] P. Ahire, K. Tilekar, T. Jawake, and P. Warale, "Two way communicator between deaf and dumb people and normal people," in *2015 International Conference on Computing Communication Control and Automation*, pp. 641–644, Pune, India, 2015.

[24] J. VanSwearingen and J. Brach, "The facial disability index: reliability and validity of a disability assessment instrument for disorders of the facial neuromuscular system," *Physical Therapy*, vol. 76, no. 12, pp. 1288–1298, 1996.

[25] F. Cooke Macgregor, "Facial disfigurement: problems and management of social interaction and implications for mental health," *Aesthetic Plastic Surgery*, vol. 14, no. 1, pp. 249–257, 1990.

[26] J. Valls-Solé, "Electrodiagnostic studies of the facial nerve in peripheral facial palsy and hemifacial spasm," *Muscle & Nerve*, vol. 36, no. 1, pp. 14–20, 2007.

[27] S. Baron-Cohen, H. Ring, J. Moriarty, B. Schmitz, D. Costa, and P. Ell, "Recognition of mental state terms," *British Journal of Psychiatry*, vol. 165, no. 5, pp. 640–649, 1994.

[28] H. Cai, H. Chen, C. Zhang et al., "Learning from easy to complex: adaptive multi-curricula learning for neural dialogue generation," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 34, no. 5, pp. 7472–7479, 2020.

[29] S. Salimi, E. Mohammadi, and A. Sadeghi, "A comparison of emotion regulation strategies of blind students with sighted students," *Iranian Rehabilitation Journal*, vol. 14, no. 2, pp. 127–132, 2016.

[30] X. Zhou and W. Wang, "MojiTalk: generating emotional responses at scale," 2018, https://arxiv.org/abs/1711.04090.

[31] K. Ellis and G. Goggin, "Disability media participation: opportunities, obstacles and politics," *Media International Australia*, vol. 154, no. 1, pp. 78–88, 2015.

[32] M. Yan, S. Li, C. A. Chan, Y. Shen, and Y. Yu, "Mobility prediction using a weighted Markov model based on mobile user classification," *Sensors*, vol. 21, no. 5, p. 1740, 2021.

[33] M. Yan, H. Yuan, Z. Li, Q. Lin, and J. Li, "Energy savings of wireless communication networks based on mobile user environmental prediction," *Journal of Environmental Protection and Ecology*, vol. 22, no. 1, pp. 206–217, 2021.

[34] M. Yan, H. Yuan, J. Xu, Y. Yu, and L. Jin, "Task allocation and route planning of multiple UAVs in a marine environment based on an improved particle swarm optimization algorithm," *EURASIP Journal on Advances in Signal Processing*, vol. 2021, Article ID 94, 2021.

[35] J. Baker and B. Myles, *Social Skills Training for Children and Adolescents with Asperger Syndrome and Social-Communication Problems*, Autism Asperger Publishing Company, Shawnee Mission, KA, 2004.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 3364680, 9 pages
https://doi.org/10.1155/2021/3364680

Research Article

Enhancing Feature Point-Based Video Watermarking against Geometric Attacks with Template

Zhongze Lv[1,2] **Hu Guan**[1] **Ying Huang**[1] **Shuwu Zhang**[1] **and Yang Zheng**[1]

[1]*Beijing Engineering Research Center of Digital Content Technology, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China*
[2]*School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China*

Correspondence should be addressed to Ying Huang; ying.huang@ia.ac.cn

Received 26 September 2021; Accepted 3 November 2021; Published 19 November 2021

Academic Editor: Ivan Lee

Copyright © 2021 Zhongze Lv 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.

As the Internet and communication technologies have developed quickly, the spread and usage of online video content have become easier, which results in major infringement problems. While video watermarking may be a viable solution for digital video content copyright protection, overcoming geometric attacks is a significant challenge. Although feature point-based watermarking algorithms are expected to be very resistant to these attacks, they are sensitive to feature region localization errors, resulting in poor watermark extraction accuracy. To solve this issue, we introduce the template to enhance the location accuracy of feature point-based watermarking. Furthermore, a scene change-based frame allocation method is presented, which arranges the template and the watermark to be embedded into different frames and eliminates their mutual interference, enhancing the performance of the proposed algorithm. According to the experimental results, our algorithm outperforms state-of-the-art methods in terms of robustness against geometric attacks under close imperceptibility.

1. Introduction

Copyright infringement occurs as a result of the rapid development of the Internet [1, 2], which makes it easy to disseminate and utilize digital media assets such as images, videos, and audios. Watermarking, on the other hand, may offer a solution for copyright tracking and verification, which is an algorithm that imperceptibly embeds a watermark containing copyright information into digital content. When copyright complications arise, the watermark is extracted to establish the ownership of the creator. Moreover, the focus of this paper is on video watermarking [3–12], which employs video as its carrier.

Throughout the Internet transmission, several deliberate or unintentional attacks on the watermarked digital video may occur. Common types of attacks include signal processing attacks (adding noise, filtering, transcoding, and so on), frame adjustment attacks (frame insertion, frame dropping, and so on), and geometric attacks (scaling, cropping, and so on). By the way, watermark energy is reduced by signal processing attacks, frame adjustment attacks change the quantity and relative placement of frames carrying the watermark, and geometric attacks cause the position of the watermark to be desynchronized between embedding and extraction. Most existing video watermarking methods own good robustness against signal processing and frame adjustment attacks, but they are usually vulnerable to geometric attacks.

Because a video can be regarded as a temporally continuous collection of still images, most image watermarking schemes [13–29] are also applicable for video watermarking. Existing image watermarking algorithms offer some ideas for resisting geometric attacks, and video watermarking can use these as references. In addition, feature point-based watermarking [13–21] is a common type of image watermarking against geometric attacks, but it suffers from the watermark location inaccuracy. This paper is aimed at extending feature point-based image watermarking to video watermarking and addressing its existing issue, in order to improve its resistance against geometric attacks.

Feature point-based watermarking extracts feature points from images or frames and uses them to locate nonoverlapping local regions named as feature regions, where the watermark is inserted. The feature regions can be kept as unaltered as feasible before and after geometric attacks by utilizing the invariance of the feature points. The scale information and the spatial coordinate of their associated feature points determine the location and the size of the feature regions; thus, the robustness of the feature regions is largely dependent on the stability of the scale information and the spatial coordinate. However, it is difficult to reappear the picked locations unbiased after suffering geometric attacks, resulting in feature regions locating mistake and a serious decline in the watermark extraction accuracy.

In order to improve the location accuracy, we introduce the template [22–25], which can recover geometrically distorted frames and help feature points in locating the feature regions. To reappear the feature regions from the image which have already recovered by the template, the scale information of the feature points is no longer required, but only the spatial coordinate is required. As a result, the inaccuracy in locating the feature regions can be significantly minimized.

The template is embedded in an image before embedding the watermark, and it is extracted in a possible damaged watermarked image to acquire the affine transform parameters in the extraction procedure. The damaged image is then restored to its original shape, allowing the watermark position to be synchronized between embedding and extraction. However, the watermark and the template are both embedded in the same image at the same time in the existing study, and they will interact with each other. Hence, the algorithm cannot get an ideal performance.

In summary, this paper contributes to the ongoing studies in the following two ways:

(1) A scene change-based frame allocation strategy is presented. This strategy can effectively arrange different frames to embed the watermark and the template separately, eliminating their mutual interference

(2) A video watermarking scheme against geometric attacks is designed, combining with the feature points and the template. The template aids feature points in locating the feature regions, decreasing the locating errors and increasing the robustness

After a series of tests on real-world datasets, we find that our approach outperforms the state-of-the-art approaches in terms of robustness, under good imperceptibility.

2. Related Work

The earliest research on video watermarking is in 1994. Matsui et al. [12] consider the video as a collection of sequential images in a specified order and then embed the watermark in these images. A complete video watermarking framework addresses three issues: frame selection for embedding and extraction, embedding region determination, and embedding and extracting scheme design. Selection of frames for embedding and extraction is a unique task of video watermarking, differing from other digital media watermarking, and it consists of selection of the whole [12] or partial frames (such as I-frames [6, 7], keyframes [8, 9], and scene change frames [10, 11]). The processes of determining embedding regions and designing embedding and extraction schemes are both performed on the selected frames so that the image watermarking algorithms can be used as references. Based on distinct embedding regions, the existing watermarking can be classified into global and local watermarking algorithms. The embedding process of the global watermarking algorithms employs the whole pixels of an image or a frame, making them vulnerable to cropping attacks. The local watermarking algorithms usually select the embedding regions by exploiting the feature point invariance, to make these regions roughly constant before and after attacks. Furthermore, the former has worse imperceptibility than the latter, and the correctness of watermark extraction is related to the precision of locating embedding regions. Quantization [5–7, 26, 27] and spread spectrum [28, 29] are two common types of embedding and extraction schemes. Quantization uses various quantizers to quantize the original carrier data into various index intervals, and the watermark information is extracted based on the index interval to which the quantized data belongs during extraction. Spread spectrum applies the orthogonality of the codebook vector to embed the watermark into the host signal. Quantization is easy to implement with low algorithm complexity, but it is difficult to resist scaling attacks. Spread spectrum has strong robustness against scaling attacks; however, it suffers from the host signal interference.

The feature point-based image watermarking takes some local feature points as the reference points and then uses them to locate some nonoverlapping feature regions, which the watermarks will be embedded into. Tang and Hang [13] utilize the Harris detector to extract feature points and divide the picture into a collection of nonintersect triangles for watermark insertion by Delaunay tessellation. The drawback of this method is that if the feature points retrieved from the original and attacked pictures do not match, the watermark embedding and extraction triangle groups will be different, resulting in the extraction failing. Tang and Hang [14] determine the feature points by using a feature extraction approach called Mexican Hat wavelet scale interaction, and the watermark is embedded in the normalized circular regions centered on these points. Furthermore, several algorithms select the local geometric invariant feature points such as the scale-invariant feature transform (SIFT) [15–19], the Speeded Up Robust Feature (SURF) [20], and the KAZE [21] to locate the feature regions, by using the spatial coordinate and the scale information of these points. Through modifying the pixels in the spatial domain, Lee et al. [16] embed the watermark into the circular patches centered on the chosen SIFT feature points. Zhang et al. [20] present a new watermarking scheme against RST distortion based on SURF and embed the watermark by using the odd-even quantization technique. Liu et al. [21] repeatedly embed the watermarks into the significant bitplanes of the KAZE feature regions, by modifying their histograms. In summary, the general framework of the feature point-based watermarking algorithms follows.

(i) The watermark embedding process

Step 1. Extract the feature points from the image or the frame.

Step 2. Select a particular number of feature points based on certain specified criteria to locate the nonoverlapping feature regions, and their shape is typically square or circular, which is determined as follows:

$$(x - t_1)^2 + (y - t_2)^2 = (ks)^2, \quad (1)$$

where (t_1, t_2) is the spatial coordinate of the selected feature point, s represents the scale information, which is approximately proportional to the scaling factor, and k is a magnification factor to control the radius of the feature regions.

Step 3. Embed the same watermark into these determined feature regions repeatedly, using a specific watermark embedding method. The watermarked image or frame is generated.

(ii) The watermark extraction process

Obtain the feature regions of the watermarked image or frame in the same manner that the embedding process did and then repeatedly extract the watermarks from these regions using the extraction method which corresponds to the embedding method described above. The ownership is proven if the watermark can be identified effectively in at least one region. However, neither (t_1, t_2) nor s can be completely reappeared during extraction, resulting in the feature region desynchronization.

Many watermarking algorithms insert the template in order to recover the geometric attacks, which are mostly manifested in the following structures: Pereira et al. [22, 23] utilize numerous discrete points placed at a specific distance on two straight lines as the template points; Qi et al. [24] use two straight lines as the template; Tokar et al. [25] suggest embedding the square templates in the intermediate frequency transform domain of the images. These methods are resistant to geometric attacks such as scaling and cropping. However, the template and the watermark are both contained in the same image and will interfere with each other. As a consequence, it is difficult for the template-based watermarking algorithms to attain optimal results in terms of imperceptibility and robustness.

3. A Scene Change-Based Frame Allocation Strategy

Because only one image is available as the carrier for the typical template-based image watermarking, the template and the watermark must be embedded into the same image, causing mutual interference. A video, on the other hand, could be seen as a series of images, so it has multiple carriers for embedding. Based on this, we can embed the template and the watermark into different frames, respectively.

In order to embed the template, all of the pixel values of a frame must be changed, which may decrease the imperceptibility. Hence, for embedding the template, it is essential to pick the frames that are insensitive to human eyes, and the scene change frame fulfils this criterion. The scene change frame is the initial frame of each scene in a video. It changes so quickly during playback that the embedded information is difficult to discover, and using it as a reference point to locate the watermarked frame can significantly increase the extraction efficiency [10]. Based on these advantages, a scene change-based frame allocation strategy shown in Figure 1 is proposed: the scene change frames are chosen to embed the template, and the N frames behind each scene change frame are chosen to embed the watermark, where N is an empirical value.

Moreover, if the current frame is a scene change frame, the correlation coefficient between its histograms and those of the preceding frame will not exceed an empirical threshold. So, we extract the scene change frames based on the correlation coefficient. Denote the correlation coefficient as $d(H(x), H(x+1))$, and then, it is calculated as follows:

$$d(H(x), H(x+1)) = \frac{\operatorname{cov}(H(x), H(x+1))}{\sqrt{D(H(x))D(H(x+1))}}, \quad (2)$$

where $H(x)$ is the histogram of the Y component in the x-th frame, $\operatorname{cov}(H(x), H(x+1))$ is the covariance between $H(x)$ and $H(x+1)$, $D(x)$ is the variance of $H(x)$, and $D(x+1)$ is the variance of $H(x+1)$. If $d(H(x), H(x+1))$ does not exceed a threshold denoted as Thre, then the corresponding frame of $H(x+1)$ is regarded as a scene change frame.

4. The Proposed Video Watermarking Scheme

This section focuses on the video watermarking scheme based on the feature points and the template, which includes the embedding and extraction process.

4.1. Embedding Process. The embedding process is divided into two stages: the template embedding and the watermarking embedding, as shown in Figure 2.

4.1.1. Template Embedding. The template for recovering the linear transform exploits the general property of the Fourier Transform. Define an image as $f(x, y)$, where $0 \le x < M$, $0 \le y < N$, and M and N are the width and the height of the image. The DFT is defined as

$$F(u, v) = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} f(x, y) \exp\left(-\frac{j2\pi xu}{M} - \frac{j2\pi yv}{N}\right). \quad (3)$$

And its inverse transform is

$$f(x, y) = \frac{1}{MN} \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} F(u, v) \exp\left(\frac{j2\pi xu}{M} + \frac{j2\pi yv}{N}\right). \quad (4)$$

A linear transform in the spatial domain leads to another corresponding linear transform in the DFT domain. That means, if a linear transform G is suffered in the spatial domain as follows:

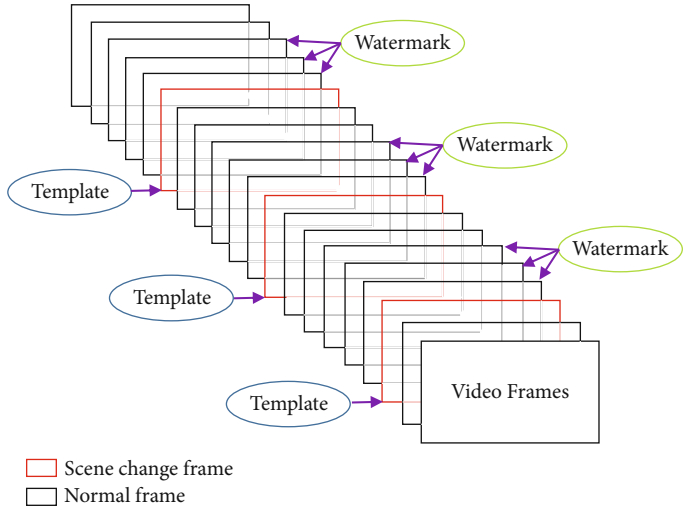


Figure 1: The frame allocation strategy based on scene change.

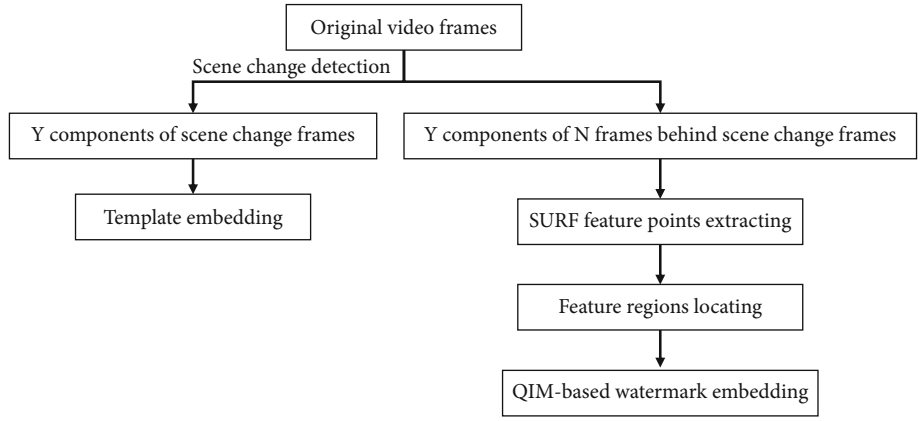


Figure 2: The embedding process.

$$\begin{bmatrix} x \\ y \end{bmatrix} \longrightarrow G \begin{bmatrix} x \\ y \end{bmatrix}. \tag{5}$$

Then, the following transform is carried out in the DFT domain, accordingly.

$$\begin{bmatrix} u \\ v \end{bmatrix} \longrightarrow \left(G^{-1}\right)^{T} \begin{bmatrix} u \\ v \end{bmatrix}. \tag{6}$$

Thus, by detecting the linear transform of the template in the DFT domain, the corresponding transform in the spatial domain can be deduced.

The template embedding process is shown in the following ways.

Step 1. Perform scene change detection on the host video, according to Equation (2); obtain the Y components of the scene change frames.

Step 2. For each obtained Y component, pad it with zeros to a size of 3000×3000 and then apply the Fast Fourier transform (FFT) to it.

Step 3. Choose two radial lines with an angle of 45 degrees to the coordinate axis in the DFT domain (fixing the angle of 45 degrees can make the extraction process easier because we just need to match the abscissa of the template point); pick seven points in the middle frequency band $[f_1, f_2]$ of each line, and the interval between two adjacent points is equal to 11 pixels (refer to Figure 3); it is necessary to select 7 points at symmetrical positions to obtain real coefficients.

Step 4. For each selected point (x_i, y_i), embed the template by following equation:

$$\mathrm{MagT}(x_i, y_i) = \mathrm{LocalMean}(x_i, y_i) + \alpha \mathrm{Std}, \tag{7}$$

where $i = 1, \cdots, 7$, $\mathrm{MagT}(x_i, y_i)$ is the magnitude of the i-th point containing the template, $\mathrm{LocalMean}(x_i, y_i)$ is the

average of the magnitude of 120 pixels adjacent to the i-th point, and Std is the standard deviation of the whole DFT spectrum.

Step 5. Apply the inverse FFT to the Y component with the template, depadding it to the original size.

4.1.2. Watermark Embedding. We use the QRCode with a size of $W \times W$ as the original watermark, due to its high decoding reliability and strong error correction [11]. The detailed steps of the watermark embedding process are presented below.

Step 1. The scene change frames are extracted from the host video, according to Equation (2), and then, the next N frames of each scene change frame are selected for watermark embedding.

Step 2. Extract the SURF feature points from the Y component of each selected frame, and construct the $R \times R$ feature region with each point as the center; eliminate the points whose corresponding feature regions exceeding the frame boundary; if there are overlapping regions, reserve the point with higher intensity.

Step 3. Select the first three points with the highest intensity among the remaining feature points, and the watermark will be repeatedly embedded in the feature regions corresponding to these points.

Step 4. Segment each selected feature region into $W \times W$ blocks; for each block, calculate the DC coefficient in the discrete cosine transform based on Equation (8).

$$\mathrm{DC}_{ij} = \frac{1}{\sqrt{B_w B_h}} \sum_{x=0}^{B_w-1} \sum_{y=0}^{B_h-1} I_{ij}(x, y), \tag{8}$$

where DC_{ij} is the DC coefficient of the block located in the i-th row and the j-th column of the feature region, B_w and B_h are the width and the height of the block, and $I_{ij}(x, y)$ is the block in the i-th row and j-th column of the feature regions.

Step 5. Embed one-bit information of the watermark into DC_{ij}, according to Equation (9).

$$\mathrm{DC}_{ij}^{*} = \begin{cases} (x - 0.5)\delta, \text{if } (x + w_{ij}) \bmod 2 = 1, \\ (x + 0.5)\delta, \text{if } (x + w_{ij}) \bmod 2 = 0, \end{cases} \tag{9}$$

where $x = \mathrm{round}(\mathrm{DC}_{ij}/\delta)$, $\mathrm{round}()$ is the rounding function, DC_{ij}^{*} is the corresponding value after embedding the watermark into DC_{ij}, and δ is the quantization step to adjust imperceptibility and robustness.

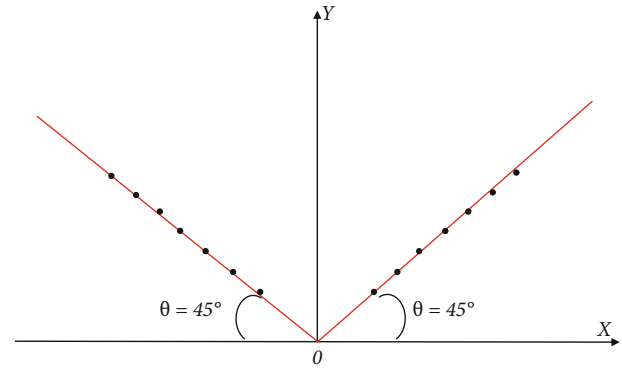


Figure 3: The template embedding.

Step 6. Obtain the watermarked block $I_{ij}^{*}(x, y)$ by the following equation [24]:

$$I_{ij}^{*}(x, y) = \frac{1}{\sqrt{B_w B_h}} \left(\mathrm{DC}_{ij}^{*} - \mathrm{DC}_{ij}\right) + I_{ij}(x, y) \tag{10}$$

When all the blocks in the current frame are embedded with the watermark, continue to do the same operation on the next frame.

4.2. Extraction Process. The template is extracted before the watermark extraction process, in order to recover the watermarked videos which may have been destroyed by geometric attacks, and then, the watermark is extracted from the recovered videos. In summary, the whole extraction process is shown in Figure 4.

4.2.1. Template Extraction. The whole procedures for extracting the template are shown below.

Step 1. Perform scene change detection on the watermarked video, according to Equation (2); obtain the Y components of the scene change frames.

Step 2. For each obtained Y component, pad it with zeros to a size of 3000×3000 and then apply the FFT to it.

Step 3. For the points on the two lines with an angle of 45 degrees between the radial and the coordinate axis of the DFT domain, all the local peak points (px_j, py_j) are extracted by the following formula:

$$\mathrm{MagT}\left(px_j, py_j\right) - \mathrm{LocalMean}\left(px_j, py_j\right) - \beta \mathrm{Std}' \geq 0, \tag{11}$$

where j is the index of the peak points, $\mathrm{MagT}(px_j, py_j)$ is the magnitude value of the point (px_j, py_j), $\mathrm{LocalMean}(px_j, py_j)$ is the average of the magnitude of 120 pixels adjacent to (px_j, py_j), Std' is the standard deviation of the whole DFT spectrum of the selected frame, and β is the detection strength.

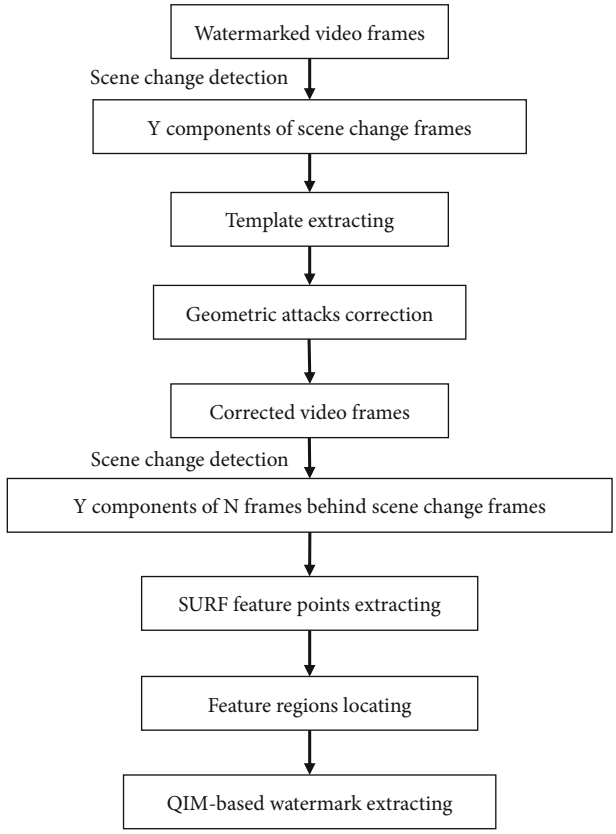


Figure 4: The extraction process.

Step 4. If there are at least 4 points that match the points in the original template line, it is considered that a matching line is found, and the matching rule is

$$px_j - Kx_i \le \text{thresh}, \tag{12}$$

where px_j is the abscissa of the extracted peak point, x_i is the abscissa of the original template point, K is the scaling factor between 0.4 and 1.5, and thresh is the empirical value.

Step 5. Perform statistical analysis on the whole results to get the final matching result.

Step 6. According to the matching result of the template, the geometric attack correction is performed on the attacked video.

4.2.2. Watermark Extraction. The detailed steps of the watermark extraction are proposed below.

Step 1. Perform scene change detection on the watermarked video which is recovered by the template, based on (2), and then select the next N frames of each scene change frame for watermark extraction.

Step 2. Extract the SURF feature points on each selected frame, construct a $R \times R$ feature region with each point as the origin, and remove the points corresponding to the cross-border regions.

Step 3. Select the first ten points with the highest intensity, and the watermark is extracted from the feature regions corresponding to the selected points and their eight neighborhoods, according to the following equation:

$$w'_{ij} = \begin{cases} 1, & \text{if floor}\left(\dfrac{\text{DC}'_{ij}}{\delta}\right) \bmod 2 = 1, \\ 0, & \text{if floor}\left(\dfrac{\text{DC}'_{ij}}{\delta}\right) \bmod 2 = 0, \end{cases} \tag{13}$$

where DC'_{ij} is the DC coefficient of the image block in the i-th row and j-th column of the feature region, floor() is the function of rounding down, w'_{ij} is the watermark information extracted from the image block, locating in the i-th row and j-th column of the feature region, and δ is the quantization step the same as that in the watermark embedding process.

Furthermore, the high decoding reliability of the QRCode is used to determine if the watermark is effectively extracted, which implies that when a QRCode can be successfully decoded by the decoder, the error rate of decoding is near zero. As a result, if the QRCode extracted from at least one feature region can be successfully decoded, the likelihood of treating it as the embedded watermark is close to 100 percent, and the ownership is established [11].

5. Experimental Evaluation

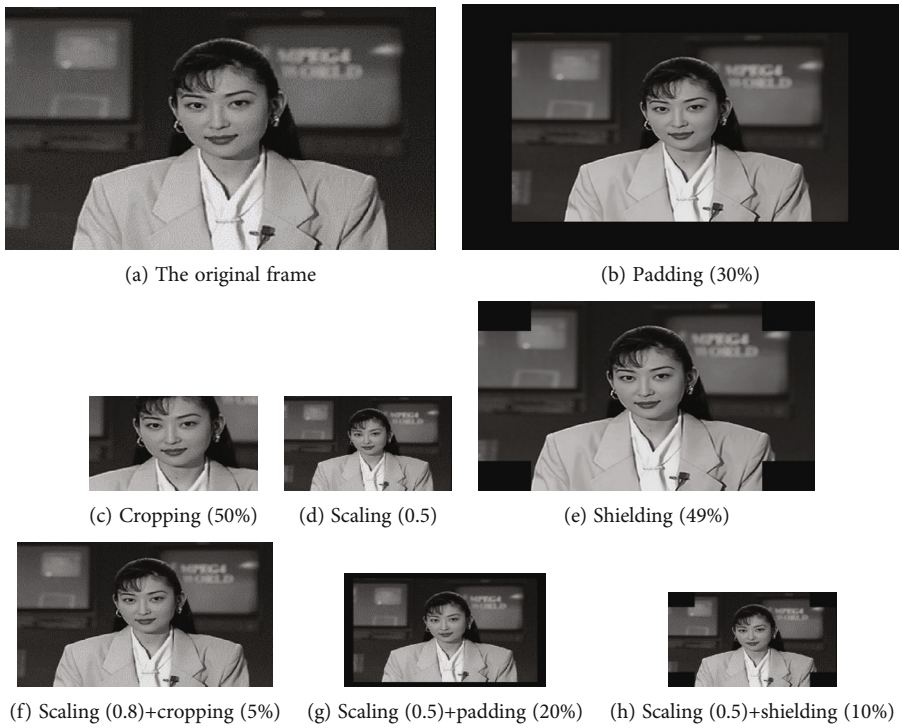
We evaluate the performance of the proposed video watermarking in this section. The experimental setup is introduced in Section 5.1. Section 5.2 verifies the effectiveness of the scene change-based frame allocation strategy. Section 5.3 compares the robustness between our algorithm and the state-of-the-art methods.

5.1. Experimental Setup. Test set. The test set includes 50 1080 P (1920 × 1080) and 50 720 P (1280 × 720) videos. They are all in the mp4 format, with a frame rate between 23.98 and 30 (frames per second) and a duration ranging from 90 to 360 (s).

Environments. The experiments were performed on a PC with 16 GB RAM and 3.4 GHz Intel Core i7 CPU, running on 64-bit Windows 10. The simulation software was Visual Studio 2010 with FFmpeg 2.1 and OpenCV 2.4.9.

Parameters. The number of frames behind every scene change frames to embed the watermark N is set to 10. The threshold of scene change Thre is set to 0.6. The middle frequency band $[f_1, f_2]$ is set to [400, 478]. The template embedding strength α is set to 30. The size of the QRCode to be embedded $W \times W$ is selected as 25×25. The size of the feature region $R \times R$ is set to 200×200. The quantization step δ is set to 60. The match threshold of the template thresh is set to 0.75. The detection strength β is set to 0.05.

Evaluation indexes. We evaluate the imperceptibility and the robustness of the algorithms by using the mean peak signal to noise ratio (MPSNR) and byte error rate (BER), respectively. The larger the MPSNR is, the better the imperceptibility is. On



(a) The original frame (b) Padding (30%)

(c) Cropping (50%) (d) Scaling (0.5) (e) Shielding (49%)

(f) Scaling (0.8)+cropping (5%) (g) Scaling (0.5)+padding (20%) (h) Scaling (0.5)+shielding (10%)

Figure 5: The original frame and its corresponding geometric attacked frames.

Table 1: The imperceptibility comparison results between two different strategies.

Strategies	MPSNR (dB)
Proposed	47.94
TTWAS	41.64

the contrary, the smaller the BER is, the better the robustness is. Furthermore, they are calculated as Equations (14) and (15), separately.

$$\mathrm{MPSNR} = \frac{1}{I}\sum_{i=1}^{I} 10\log_{10}\frac{255^2 MN}{\sum_{m=1}^{M}\sum_{n=1}^{N}\left[F_i(m,n) - F_{iw}(m,n)\right]^2}, \tag{14}$$

where I is the number of the watermarked frames, M and N are the width and the height of the video, and F_i and F_{iw} are the i-th original frame and its corresponding watermarked frame.

$$\mathrm{BER} = \frac{N(\mathrm{errorBytes})}{N(\mathrm{totalBytes})}, \tag{15}$$

where $N(\mathrm{errorBytes})$ is the number of bytes inconsistent between the extracted watermark and the original watermark and $N(\mathrm{totalBytes})$ is the number of bytes of the original watermark.

The types of geometric attacks. Padding, cropping, scaling, shielding, scaling and cropping, scaling and padding, and scaling and shielding are the mainly geometric attacks

Table 2: The comparisons of the robustness against geometric attacks between two different strategies.

Attack types	Parameters	Proposed (%)	TTWAS (%)
Padding	30%	0	-
	40%	0	-
	50%	0	-
Cropping	40%	0	-
	50%	0	-
	60%	0	-
Scaling	0.4	36	-
	0.5	2	-
	1.5	1	-
Shielding	25%	0	-
	36%	0	-
	49%	0	-
Scaling & cropping	0.8 & 5%	2	-
	0.8 & 10%	2	-
	0.8 & 20%	2	-
	0.8 & 30%	1	-
Scaling & padding	0.4 & 2%	79	-
	0.4 & 5%	79	-
	0.4 & 7%	88	-
	0.4 & 10%	89	-
Scaling & shielding	0.5 & 5%	1	-
	0.5 & 10%	5	-
	0.4 & 5%	50	-
	0.4 & 10%	56	-

Table 3: The comparisons of the robustness against geometric attacks between two algorithms.

Attack types	Parameters	Proposed (%)	TFPA (%)
Padding	30%	0	0
	40%	0	0
	50%	0	0
Cropping	40%	0	0
	50%	0	0
	60%	0	0
Scaling	0.4	36	62
	0.5	2	4
	1.5	1	0
Shielding	25%	0	0
	36%	0	0
	49%	0	1
Scaling & cropping	0.8 & 5%	2	9
	0.8 & 10%	2	9
	0.8 & 20%	2	5
	0.8 & 30%	1	8
Scaling & padding	0.4 & 2%	79	86
	0.4 & 5%	79	86
	0.4 & 7%	88	88
	0.4 & 10%	89	93
Scaling & shielding	0.5 & 5%	1	7
	0.5 & 10%	5	8
	0.4 & 5%	50	74
	0.4 & 10%	56	76

used in this paper, and Figure 5 illustrates part of attacked frames of an original frame in [30].

5.2. Verifying the Effectiveness of the Scene Change-Based Frame Allocation Strategy. This subsection verifies the effectiveness of the proposed scene change-based frame allocation strategy by comparing it with the typical template and watermark allocation strategy (TTWAS) [22–25]. For a fair comparison, only the allocation manner of the template and the watermark of the two strategies differs, and the other settings are the same. We evaluate the effectiveness through the imperceptibility and the robustness.

The imperceptibility comparison results are shown as Table 1, and we can find that our strategy can enhance the imperceptibility efficiently. By using the TTWAS, the template and the watermark should be embedded in the same frame at the same time. However, in our proposed strategy, we allocate them to embed into different frames, so that the amount of the embedded data in a frame is reduced, and the imperceptibility becomes better.

Furthermore, the comparisons of the robustness against geometric attacks between the two different strategies are shown in Table 2, where "-" empresses that extracting the watermark fails. When the embedding strength of the two strategies is equal, which is set as before in Section 5.1, TTWAS cannot successfully withstand all of the attacks in the table, while our approach has certain robustness. The mutual interference phenomenon that exists in TTWAS can be effectively eliminated through our strategy, for embedding the template and the watermark in different frames.

5.3. Comparisons of Robustness against Geometric Attacks. This subsection compares the ability of our algorithm and the baseline to recover the embedded watermarks in different geometric attacks. The typical and widely used location method of the feature point-based watermarking [17, 18] is selected as the baseline of the proposed video watermarking algorithm, which is named as TFPA (typical feature point-based algorithm).

The comparison results are shown in Table 3. In most cases, our algorithm outperforms the TFPA, due to introducing the template to assist the location of the feature points. After the video frames subjected to geometric attacks are restored, it is no longer necessary to use the scale information of the feature points when determining the feature regions, which reduces the locating error and improves the accuracy of the watermark extraction. However, our algorithm performs a bit worse against enlarging scaling, because the embedded templates may occasionally be undetected.

6. Conclusions

In this paper, a feature point-based video watermarking algorithm is presented, combining with the template. Because a video is made up of a series of images, it has more hosts for embedding the information. Based on this, a scene change-based frame allocation approach is proposed, which embeds the template and the watermark into different frames. This strategy can enhance the imperceptibility and avoid the mutual interference between the watermark and the template effectively. Furthermore, the insertion of the template reduces the location error of the feature points, improving the robustness against geometric attacks. In most cases, and the experimental results indicate that our algorithm outperforms the state-of-the-art approach. In future study, we will improve the robustness of our algorithm against more geometric attacks so that it can be applied to realistic scenarios.

Data Availability

The videos used to support the findings of this paper are subject to privacy or copyright, so that they cannot be shared.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the National Key R&D Program of China (2020YFB1406900) and the Key R&D Program of Shanxi (201903D421007). It was also the research achievement of the Key Laboratory of Digital Rights Services, which is one of the National Science and Standardization Key Labs for Press and Publication Industry.

References

[1] M. Yan, S. Li, C. A. Chan, Y. Shen, and Y. Yu, "Mobility prediction using a weighted Markov model based on mobile user classification," *Sensors*, vol. 21, no. 5, p. 1740, 2021.

[2] M. Yan, H. Yuan, Z. Li, Q. Lin, and J. Li, "Energy savings of wireless communication networks based on mobile user environmental prediction," *Journal of Environmental Protection and Ecology*, vol. 22, no. 1, pp. 206–217, 2021.

[3] M. Asikuzzaman and M. R. Pickering, "An overview of digital video watermarking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 9, pp. 2131–2153, 2018.

[4] X. Yu, C. Wang, and X. Zhou, "A survey on robust video watermarking algorithms for copyright protection," *Applied Sciences*, vol. 8, no. 10, pp. 1891–1917, 2018.

[5] C. Li, Y. Yang, K. Liu, and L. Tian, "A semi-fragile video watermarking algorithm based on H.264/AVC," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8848553, 2020.

[6] A. Cedillo-Hernandez, M. Cedillo-Hernandez, M. N. Miyatake, and H. P. Meana, "A spatiotemporal saliency-modulated JND profile applied to video watermarking," *Journal of Visual Communication and Image Representation*, vol. 52, pp. 106–117, 2018.

[7] H. Zhao, Q. Dai, J. Ren, W. Wei, Y. Xiao, and C. Li, "Robust information hiding in low-resolution videos with quantization index modulation in DCT-CS domain," *Multimedia Tools and Applications*, vol. 77, no. 14, pp. 18827–18847, 2018.

[8] S. Ponni alias Sathya and S. Ramakrishnan, "Non-redundant frame identification and keyframe selection in DWT-PCA domain for authentication of video," *IET Image Processing*, vol. 14, no. 2, pp. 366–375, 2020.

[9] S. P. Alias Sathya and S. Ramakrishnan, "Fibonacci based key frame selection and scrambling for video watermarking in DWT–SVD domain," *Wireless Personal Communications*, vol. 102, no. 2, pp. 2011–2031, 2018.

[10] X. Li, X. Wang, W. Yang, and X. Wang, "A robust video watermarking scheme to scalable recompression and transcoding," in *2016 6th International Conference on Electronics Information and Emergency Communication (ICEIEC)*, pp. 257–260, Beijing, China, 2016.

[11] Z. Lv, Y. Huang, H. Guan, J. Liu, S. Zhang, and Y. Zheng, "Adaptive video watermarking against scaling attacks based on quantization index modulation," *Electronics*, vol. 10, no. 14, p. 1655, 2021.

[12] K. Matsui and K. Tanaka, "Video-steganography: how to secretly embed a signature in a picture," *IMA Intellectual Property Project Procedure*, vol. 1, pp. 187–206, 1994.

[13] P. Bas, J.-M. Chassery, and B. M. Macq, "Geometrically invariant watermarking using feature points," *IEEE Transactions on Image Processing*, vol. 11, no. 9, pp. 1014–1028, 2002.

[14] C. -W. Tang and H. -M. Hang, "A feature-based robust digital image watermarking scheme," *IEEE Transactions on Signal Processing*, vol. 51, no. 4, pp. 950–959, 2003.

[15] X. Wang, P. Niu, H. Yang, C. Wang, and A. Wang, "A new robust color image watermarking using local quaternion exponent moments," *Information Sciences*, vol. 277, pp. 731–754, 2014.

[16] H. Y. Lee, H. Kim, and H. K. Lee, "Robust image watermarking using local invariant features," *Optical Engineering*, vol. 45, no. 3, article 037002, 2006.

[17] M. Kawamura and K. Uchida, "Sift feature-based watermarking method aimed at achieving IHC ver.5," in *International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, pp. 381–389, Springer, Cham, 2018.

[18] W. -L. Lyu, C. -C. Chang, T. -S. Nguyen, and C. -C. Lin, "Image watermarking scheme based on scale-invariant feature transform," *KSII Transactions on Internet and Information Systems*, vol. 8, no. 10, pp. 3591–3606, 2014.

[19] X. Gao, C. Deng, X. Li, and D. Tao, "Local feature based geometric-resistant image information hiding," *Cognitive Computation*, vol. 2, no. 2, pp. 68–77, 2010.

[20] B. Zhang, J. Wang, J. Wen, and Z. Tong, "A novel digital watermark against RST distortion based on surf," in *2010 IEEE International Conference on Information Theory and Information Security*, pp. 130–133, Beijing, China, 2010.

[21] X. Liu, Y. Wang, J. Du, S. Liao, J. Lou, and B. Zou, "Robust hybrid image watermarking scheme based on KAZE features and IWT-SVD," *Multimedia Tools and Applications*, vol. 78, no. 5, pp. 6355–6384, 2019.

[22] S. Pereira and T. Pun, "Robust template matching for affine resistant image watermarks," *IEEE transactions on image Processing*, vol. 9, no. 6, pp. 1123–1129, 2000.

[23] S. Pereira and T. Pun, "Fast robust template matching for affine resistant image watermarks," in *International Workshop on Information Hiding*, pp. 199–210, Springer, Berlin, Heidelberg, 2000.

[24] X. Qi and J. Qi, "Improved affine resistant watermarking by using robust templates," *2004 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, pp. 405–408, 2004.

[25] T. Tokar and D. Levicky, "Robust watermarking of gray scale images by using synchronization templates," in *2007 17th International Conference Radioelektronika*, Brno, Czech Republic, 2007.

[26] Q. Su and B. Chen, "Robust color image watermarking technique in the spatial domain," *Soft Computing*, vol. 22, no. 1, pp. 91–106, 2018.

[27] C. Wang, X. Li, M. Xu, J. Wang, and W. Wan, "Blind photograph watermarking with robust defocus-based JND model," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8892349, 2020.

[28] Y. Huang, B. Niu, H. Guan, and S. Zhang, "Enhancing image watermarking with adaptive embedding parameter and psnr guarantee," *IEEE Transactions on Multimedia*, vol. 21, no. 10, pp. 2447–2460, 2019.

[29] H. Sadreazami, M. O. Ahmad, and M. Swamy, "Multiplicative watermark decoder in contourlet domain using the normal inverse Gaussian distribution," *IEEE Transactions on Multimedia*, vol. 18, no. 2, pp. 196–207, 2016.

[30] https://media.xiph.org/video/derf/.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 2174190, 11 pages
https://doi.org/10.1155/2021/2174190

Research Article

Dynamic Mathematical Model of Information Spreading on News Platform

Guohui Song[1], **Yongbin Wang**[1] **and Yugang Li**[2]

[1]*State Key Laboratory of Media Convergence and Communication, Communication University of China, Beijing 100024, China*
[2]*Academy of Broadcasting Science, Beijing 100866, China*

Correspondence should be addressed to Yongbin Wang; ybwang@cuc.edu.cn

Received 1 September 2021; Revised 8 October 2021; Accepted 21 October 2021; Published 8 November 2021

Academic Editor: Yun Tie

Copyright © 2021 Guohui 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.

In recent years, the news platform has become the primary source of information for users. However, there are few studies on the news platform, especially for the analysis and modeling of the spreading process of information. This article models the dynamic process of information spreading on the news platform. Firstly, we analyze the dynamic characteristics of user state and information value. Users of news platforms have two states, active and silent states, and users can switch between these two states. The information value determines the probability of user state conversion. We construct the mathematical model for the dynamic features of user state and information value considering these characteristics. Then, with appropriate parameter assumptions, simulation experiments are performed to analyze the regularity of information spreading. The results of the experiment show that the user's reading speed $1/r$ and the conversion probability $p(\alpha, \beta, t)$ are important indicators that affect user state conversion. The lower reading speed and higher conversion probability can improve the transformation of the user state. Furthermore, we present some applications to promote information spreading, such as assessing the effectiveness of information spreading and controlling rumors on news platforms. Finally, we analyzed the effect of its information dissemination by taking Toutiao as an example and confirmed that the visibility and quality of information are important factors that affect information spreading. The experiments and analysis show that the dynamic mathematical model can reflect the information spreading in different situations with different parameters on the news platform.

1. Introduction

At present, the news platform has become the essential way for users to get information. In China, some news platforms represented by Toutiao and Tencent News rely on big data technology and recommendation algorithms to increase user stickiness by meeting users' personalized demands, making them a media platform with strong competitiveness. At the end of 2020, the mobile news platform recorded monthly active users of more than 600 million, and the penetration rate of the entire network is close to 80%. Among them, the top platforms such as Toutiao and Tencent News maintain their dominant positions and reach the level of billions MAU [1] (Figure 1). These platforms play an important role in the spread of information between users and platforms. Therefore, understanding the mechanism of information spreading can facilitate assessing the effectiveness of information spreading and the design of information pushing strategies on the news platform.

In recent years, studying the dynamic process of information spreading has become a challenging task because it can help people understand the mechanism of information spreading [2]. Among them, the most common methods are to observe the flow of information by tracking the amount of information spread on the social network. Most studies focus on rumor control [3–5], traceability of fake news [6–8], avoiding economic crises [9], and improving the effectiveness of information spreading [10, 11]. Due to the enormous value in the political and economic fields and the demand for commercial engineering applications, increasing attention has been paid to information spreading [12, 13]. Meanwhile, the above research focuses mainly on

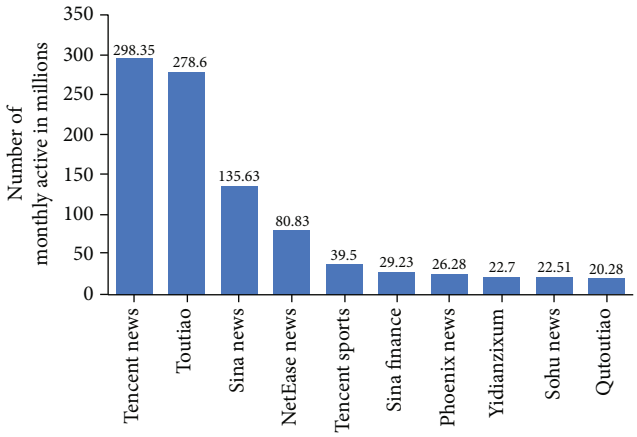


Figure 1: Number of monthly active users (MAU) of the leading news platform in China in December 2020.

social networks. The theory of epidemic spreading [14–17] and the complex network model [18–21] can be applied to information spreading on the social network. In addition, with the rapid development of the 5G mobile Internet, user mobility prediction [22, 23] based on mobile network access also has an important impact on the information spreading on the social network. The above research focuses mainly on social networks, and the main idea is to use differential equations to model dynamic changes in user states. Although the research objects are different, the idea of modeling inspires our work.

Through the investigation of existing researches, the current works on information spreading are mainly applied to social networks, such as Twitter and Facebook. Although some news platforms have become the primary sources of information for users, few studies focus on them, especially for analyzing and modeling the spreading process of information on news platforms.

Unlike social networks, information spreading on news platforms mainly relies on some factors such as information content and timeliness. News platforms such as Toutiao and Tencent News are not information producers themselves. They collect news from the Internet through crawler technology and use personalized recommendation technology to push information. However, most news platforms serve only as a medium for information spreading and have minor social attributes compared to social network platforms such as Weibo or Twitter. From a global perspective, the news platform pushes a large amount of information every day. The push of the information from the platform, information consumption, and the demise of the information is a dynamic process. The dynamic characteristics of information on news platforms include two main aspects: (1) The dynamic characteristics of the user state: the users on news platforms are consumers of information. At a particular moment t, the user's state may be active or silent, which means whether the user is reading the information. The user's state will change from silent to active according to a particular probability $p(\alpha, \beta, t)$ due to the information value. Based on the dynamic characteristics of users, the information spreading on news platforms means that users turn from the silent state to the active state. (2) The dynamic characteristics of information: the value of information on news platforms changes dynamically, and changes in information value are related to some factors such as the quantity value of information, visibility value of information, and the time t. Because the information pushed by news platforms is diverse, it should be noted that we are concerned with the value of information on the same topic. The value of information determines the possibility of the user's transformation from a silent state to an active state. The greater the value of information, the easier it is for silent users to convert and the more helpful it is to spread information. Therefore, the mathematical model of the user state and information value allows us to study further information spreading on news platforms.

Based on the dynamic characteristics of information spreading on news platforms, we can construct a mathematical model to describe news platforms' dynamic information spreading process. This paper tries to answer the following two questions: (1) On the news platform, how does the user's state change dynamically? What guidance do dynamic changes of user state have on news platform information push? (2) How does the value of information on news platforms change? How do we quantitatively represent the changing process of information spreading?

The remainder of this paper is organized as follows. We introduce the dynamic characteristics of information spreading on news platforms in Section 2. In Section 3, we construct the dynamic mathematical model based on the dynamic characteristics of user state and information value. In Section 4, the numerical simulation method and parameter settings are introduced. In Section 5, we introduce some applications based on the dynamic mathematical model. Meanwhile, we take Toutiao as a research object to verify the influence of model parameters on the effect of information spreading. In Section 6, conclusions are presented.

2. Dynamic Characteristics during Information Spreading on News Platform

In this section, we first introduce the dynamic characteristic of the user state. Then, the dynamic characteristics of the information are presented. Analyzing these characteristics can help understand the mechanism of information spreading on the news platform and provide a basis for constructing a dynamic mathematical model.

2.1. Dynamic Characteristics of the User State. In the process of information spreading on the news platform, the subjects are users whose state triggers the spreading of information. Dynamic is a typical feature of the user on news platforms, and the user state determines whether the information is effectively spread. At a particular moment t, some users may be in the active state. At the same time, the other users may be silent. The state of the users is transformed according to a certain probability. Some factors determine the probability, and the following factors are closely linked.

(1) Quantity value of information α indicates the attractiveness of information to silent users. The more the quantity value, meaning people are more willing to read, the higher the conversion probability

(2) Visibility value of information β. The visibility of information depends mainly on the push method of the news platform, such as personalized recommendations. The higher the visibility of the information, the higher the conversion probability

(3) Time value of information appearing t indicates the freshness of information. The more it means that the information is outdated and less attracted, the lower the conversion probability. The conversion probability can be denoted by $p(\alpha, \beta, t)$ considering the dynamic characteristics of the user state. The quantity value α and the visibility value β are determined only by the information itself. Thus, α and β are ascertained once the information emerges. We define α and β as the constant that will not change in the process of information spreading. Thus, in the function of $p(\alpha, \beta, t)$, the parameter t is only a variable. Therefore, we use $p(t)$ instead of $p(\alpha, \beta, t)$ in the following part. Since we consider the overall dynamic characteristics of news platforms, the value of $p(t)$ describes the average probability of user state conversion

Based on the above analysis, we can make the following definition.

(1) At any moment t, the probability that the users transfer from the silent state to the active state was denoted as $p(t) \in (0, 1]$

(2) Suppose that the average speed for the users to read information is $1/r$. After reading a piece of information, the users will change from the active state to the silent state at a speed of $1/r$. r is the average time for users to read information. Through the investigation of current research, some studies indicate that the user's reading speed is related to the length of information content, the user satisfaction, the user's historical average reading speed, and the reading dwell time [24, 25]. Some scholars have also studied the effects of mobile device display (such as fonts and line spacing) on reading speed [26, 27]. Therefore, the factors that affect reading speed are complex. In this paper, we set $1/r$ as a constant, representing the average reading speed of all users

Based on the above assumptions, we can describe the dynamic characteristics of the user state as follows: at any moment t, users in the silent state can convert into the active state with a probability of $p(t)$. Meanwhile, users in the active state will turn into the silent state at a speed of $1/r$. It will implement effective information spreading once silent users become active users.

2.2. Dynamic Characteristics of Information Value. The information value on news platforms also has dynamic characteristics. The dynamic characteristic of information value refers to the fact that the attention of information content will change over time. Users' attention to a particular topic is a decisive factor in determining the effectiveness of information dissemination. From a global perspective, at any moment t, we assume that the average value of information on a particular topic is determined by the push speed of information and the total amount of information about the topic. The following factors are closely linked.

(1) The value L indicates the life cycle of the information

(2) At any moment t, for information about a particular topic, the total amount of information transmitted by the news platform is defined as $w(t)$. Because of the limitation of the information push capability of the news platform, we assume that the maximum amount of information is R in the life cycle of the information. So, the value of $w(t)$ is between 0 and R. It means that at the initial moment $t = 0$, the news platform does not push information. At the moment $t = L$, the maximum amount of information transmitted by the news platform is R

(3) At any moment t, the information push speed of the news platform is expressed as $w'(t)$

(4) In the information life cycle L, the more information-pushed before t moment, the easier it is for information-pushed after t moment to lose attention for a particular topic. At any moment t, the mean value of the information value of the news platform is expressed as $v(w', w)$. The faster the topic information is updated, the easier it is for the users to pay attention to a particular topic. However, the more information on this topic, the easier it is to lose interest in the users. For simplicity, we assume that $v(w', w) = w'(t)/w(t)$, it means that the average information value is directly with $w'(t)$ and inversely with $w(t)$

3. Dynamic Mathematical Model of Information Spreading

From the above analysis of the dynamic characteristics of the news platform, we can consider the effect of information spreading on news platforms from different perspectives. Based on the dynamic characteristics of the user state, we can analyze the number of active users at a particular moment t. Based on the dynamic characteristics of information value, we can analyze the changes of information value of a particular topic during the information life cycle.

3.1. Modeling for Dynamic Characteristics of User State. In the process of information spreading, users of news platforms have two states: the active state and the silent state. At any moment t, users who are in the silent state do not get information from the news platform, and they can convert to the active state with a probability of $p(t)$. Meanwhile, the users in the active state will turn into the silent state at a

speed of $1/r$. It will implement effective information spreading once the silent users become active users. The conversion process of the user state is shown in Figure 2.

Based on the above rules, we can make the following assumptions and definitions for the model parameters.

(1) The total number of users on the news platform is defined as n, and the information is sufficient

(2) At moment t, the active and silent users on the news platform are $g(t)$ and $f(t)$, respectively. The $g(t)$ and $f(t)$ meet the following equation

$$f(t) + g(t) = n. \tag{1}$$

(3) Every piece of information on the news platform has the same life cycle, which is defined as L

(4) At the moment t, the rate of change of the user from the silent state to the active state can be expressed as $f(t) \times p(t)$. The $p(t)$ is the probability that users transition from the silent state to the active state

The dynamic model can be constructed according to the user state's dynamic characteristics and the information value's dynamic characteristics. Suppose that the number of users in the active state at the moment t is $g(t)$ and the number of users in the silent state is $f(t)$, the keynote of the dynamic characteristics of users can be presented in the following formula:

$$g(t) + f(t) \xrightarrow{p(t),1/r} g(t+\Delta t) + f(t+\Delta t). \tag{2}$$

The above formula expresses the dynamic change process of the user state when the total number of users is constant n. This process is mainly affected by conversion probability $p(t)$ and reading speed $1/r$. At the moment $t + \Delta t$, the dynamic change of the number of users in the active state can be expressed by the following formula:

$$g(t+\Delta t) = g(t) + f(t)p(t)\Delta t - \frac{g(t)}{r}\Delta t. \tag{3}$$

From the moment t to $t + \Delta t$, the change process of the number of users in the active state can be described as follows. (1) The silent state users turn into the active state with probability $p(t)$. (2) The users in the active state turn into the silent state after reading some information at a speed of $1/r$. Therefore, the rate of change of active users can be calculated by the following formula:

$$\begin{aligned}\frac{dg(t)}{dt} &= \lim_{\Delta t \longrightarrow 0} \frac{(g(t) + f(t)p(t)\Delta t - (g(t)/r)\Delta t)}{\Delta t} \\ &= f(t)p(t) - \frac{g(t)}{r} = (n - g(t))p(t) - \frac{g(t)}{r}.\end{aligned} \tag{4}$$

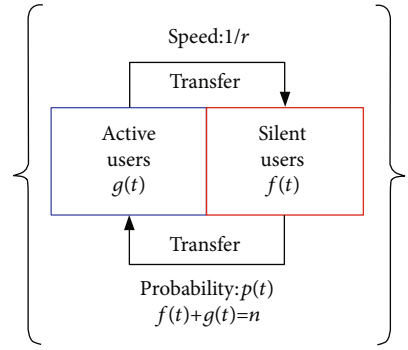


Figure 2: The dynamic conversion process of user state on news platform.

Through the above transformation, the change rate of the active users can be obtained by solving the following first-order linear differential equation:

$$\frac{dg(t)}{dt} = -\left(p(t) + \frac{1}{r}\right)g(t) + np(t). \tag{5}$$

Solving the first-order linear differential equation generally uses the method of constant variation, and this method can obtain the solution of the first-order linear differential equation. First, we can solve the corresponding linear equation, and the linear equation is as follows:

$$\frac{dg(t)}{g(t)} = -\left(p(t) + \frac{1}{r}\right)dt. \tag{6}$$

Then, we can transform the above formula:

$$\ln g(t) = -\int\left(p(t) + \frac{1}{r}\right)dt. \tag{7}$$

Finally, we can get the general solution of this homogeneous equation; the C is constant and it is determined by the initial conditions of the function. Therefore, the number of users in the active state $g(t)$ at the moment t can be represented as

$$g(t) = Ce^{-\int(p(t)+1/r)dt}. \tag{8}$$

Let $C = c(t)$; the above general solution of $g(t)$ can be expressed as

$$g(t) = c(t)e^{-\int(p(t)+1/r)dt}. \tag{9}$$

We further process the above formula:

$$\frac{dg(t)}{dt} = \frac{dc(t)}{dt}e^{-\int(p(t)+1/r)dt} - c(t)\left(p(t)+\frac{1}{r}\right)e^{-\int(p(t)+1/r)dt}. \tag{10}$$

After inserting equations (5) and (9) into equation (10), we can obtain

$$\frac{dc(t)}{dt}e^{-\int(p(t)+1/r)dt} - c(t)\left(p(t)+\frac{1}{r}\right)e^{-\int(p(t)+1/r)dt} = -\left(p(t)+\frac{1}{r}\right)c(t)e^{-\int(p(t)+1/r)dt} + np(t). \tag{11}$$

We further deduce the above equation (11):

$$\frac{dc(t)}{dt} = np(t)e^{\int(p(t)+1/r)dt}dt, \tag{12}$$

$$c(t) = \int np(t)e^{\int(p(t)+1/r)dt}dt + \bar{c}. \tag{13}$$

After inserting equation (13) into equation (9), the general solution of $g(t)$ can be described as the following formula:

$$g(t) = e^{-\int(p(t)+1/r)dt}\left(\int np(t)e^{\int(p(t)+1/r)dt} + \bar{c}\right). \tag{14}$$

The value of $\bar{c}$ is determined by the initial conditions of the model. Therefore, according to equations (1) and (14), the number of silent users $f(t)$ at the moment t can be expressed as

$$f(t) = n - e^{-\int(p(t)+1/r)dt}\left(\int np(t)e^{\int(p(t)+1/r)dt} + \bar{c}\right). \tag{15}$$

Suppose that at the moment $t = 0$, the number of active users $g(t) = 0$. In other words, at the initial moment of the model, the information on the news platform has not been disseminated. It is worth noting that at the moment $t = 0$, $p(t) \neq 0$, because at the initial moment of the model, there is already information on the news platform for users to read. Through the above analysis, we can obtain the value of $\bar{c}$.

$$e^{-\int(p(t)+1/r)dt}\left(\int np(t)e^{\int(p(t)+1/r)dt} + \bar{c}\right) = 0, \tag{16}$$

$$\bar{c} = -\int np(t)e^{\int(p(t)+1/r)dt}. \tag{17}$$

Through the dynamic characteristics of the user state, we can construct the dynamic mathematical model to describe the process of information spreading, and at the same time, we can get a general representation of the number of active users $g(t)$ and silent users $f(t)$. With a certain number of news platform users, we can find that reading speed $1/r$ and conversion probability $p(t)$ are factors that affect $g(t)$. The conversion probability $p(t)$ is determined by information value. The number of active users at the moment t can be used to measure the effect of information spreading on the news platform.

3.2. Modeling for Dynamic Characteristics of Information Value. To further study information spreading based on the dynamic characteristics of information value, the following dynamic function is assumed:

$$v(w', w) = \frac{w'(t)}{w(t)}. \tag{18}$$

The function can reflect a characteristic of information value dynamics with time. At the moment t, the mean value of the information about a particular topic on the news platform is expressed as $v(w', w)$. Therefore, the total value of information from the initial time to t moment can be expressed as

$$\int_0^t \mathrm{v}(\mathrm{w}', \mathrm{w})\mathrm{w}'(\mathrm{t})dt. \tag{19}$$

Equation (19) means that the total information value of a particular topic is determined by the speed of information push speed $w'(t)$ and the total amount of information pushed by the news platform $w(t)$. One of the purposes of news platforms to push information is to maximize the value of information during the information life cycle. Then, the total value of information of a particular topic can be calculated by the following variational equation:

$$\begin{cases} \max \int_0^L \mathrm{v}(\mathrm{w}', \mathrm{w})\mathrm{w}'(\mathrm{t})dt, \\ w(0) = 0, w(L) = R. \end{cases} \tag{20}$$

The value of $w(t)$ is between 0 and R. The L represents the life cycle of information. For a convenient expression, we rewrite $F(w', w) = w'(t)^2/w(t)$. According to the variational method of extreme functional value and the famous Eulerian equation, the above variational equation can be transformed to solve the following problems:

$$F_w - w'F_{w'w} - w''F_{w'w} = 0. \tag{21}$$

Solving the above Euler equations, the following solution process can be obtained:

$$\frac{d(F - w'F_w)}{dt} = 0 \text{ or } F - w'F_w = c, \tag{22}$$

$$w'(t)^2 = cw(t), \tag{23}$$

$$\frac{dw}{\sqrt{w(t)}} = cdt, \tag{24}$$

$$w(t) = (c_1 t + c_2)^2. \tag{25}$$

According to the initial condition $w(0) = 0, w(L) = R$, the value of c_1 and c_2 can be obtained as follows:

$$c_2 = 0, c_1 = \frac{\sqrt{R}}{L}. \tag{26}$$

Therefore, combing equations (25) and (26), we can obtain

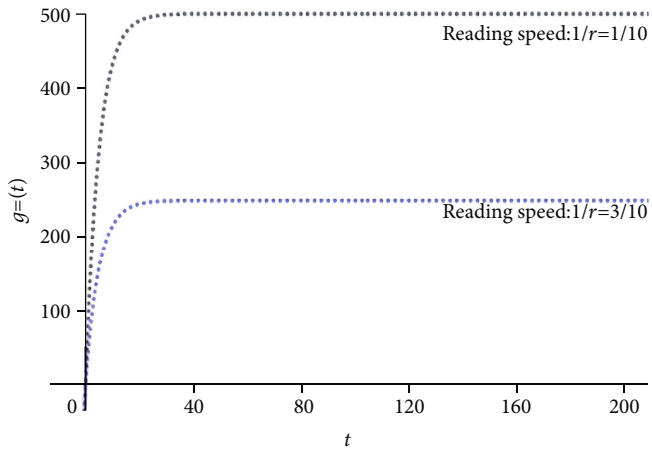


Figure 3: Simulation results of active users when reading speed $1/r = 1/10$ or $3/10$ and $p(t) \equiv 1/10$.

$$w(t) = \frac{Rt^2}{L^2}. \tag{27}$$

After inserting $w(t)$ into equations (18), we thus obtain the information value at the moment t:

$$v = \frac{w'(t)}{w(t)} = \frac{2}{t}. \tag{28}$$

From the above derivation, the value of information on news platforms is closely related to time. Through equation (28), we can quantitatively express the relationship between time and information value. The value of information determines the conversion probability $p(t)$. In this section, we only consider the impact of time on the value of information. In the experiment, we will consider the impact of information quality and information visibility on the value of information.

4. Parameter Assumption and Simulation Experiment

To further study information spreading based on the dynamic model, we can make assumptions about the model parameters. From the definition above, at the moment $t = 0$, the number of active users $g(t) = 0$, $p(t) \neq 0$. If we set the values of $p(0)$ and $1/r$, according to equation (17), we can get the value of $\bar{c}$. Assuming $p(0) = 1/10$, $n = 1000$, $1/r = 1/10$, or $1/r = 3/10$, we can get $\bar{c} = -500$ or $\bar{c} = -250$. This section will discuss the impact of the three manifestations of conversion probability on information spreading.

(1) The dynamic conversion probability $p(t)$ is set to a constant. We define $p(t)$ as the initial value, that is, $p(t) \equiv 1/10$. The $p(t)$ is constant which means that the value of information remains stable on the news platform, resulting in the dynamic conversion probability remaining unchanged. Although this case is only a hypothetical state, it can have a particular guiding significance for the platform's information push. Then, at the moment t, the active users $g(t)$ can be calculated by equation (14):

$$g(t) = e^{-c_1 - t/5}\left(500e^{c_1 + t/5} + c_1 - 500\right). \tag{29}$$

We set c_1=0. We will still set such parameters to effectively represent the solution of the indefinite integral in the following part. Then, equation (29) can be simplified as the following formula:

$$g(t) = 500e^{-t/5}\left(e^{t/5} - 1\right). \tag{30}$$

At the same time, to analyze the influence of user reading speed on information spreading, we set $1/r = 3/10$. Based on the parameters above, we can obtain the number of active users at the moment t on the news platform as shown in Figure 3.

As mentioned above, $p(t) \equiv 1/10$ is assumed, which means the value of information is unchanged during information spreading. In this case, the dynamic characteristic of users is only considered, and simulation experiments are carried out without considering the characteristic of information value. It can be seen that the active users reach a steady state due to the dynamic characteristic of users at the moment t. The final value of $g(t)$ will be different according to the initial conditions. The lower the average reading speed, the faster the users will convert to an active state.

(2) The value of the conversion probability $p(t)$ is related to the dynamic characteristics of information value. Based on the above definition, we can define information value without considering other factors affecting information value by referring to equation (28). The probability of users changing from the silent state to the active state is related to time. The higher the value of the information, the easier it is for the silent users to become active users. Regardless of other factors that affect the information value, the dynamic conversion probability $p(t)$ should be

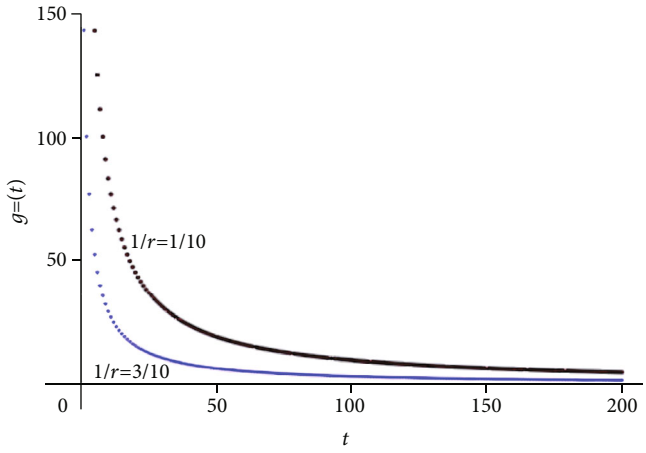


FIGURE 4: Simulation results of active users when $1/r = 1/10$ or $3/10$ and $p(t) = c(2/(t+1))$, $c = 1/20$.

proportional to v and meet the condition $0 < p(t) \leq 1$, so the $p(t)$ can be defined as the following formula:

$$p(t) = c\frac{2}{t+1}. \tag{31}$$

The c is constant. If $c = 1/20$, then $p(0) = 1/10$.

Based on equations (14) and (31), assuming $n = 1000$ and $1/r = 1/10$, the $g(t)$ can be calculated. At the same time, to analyze the influence of reading speed on information spreading, we also set $1/r = 3/30$. Based on the parameters above, we can obtain the number of active users on the news platform, as shown in Figure 4.

Because of the dynamic characteristics of information value, the conversion possibility will decline continuously. The variable conversion possibility is the fundamental principle of information spreading. In this case, we also consider the dynamic characteristic of the users and the dynamic characteristic of information value. The information value is due to time. As shown in Figure 4, it can be seen that the number of active users gradually decreases to a certain extent and reaches a stable value. Information value will be reduced by time because of the dynamic characteristics. It is difficult for active users to reach a high value. The lower the average reading speed, the faster the users will convert into an active state.

(3) The quantity value of information α and the visibility value of information β are essential factors that influence the conversion possibility. When the quantity value of information is higher, users are more willing to read more information to increase the conversion possibility. Meanwhile, the higher the visibility of information value, the easier it is for users to get information. Furthermore, considering the visibility and quantity value of information, we can define the conversion probability as the following equation

$$p(\alpha, \beta, t) = p(\alpha, \beta) + p(t). \tag{32}$$

The exact value of $p(\alpha, \beta)$ will lie on the specific factors of information. Since the values of α and β are determined from the beginning of the information, to simplify the problem, we can consider them together. Without generality, the range of $p(\alpha, \beta)$ is shown as follows:

$$p(\alpha, \beta) = \text{rand}\,(0, 1 - p(t)). \tag{33}$$

It means that it is random in its value range. The random function (33) satisfies uniform distribution. As a result, considering the visibility and quantity value of information, the conversion probability can be calculated by the following formula:

$$p(t) = \text{rand}\left(0, 1 - c\frac{2}{t+1}\right) + c\frac{2}{t+1}. \tag{34}$$

Taking the same parameters as mentioned above, combing equations (14) and (34), and simulation experiments of this case repeated 100 times, the active users $g(t)$ can be obtained as shown in Figure 5. It can be seen that the final value of $g(t)$ is very random due to the quantity value and the visibility value of information. As in the above experiment, the lower the reading speed, the easier it is to promote the conversion of the user state.

Although the value of information decays over time, due to the information value, especially considering the visibility and quantity of information, users can more easily obtain information, which helps users transform from the silent state to the active state. For example, in the experiment shown in Figure 4, if only the time utility of the value of information is considered, the number of active users will gradually decrease and ultimately remain at a low level, less than 2% of the total number of users. In the experiment shown in Figure 5, if the visibility and quantity value of information is considered, the number of active users is more than 20% of the total users most of the time. Sometimes, it can activate almost 80% of users. Meanwhile, we can find that the lower reading speed is conducive to the conversion of the user state. In order to better illustrate the problem, we randomly observe 10 time points, and the results are shown in Table 1.

5. Analysis of Information Spreading

The effect of information spreading depends on how the user transitions from the silent state to the active state. The conversion ability is determined by the conversion probability $p(t)$ and the reading speed $1/r$.

First, only considering the dynamic characteristic of the user state, the conversion probability $p(t)$ is defined as a constant. In this case, the value of $g(t)$ is closely linked with the reading speed $1/r$.

Considering the characteristic of information value, the conversion probability $p(t)$ is related to the time, and the probability continues to decrease over time. So, the value of $g(t)$ will stabilize at a lower level. The conversion

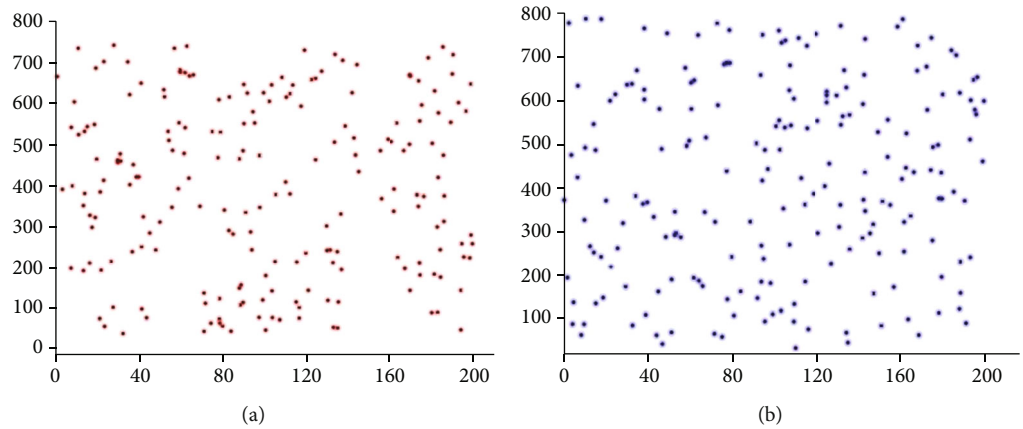


Figure 5: Simulation results of active users when considering the quantity and visibility value of information in the dynamic characteristics of information value: (a) $1/r = 3/10$; (b) $1/r = 1/10$.

Table 1: The results of information spreading with different time t and reading speed 1/r by considering the quantity value and the visibility value of information.

Reading speed: 1/r	Active users: $g(t)$									
	$g(20)$	$g(40)$	$g(60)$	$g(80)$	$g(100)$	$g(120)$	$g(140)$	$g(160)$	$g(180)$	$g(200)$
1/10	210	375	198	245	105	389	565	423	431	603
3/10	85	98	391	83	87	228	543	525	96	685

probability should be enough so that the silent users can effectively convert to active users.

In addition to time, the conversion probability is also determined by the quantity value of information α and the visibility value of information β. Therefore, the conversion probability has certain degrees of randomness because the values of α and β are uncertain on the news platform. Meanwhile, the number of active users is random.

In different cases with different $p(t)$, the results of information spreading can be shown in Table 2. The simulation experiments show that the active users are different due to different cases. When $p(t)$ is constant, the conversion probability will not change. Under this condition, the number of active users will reach a stable value due to the initial parameters set. When considering the relationship between the value of information and time, the conversion probability will generally decrease, so the final number of active users will reach a minimal level. Considering the value of quantity and visibility of information, the probability of conversion can change randomly, making silent users effective at some time.

With the adjustment of the reading speed $1/r$, it can be seen that the reading speed has a particular impact on the user state conversion. The lower the reading speed, the more conducive to the user state conversion.

6. Application and Case Study

6.1. Application. The characteristics of user state and information value can reflect the dynamic process of information spreading on the news platform as a whole. So, we can understand the dynamic transform process of the user's state well and predict the information spreading effect according to the model mentioned above. Meanwhile, we can apply some measures to promote the information spreading.

(1) *Assess the effectiveness of information spreading*: on news platforms, the number of active users and the total amount of information read are critical indicators to evaluate the effect of information spreading. The total amount of information read can be described as the following formula

$$\text{total} = \int_0^L f(t)p(t)dt. \tag{35}$$

According to equations (15) and (35), the total reading users of the information on the news platform can be obtained as follows:

$$\text{total} = \int_0^L \left(n - e^{-\int (p(t)+1/r)dt}\left(\int np(t)e^{\int (p(t)+1/r)dt} + \bar{c}\right)\right)p(t)dt. \tag{36}$$

The L is the life cycle of information. The total represents the number of silent users converted into active users during the news life cycle, and it can dynamically describe the process of information spreading on the news platform. Therefore, we define total as the effect of information spreading on news platforms. During the operation of the news platform, one of the purposes of the news platform is

Table 2: The results of information spreading with different conversion probability $p(t)$ and reading speed 1/r, $c = 1/20$, and $n = 1000$.

Conversion probability: $p(t)$	Reading speed: 1/r	Active users: $g(t)$		
		$g(20)$	$g(100)$	$g(200)$
1/10	1/10	491	500	500
	3/10	245	250	250
$\frac{2c}{(t+1)}$	1/10	72	16	6
	3/10	28	7	2
$\text{rand}\left(0, 1-\frac{2c}{t+1}\right)+\frac{2c}{t+1}$	1/10	210	105	603
	3/10	85	87	685

to increase the reading volume of information. For this purpose, the goal of optimization is the value of total.

(2) *Promote information spreading*: to make silent users turn into active users as quickly as possible is an essential task on the news platform. The process of user state conversion is also the process of information spreading. The experiments and theoretical analysis show that the conversion probability $p(t)$ and reading speed $1/r$ are essential factors in information spreading. As shown in Figures 3–5, different information spreading effects can be achieved by formulating different probability distributions. For example, suppose that the news platform needs a relatively stable information spreading effect. In that case, as shown in Figure 3, the quantity and visibility value of the information must be dynamically adjusted so that the conversion probability maintains a relatively stable value. If the news platform needs to improve the information spreading at specific time points, the visibility and quantity of information should be adjusted. Meanwhile, information spreading can be restrained by reducing the value of information in some areas, such as rumor control. The news platform can take effective measures by revealing malicious rumor information and reducing the visibility of this type of information. The value of the incorrect information or rumor will be decreased in a straight line so that the value of $p(\alpha, \beta)$ is 0. Then, the value of information will decrease progressively so that the dissemination of rumor information will end naturally, as shown in Figure 4

6.2. A Case Study of Toutiao. Toutiao is a popular mobile news platform in China. In order to verify the influence of model parameters on the effect of information spreading, we take Toutiao as the object for further research. At present, Toutiao promotes the dissemination of mainstream value information by placing 2-3 pieces of mainstream value information on the top of the homepage. We got the two pieces of information that Toutiao put on top and analyzed the effect of its information spreading. We use the number of comments as an indicator to evaluate the user activation state. We take a crawl every 15 minutes to obtain a total of 1624 valid data and draw a scatter chart of the data in time

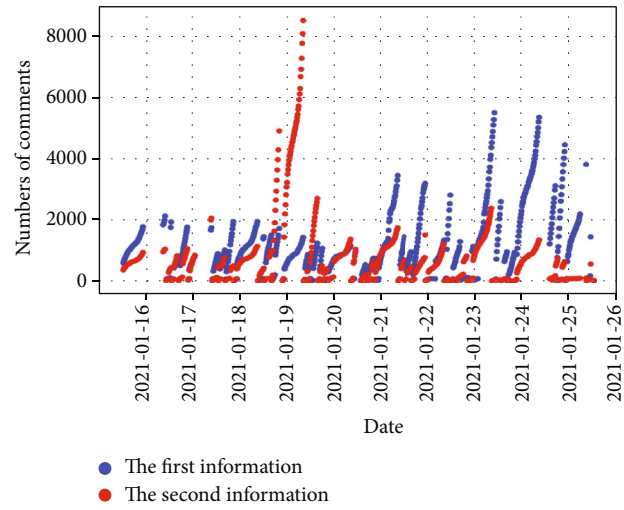


Figure 6: The information spreading effect of Toutiao.

series (Figure 6). Assuming that the average reading speed of users is constant, the effects of model parameters are discussed as follows.

(1) The visibility value of information β. As shown in Figure 6, we can find that the dissemination effect of the first piece of information is better in most cases because the first information is for all users, and the second message is for different users. Therefore, the first piece of information has a higher visibility value. This phenomenon shows that the high visibility value promotes the information spreading on the news platform

(2) The quantity value of information α. Although the first piece of information has a higher visibility value, the second piece of information sometimes spreads better. The main reason is that sometimes, users are more concerned with the second piece of information. Therefore, the quality of information is an important factor affecting information dissemination

(3) As shown in Figure 6, the influence of time on information dissemination is not obvious, mainly because the two pieces of information have high visibility value. Therefore, the news platform can enhance the visibility of specific topic information to maintain its dissemination effect

7. Conclusions

Based on the analysis of dynamic characteristics of the news platform, the paper models the dynamic characteristics of the user state and the value of information. Simulation experiments have been done to analyze the impact of dynamic characteristics on information spreading. Additionally, some applications are proposed and discussed. The main works of the paper are following:

(1) The dynamic characteristics of information spreading on news platforms have been introduced. It is proposed that the dynamic of the user state and information value are involved in the process of information spreading

(2) We model the dynamic characteristics of the user state and information value and provide a detailed calculation process

(3) The related parameters of the dynamic model are assumed, and simulation experiments have been achieved under different parameters and conditions. The experimental results are analyzed in detail

(4) Based on the dynamic characteristics and the dynamic model, some applications are presented. Meanwhile, we analyzed the effect of its information dissemination by taking Toutiao as an example and confirmed that the visibility and quality of information are important factors that affect information spreading

Although this paper uses differential equations to construct a dynamic model of information spreading on news platforms, there are still some limitations. Firstly, we make some assumptions about some critical parameters of the model, such as the average reading speed $1/r$, the value of information quantity α, and information visibility β. How to determine these parameters is for further study. Secondly, a large amount of data is needed to verify and optimize the model, which is the goal of our future work.

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 no conflict of interest.

Acknowledgments

This work was supported by the National Key R&D Program of China (No. 2019YFB1406201).

References

[1] "Number of monthly active users (MAU) of the leading news apps in China in December 2020," 2020, https://www.Statista.Com/Statistics/910787/China-Monthly-Active-Users-on-Leading-News-Apps.

[2] E. Mussumeci and F. C. Coelho, "Reconstructing news spread networks and studying its dynamics," *Social Network Analysis and Mining*, vol. 8, no. 1, pp. 1–8, 2018.

[3] Y. Zhang, Y. Su, W. Li, and H. Liu, "Rumor and authoritative information propagation model considering super spreading in complex social networks," *Physica A: Statistical Mechanics and its Applications*, vol. 506, pp. 395–411, 2018.

[4] Y. Xiao, D. Chen, S. Wei, Q. Li, H. Wang, and M. Xu, "Rumor propagation dynamic model based on evolutionary game and anti-rumor," *Nonlinear Dynamics*, vol. 95, no. 1, pp. 523–539, 2019.

[5] M. Jiang, Q. Gao, and J. Zhuang, "Reciprocal spreading and debunking processes of online misinformation: a new rumor spreading–debunking model with a case study," *Physica A: Statistical Mechanics and its Applications*, vol. 565, article 125572, 2021.

[6] K. Shu, D. Mahudeswaran, S. Wang, D. Lee, and H. Liu, "Fakenewsnet: a data repository with news content, social context, and spatiotemporal information for studying fake news on social media," *Big Data*, vol. 8, no. 3, pp. 171–188, 2020.

[7] O. N. Berduygina, T. N. Vladimirova, and E. V. Chernyaeva, "Trends in the spread of fake news in mass media," *Media Watch*, vol. 10, no. 1, pp. 122–132, 2019.

[8] D. Yang, T. W. S. Chow, L. Zhong, Z. Tian, and Q. Zhang, "True and fake information spreading over the Facebook," *Physica A: Statistical Mechanics and its Applications*, vol. 505, pp. 984–994, 2018.

[9] S. Jiang and H. Fan, "Credit risk contagion coupling with sentiment contagion," *Physica A: Statistical Mechanics and its Applications*, vol. 512, pp. 186–202, 2018.

[10] J. Li and C. Bin, "Modeling for information spreading basing on dynamics," *Concurrency and Computation: Practice and Experience*, vol. 31, no. 9, 2019.

[11] L. Gao, W. Wang, L. Pan, M. Tang, and H. F. Zhang, "Effective information spreading based on local information in correlated networks," *Scientific Reports*, vol. 6, no. 1, pp. 1–10, 2016.

[12] X. Liu, D. He, and C. Liu, "Information diffusion nonlinear dynamics modeling and evolution analysis in online social network based on emergency events," *IEEE Transactions on Computational Social Systems*, vol. 6, no. 1, pp. 8–19, 2019.

[13] Z. K. Zhang, C. Liu, X. X. Zhan, X. Lu, C. X. Zhang, and Y. C. Zhang, "Dynamics of information diffusion and its applications on complex networks," *Physics Reports*, vol. 651, pp. 1–34, 2016.

[14] Y. Wang, J. Wang, H. Wang, R. Zhang, and M. Li, "Users' mobility enhances information diffusion in online social networks," *Information Sciences*, vol. 546, pp. 329–348, 2021.

[15] Y. Wu, M. Li, J. Xiao, M. A. Orgun, and L. Xue, "The impact of discrimination on the spread of infectious diseases in complex networks," *New Generation Computing*, vol. 32, no. 3-4, pp. 193–211, 2014.

[16] Y. Li, H. V. Zhao, and Y. Chen, "An epidemic model for correlated information diffusion in crowd intelligence networks," *International Journal of Crowd Science*, vol. 3, no. 2, pp. 168–183, 2019.

[17] R. Narang, S. Sarin, P. Singh, and R. Goyal, "Impact of reciprocity in information spreading using epidemic model variants," *Information*, vol. 9, no. 6, p. 136, 2018.

[18] B. Xu and L. Liu, "Information diffusion through online social networks," in *2010 IEEE International Conference on Emergency Management and Management Sciences*, pp. 53–56, Beijing, China, 2010.

[19] V. D. Blondel, J. L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. 10008, 2008.

[20] G. X. Sun and S. Bin, "A new opinion leaders detecting algorithm in multi-relationship online social networks," *Multimedia Tools and Applications*, vol. 77, no. 4, pp. 4295–4307, 2018.

[21] H. Gong, C. Guo, and Y. Liu, "Measuring network rationality and simulating information diffusion based on network structure," *Physica A: Statistical Mechanics and its Applications*, vol. 564, article 125501, 2021.

[22] M. Yan, S. Li, C. A. Chan, Y. Shen, and Y. Yu, "Mobility prediction using a weighted Markov model based on mobile user classification," *Sensors*, vol. 21, no. 5, p. 1740, 2021.

[23] M. Yan, H. Yuan, Z. Li, Q. Lin, and J. Li, "Energy savings of wireless communication networks based on mobile user environmental prediction," *Journal of Environmental Protection and Ecology*, vol. 22, no. 1, pp. 206–217, 2021.

[24] H. Lu, M. Zhang, and S. Ma, "Between clicks and satisfaction: study on multi-phase user preferences and satisfaction for online news reading," in *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*, pp. 435–444, Ann Arbor, MI, USA, 2018.

[25] C. Wu, F. Wu, T. Qi, and Y. Huang, "User modeling with click preference and reading satisfaction for news recommendation," in *International Joint Conference on Artificial Intelligence*, pp. 3023–3029, Yokohama, Japan, 2020.

[26] S. Wallace, R. Treitman, N. Kumawat, and K. Arpin, "Towards readability individuation: the right changes to text format make large impacts on reading speed," *Journal of Vision*, vol. 20, no. 10, pp. 17–17, 2020.

[27] B. Shaver-Troup, K. Crowley, and M. Jordan, "Optimizing reading performance by manipulating the shape, size, and spacing of text to match the individual's visual processing capacity," Collective Impact Project, rev Reading, 2017.

Hindawi
Wireless Communications and Mobile Computing
Volume 2021, Article ID 2891303, 13 pages
https://doi.org/10.1155/2021/2891303

Research Article

SAL-Net: Self-Supervised Attribute Learning for Object Recognition and Segmentation

Shu Yang[1], **JingWang**[1], **Sheeraz Arif**[2], **Minli Jia**[3] and **Shunan Zhong**[1]

[1]*School of Information and Electronics, Beijing Institute of Technology, Beijing 100081, China*
[2]*Department of Computer Science, Faculty of Information Technology, Salim Habib University, Karachi 75400, Pakistan*
[3]*Research Institute of China Mobile Communications Co., Ltd., Beijing 100053, China*

Correspondence should be addressed to JingWang; wangjing@bit.edu.cn

Received 30 June 2021; Accepted 7 September 2021; Published 30 September 2021

Academic Editor: Ming Yan

Copyright © 2021 Shu 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.

Existing attribute learning methods rely on predefined attributes, which require manual annotations. Due to the limitation of human experience, the predefined attributes are not capable enough of providing enough description. This paper proposes a self-supervised attribute learning (SAL) method, which automatically generates attribute descriptions by differentially occluding the object region to deal with the above problems. The relationship between attributes is formulated with triplet loss functions and is utilized to supervise the CNN. Attribute learning is used as an auxiliary task of a multitask image classification and segmentation network, in which self-supervision of attributes motivates the CNN to learn more discriminative features for the main semantic tasks. Experimental results on public benchmarks CUB-2011 and Pascal VOC show that the proposed SAL-Net can obtain more accurate classification and segmentation results without additional annotations. Moreover, the SAL-Net is embedded into a multiobject recognition and segmentation system, which realizes instance-aware semantic segmentation with the help of a region proposal algorithm and a fusion nonmaximum suppression algorithm.

1. Introduction

Visual attributes are designed as midlevel semantic features to describe objects. However, most of the existing attribute learning methods [1–7] rely on manually annotated datasets [8–11]. In these datasets, domain experts predefined semantic attributes, and then, annotators give their judgments on whether a certain attribute appears in each object. On the one hand, the higher the dimensionality of the predefined attributes, the longer it takes to label each object, making attributes with limited dimensions more effective in obtaining large-scale datasets with limited resources. On the other hand, the predefined attributes with limited dimensions cannot fully describe the object, reducing the attribute dataset's usability. Since there are always some potential attributes essential for object representation, the predefined attribute is not capable enough to describe the real world, even with high dimensionality.

With the increasing demand for large-scale training sets for deep learning models [12–22], some researchers propose self-supervised learning methods [23–26], which train CNNs by automatic generation of labels based on the structure or characteristics of the image itself. We observe that each attribute of the object corresponds to a particular part of the object region. For example, the attribute "blue wings" depends on the visual features inside the wings. Certain attributes are hidden by occluding a part of the object region, making the occluded object have a slightly different attribute distribution than the original object. Inspired by the aforementioned fact, this research article proposes an automatic attribute generation method by occluding the object twice to obtain augmented objects with different attributes from the original object. The relationship between attributes can be formulated and utilized as the self-supervision to learn an attribute CNN. Based on the proposed method, this paper extends our previous work [27] by replacing the manually annotated supervision with the formulated attribute relationship to improve object recognition and segmentation.

The contributions of this paper can be summarized as follows:

(1) A self-supervised attribute learning (SAL) method is proposed by automatically generating attribute relationships to learn the CNN

(2) We design a multitask deep neural network SAL-Net, in which attribute learning is an auxiliary self-supervised task to alleviate intraclass variations in recognition tasks and refine the segmentation results

(3) We embed the SAL-Net into an instance-aware semantic segmentation system, in which a Fusion NMS algorithm is proposed to deal with repeated extraction of the same object

The remainder of this paper is organized as follows. The following section presents some relevant studies, including attribute learning methods and self-supervised learning methods. Then, the methodology of the proposed self-supervised attribute learning approach is described, and extensive experiments have been carried out to compare with baseline and state-of-the-art methods on two public datasets. Finally, the article is concluded.

2. Related Work

2.1. Attribute Learning. Visual attributes denote some semantic characteristics, which can be predefined to describe objects, including color, shape, and material. Lampert et al. [5] developed an AWA dataset consisting of 37,322 images of 50 animal classes and provided an 85-dimensional attribute distribution for each class. Deng et al. [9] composed a PETA dataset containing 19,000 images of 8705 pedestrians and provided 61 binary and four multiclass attributes for each pedestrian. Farhadi et al. [8] annotated 12,679 samples of Pascal VOC [28] to develop an attribute dataset aPY, where each sample is labeled with a 64-dimensional attribute description. Patterson and Hays [11] provided 204-dimensional attribute vectors on 29 categories of samples in COCO [29], while only 18,073 samples are labeled with over ten positive attributes.

Attribute learning has been widely used in computer vision fields, such as fine-grained recognition [2–4], zero-shot learning [5–7], face analysis [30–33], person reidentification [9, 34–38], and semantic segmentation [39–41]. Akata et al. [6] presented a model that embedded each category label in the space of attribute vectors for unseen object recognition. Han et al. [4] proposed an attribute-aware attention model to learn local attribute representation and global category representation for fine-grained image classification. In [37], pedestrian attributes and ID labels have been jointly learned with a shared backbone CNN, where these attributes are combined with low-level visual features to improve reidentification accuracy. Zhang et al. [38] jointly learned the center position, scale, offset, and semantic attributes in a multitask CNN for pedestrian detection tasks. In [41], an end-to-end attribute-aware semantic segmentation method was proposed by replacing the single category class *person* with nine orientation attribute classes.

However, most of the existing attribute learning methods are limited by the representative ability of human-defined attributes. Therefore, some researchers have proposed to learn latent attributes to obtain more discriminative features. Fu et al. [42] introduced a concept of semilatent attribute space, expressing user-defined and latent attributes in a unified framework, and proposed a novel scalable probabilistic topic model for learning multimodal semilatent attributes. Peng et al. [43] proposed a novel dictionary learning model which decomposes the dictionary space into three parts corresponding to semantic, latent discriminative, and latent background attributes. Li et al. [44] proposed an end-to-end network that is capable of learning discriminative semantic representations in an augmented space introduced for both user-defined and latent attributes. Xie et al. [45] proposed a semantic dictionary learning approach to exploit the latent visual attributes and align the visual-semantic spaces at the same time. Unlike the aforementioned methods, which learn latent attributes complementary to human-defined attributes for cross-class transfer learning, this paper proposes a self-supervised attribute learning method for object recognition and segmentation, which does not require additional attribute annotations.

2.2. Self-Supervised Learning. Convolutional neural networks (CNNs) [12–15] are always end-to-end learned in a supervised style, obtaining more discriminative features than traditional hand-craft methods [46–48]. Large-scale training samples are easy to obtain, while their human-annotated labels are costly. Thus, some researchers have proposed self-supervised learning models, which automatically generate labels for specific tasks by processing the samples. Doersch et al. [23] trained a neural network that can predict relative position, where the position labels are generated by splitting the entire image into blocks. Zhang et al. [24] transformed colorful images into gray-scale images and proposed to learn latent semantic information of pixels from the colorization task. Gidaris et al. [25] trained a network to predict the rotation angle of the image, where training samples are obtained by rotating the original image by different angles.

Different from the above methods, which pretrain the CNN on a set of unlabeled data to build representations and utilize them for downstream tasks, some researchers leverage self-supervised learning with multitask learning. Lee et al. [26] embedded the rotation prediction task into the traditional classification task to attain a significant improvement in classification accuracy. In the proposed SAL-Net, self-supervised attribute prediction, image-level classification, and pixel-level segmentation are jointly learned to more generalized features for different semantic analysis tasks.

3. Methodology

In this section, the proposed self-supervised attribute learning method is described, including an attribute generation approach and the definition of attribute loss. Then, a multitask deep neural network, SAL-Net, is designed based on the proposed method. In the network, attribute learning performs as an auxiliary task to constrain the CNN to learn more discriminative representations for object recognition

and segmentation. Moreover, the proposed self-supervised attribute learning method is extended to instance-aware semantic segmentation by a multiobject recognition and segmentation system.

3.1. Attribute Generation. There is a correspondence between attributes and each local region of the object, e.g., the *yellow forehead* attribute of the bird depends on the features in the *forehead* region. Therefore, an automatic attribute generation method is proposed, generating two additional training samples and the attribute relationship between them by differentially occluding an original training sample. As shown in Figure 1, the original sample $A = \{\mathbf{I}^A, \mathbf{c}^A, \mathbf{s}^A\}$ includes an image $\mathbf{I}^A$, a category label $\mathbf{c}^A$, and a mask $\mathbf{s}^A$. According to the contour information provided by the mask, the object's bounding box can be obtained. The area inside the bounding box is divided into 3×3 blocks, labeled $1-9$. Two blocks are randomly selected, and the first block is occluded to obtain a new sample $B = \{\mathbf{I}^B, \mathbf{c}^B, \mathbf{s}^B\}$. The pixels in the occluded block are set to a constant, as given in

$$\begin{cases} \mathbf{I}(i,j)[0] = 0.406 \times 256, \\ \mathbf{I}(i,j)[1] = 0.456 \times 256, \\ \mathbf{I}(i,j)[2] = 0.485 \times 256, \end{cases} \tag{1}$$

where mean $= [0.406, 0.456, 0.485]$ denotes the normalized mean BGR value among all pixels of the ImageNet dataset [49] and (i, j) represents the position of each occluded pixel. In the mask $\mathbf{s}^B$, the occluded pixel is set to 0. Based on sample B, the second block is occluded to obtain a new sample $C = \{\mathbf{I}^C, \mathbf{c}^C, \mathbf{s}^C\}$. In this way, the original single training sample A is expanded into a group $\{A, B, C\}$. The positive attributes of the occluded sample are more likely to be less than the positive attributes of the sample before the occlusion. In particular, sample C is more occluded than sample B, which results in the attribute difference between C and A greater than the difference between B and A. The attribute relationship is described in

$$0 \le \left|\mathbf{a}^B - \mathbf{a}^A\right| \le \left|\mathbf{a}^C - \mathbf{a}^A\right|, \tag{2}$$

where $\mathbf{a}^A$, $\mathbf{a}^B$, and $\mathbf{a}^C$ represent the attribute distribution of samples A, B, and C. $|\cdot|$ is a measure of the difference between two attribute distributions. The equal sign $\le$ means that when the occluded block does not contain any attributes, the occluded sample maintains the original attribute distribution. In the classification task, the category label of sample A is directly applied to B and C, i.e., $\mathbf{c}^A = \mathbf{c}^B = \mathbf{c}^C$, which makes the learned features invariant to occlusion for categorization.

3.2. Triplet Loss. A triplet loss function is defined to learn the attribute prediction model, which consists of two parts.

The one constrains the attribute difference between C and A greater than the attribute difference between B and A:

$$\mathscr{L}_1(\Theta) = \left[\left\|P_a(I^B,\Theta) - P_a(I^A,\Theta)\right\|_2^2 - \left\|P_a(I^C,\Theta) - P_a(I^A,\Theta)\right\|_2^2 + \alpha\right]_+, \tag{3}$$

where $P_a(\mathbf{I}, \Theta) \in \mathbb{R}^{K_a}$ represents the predicted attribute distribution of the input image $\mathbf{I}$ based on the parameter set Θ. $\|\cdot\|_2$ denotes the Euclidean distance between two distributions. α is the minimum margin between two metrics:

$$[x]_+ = \begin{cases} x, & x \ge 0, \\ 0, & x < 0. \end{cases} \tag{4}$$

It means that when the distance between $P_a(\mathbf{I}^A, \Theta)$ and $P_a(\mathbf{I}^C, \Theta)$ is larger than the distance between $P_a(\mathbf{I}^B, \Theta)$ and $P_a(\mathbf{I}^A, \Theta)$ plus α, the cost will be 0. Thus, by minimizing $\mathscr{L}_1(\Theta)$, we can keep A away from C and bring A close to B in the attribute space.

However, $\mathscr{L}_1(\Theta)$ can be minimized via $\|P_a(\mathbf{I}^B, \Theta) - P_a(\mathbf{I}^A, \Theta)\|_2^2 \longrightarrow 0$. It results in the same attribute description for A and B, which is not our expectation. Thus, an additional triple loss function is defined:

$$\mathscr{L}_2(\Theta) = \left[\alpha - \left\|P_a\left(\mathbf{I}^B,\Theta\right) - P_a\left(\mathbf{I}^A,\Theta\right)\right\|_2^2\right]_+. \tag{5}$$

It means that when the distance between $P_a(\mathbf{I}^B, \Theta)$ and $P_a(\mathbf{I}^A, \Theta)$ is smaller than α, there will be a cost.

By simultaneously minimizing $\mathscr{L}_1$ and $\mathscr{L}_2$, we can learn appropriate parameters to realize the attribute relationship represented in Equation (2). The randomly selected block probably contains attributes for the entire training set, so it is reasonable to set α to a positive constant rather than 0.

3.3. SAL-Net. In this section, an object recognition and segmentation network SAL-Net is designed based on the proposed self-supervised attribute learning method. As shown in Figure 2, the end-to-end network consists of three main components highlighted in different colors:

(1) A shared feature coding module (the green block). It consists of a backbone CNN to extract image features and an attribute feature coding model to obtain attribute feature maps for segmentation and recognition tasks

(2) A segmentation module (the orange block). It projects the attribute feature maps into multiscale segmentation predictions

(3) A classification module (the blue block). It projects semantic features to a smooth category space and combines the obtained category description with the attribute description to predict the category label

3.3.1. Attribute Feature Encoding. In CNN, features extracted by different layers contain different attribute information. Local features are extracted by low-level layers, e.g., yellow forehead and blue wing, while holistic features are

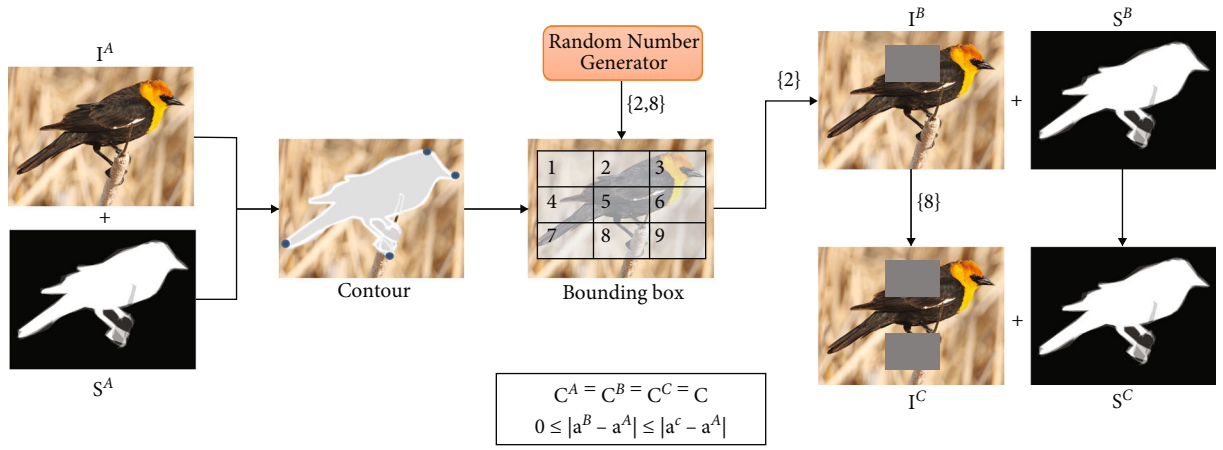


Figure 1: Illustration of the proposed automatic attribute generation approach.

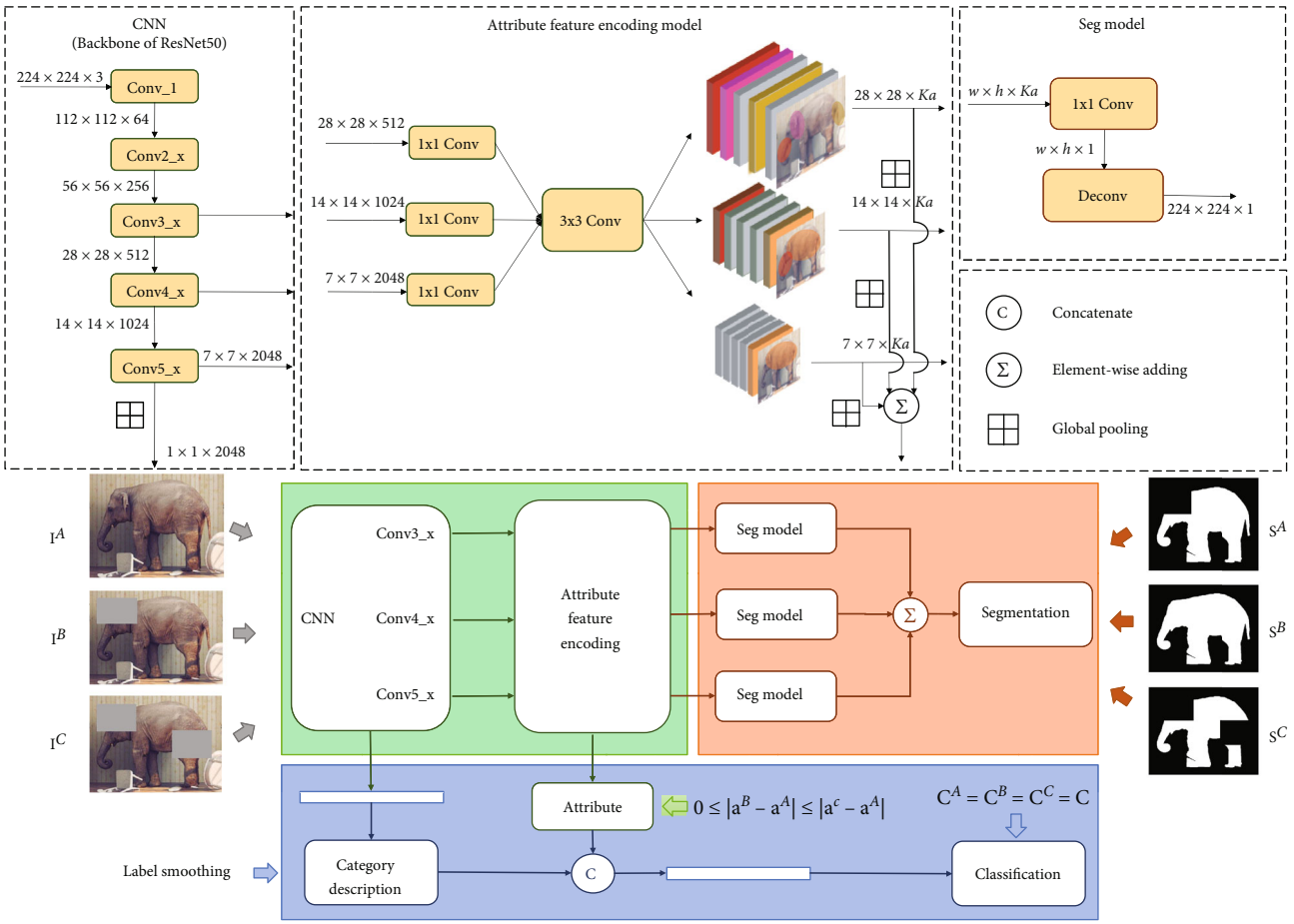


Figure 2: The architecture of the proposed SAL-Net. It contains (1) a shared feature encoding module (the green block), in which features from multiple CNN layers are encoded as attribute features under the supervision of attribute relationships; (2) a segmentation module (the orange block), which maps the attribute features into coarse-to-fine segments; (3) a classification branch (the blue block), which predicts category labels with the combination of category and attribute description.

extracted by high-level layers, e.g., chicken-like shape. Training with different features can improve the robustness of the attribute model. Therefore, we encode the visual feature $\psi^k \in \mathbb{R}^{D^k \times h^k \times w^k}$ with a shared attribute coding module f_e and obtain the attribute-wise feature $\psi_e^k \in \mathbb{R}^{K_a \times h^k \times w^k}$:

$$\psi_e^k(\Theta) = f_e\left(\psi^k, \Theta_t^k, \Theta_e\right), \quad k \in [1, K]. \tag{6}$$

There are $K = 3$ layers selected in the backbone CNN, i.e., the Conv3_x, Conv4_x, and Conv5_x. Since the dimensions of the features extracted from different layers are

different, a convolutional layer Θ_t^k is applied to convert D^k feature maps into $C = 256$ feature maps. Then, a shared 3×3 convolutional layer Θ_e is utilized for encoding the features as K_a-dimensional attribute-wise feature maps.

The encoded K features are fused to obtain the attribute prediction $P_a \in [0, 1]^{K_a}$:

$$P_a = \frac{1}{1 + e^{-\sum_{k=1}^{K} V^k}}, \quad k_a \in [1, K_a], \tag{7}$$

where each item in $V^k \in \mathbb{R}^{K_a}$ denotes the maximum in each attribute feature map ψ_e^k.

According to Equations (3) and (5), the attribute loss function is defined as

$$\begin{aligned} \mathscr{L}_a(\Theta) = \mathscr{L}_1 + \mathscr{L}_2 = \Big[& \left\| P_a\left(\mathbf{I}^B, \Theta\right) - P_a\left(\mathbf{I}^A, \Theta\right) \right\|_2^2 \\ & - \left\| P_a\left(\mathbf{I}^C, \Theta\right) - P_a\left(\mathbf{I}^A, \Theta\right) \right\|_2^2 + \alpha \Big]_+ \\ & + \left[\alpha - \left\| P_a\left(\mathbf{I}^B, \Theta\right) - P_a\left(\mathbf{I}^A, \Theta\right) \right\|_2^2 \right]_+. \end{aligned} \tag{8}$$

3.3.2. Segmentation Module. It consists of K segmentation models $\{\Theta_m^k\}_{k=1}^K$. Each segmentation model maps an attribute-wise feature ψ_e^k into a segmentation map $\psi_m^k \in \mathbb{R}^{H \times W}$:

$$\psi_m^k(\Theta) = f_m\left(\psi_e^k, \Theta_m^k\right), \quad k \in [1, K]. \tag{9}$$

The segmentation model f_m consists of a 1×1 convolutional layer to localize the object and a deconvolutional layer to upsample the segmentation to the same size as the original image.

The final segmentation is obtained by element-wise adding K coarse-to-fine segmentation maps and then performing the sigmoid operation:

$$P_m = \frac{1}{1 + e^{-\sum_{k=1}^{K} \psi_m^k}}. \tag{10}$$

$P_m \in [0, 1]^{H \times W}$ represents the probability of each pixel belonging to the foreground object.

The segmentation loss function is defined as the cross-entropy between the prediction P_m and the mask s provided by the dataset. $\mathscr{H}(v, u)$ is defined as the cross-entropy of the ground-truth v and the prediction u:

$$\mathscr{H}(v, u) = -v \log u - (1 - v) \log (1 - u). \tag{11}$$

Three training samples are fed into the network during the training phase. Thus, the segmentation loss function is defined as

$$\begin{aligned} \mathscr{L}_m(\Theta) = \frac{1}{W \times H} \sum_{w=1}^{W} \sum_{h=1}^{H} \Big[& \mathscr{H}\left(s^A, P_m\left(I^A, \Theta\right)\right)_{w,h} \\ & + \mathscr{H}\left(s^B, P_m\left(I^B, \Theta\right)\right)_{w,h} + \mathscr{H}\left(s^C, P_m\left(I^C, \Theta\right)\right)_{w,h} \Big]. \end{aligned} \tag{12}$$

3.3.3. Classification Module. It combines category descriptions and attribute descriptions to classify the image. The semantic feature ψ, which is the output of the global pooling layer in the backbone CNN, is encoded into a category description feature P_{cate}:

$$P_{\text{cate}}(\Theta) = f_{\text{cate}}(\psi, \Theta_c), \quad P_{\text{cate}} \in [0, 1]^{K_c}. \tag{13}$$

f_{cate} denotes a fully connected layer with softmax activation. The obtained description P_{cate} represents the location of the input sample in a category description space Q. The space Q is defined by applying the label smoothing [15] strategy on the typical one-hot encoding category labels:

$$Q(k_c) = \begin{cases} 1 - \varepsilon + \dfrac{\varepsilon}{K_c}, & k_c = \mathbf{c}, \\ \dfrac{\varepsilon}{K_c}, & \text{otherwise.} \end{cases} \tag{14}$$

The loss function of the category feature is defined as

$$\begin{aligned} \mathscr{L}_{\text{cate}}(\Theta) = & - \sum_{k_c=1}^{K_c} Q(k_c) \log \left(P_{\text{cate}}\left(\mathbf{I}^A, \Theta, k_c\right)\right) \\ & - \sum_{k_c=1}^{K_c} Q(k_c) \log \left(P_{\text{cate}}\left(\mathbf{I}^B, \Theta, k_c\right)\right) \\ & - \sum_{k_c=1}^{K_c} Q(k_c) \log \left(P_{\text{cate}}\left(\mathbf{I}^C, \Theta, k_c\right)\right) \\ = & -(1 - \varepsilon) \log \left(\prod_{A,B,C} P_{\text{cate}}\left(I^i, \Theta, c\right) \right) \\ & - \varepsilon \sum_{k_c=1}^{K_c} \frac{1}{K_c} \log \left(\prod_{A,B,C} P_{\text{cate}}\left(I^i, \Theta, k_c\right) \right). \end{aligned} \tag{15}$$

$\prod_{A,B,C}$ represents the product of the category features of three samples. Then, the category feature P_{cate} and the attribute feature P_a are concatenated to obtain a $(K_c + K_a)$-dimensional semantic feature vector, which is utilized to predict the category label with a classifier:

$$P_c(\Phi) = f_c(P_{\text{cate}}, P_a, \Phi), \quad P_c \in [0, 1]^{K_c}. \tag{16}$$

Φ denotes the parameter set of the classifier, e.g., a fully connected layer with softmax activation works well. An additional loss is defined to train the classifier:

$$\mathscr{L}_c(\Phi) = -\log\left(P_c\left(P^A_{\text{cate}}, P^A_a, \Phi, c\right)\right) - \log\left(P_c\left(P^B_{\text{cate}}, P^B_a, \Phi, c\right)\right) - \log\left(P_c\left(P^C_{\text{cate}}, P^C_a, \Phi, c\right)\right) = -\log\left(\prod_{A,B,C} P_c(P_{\text{cate}}, P_a, \Phi, c)\right). \tag{17}$$

In the training phase, we automatically generate N group samples $\{(A, B, C)_i\}_{i=1}^N$ based on N original samples and utilize these samples to optimize parameters of the above SAL-Net by minimizing the total cost of three tasks:

$$\Theta^*, \Phi^* = \underset{\Theta,\Phi}{\operatorname{argmin}} \frac{1}{N}\sum_{i=1}^{N}[\mathscr{L}_a(\Theta) + \mathscr{L}_m(\Theta) + \mathscr{L}_{\text{cate}}(\Theta) + \mathscr{L}_c(\Phi)]. \tag{18}$$

3.4. Multiobject System. In this section, a multiobject recognition and segmentation system is designed based on the proposed SAL-Net to achieve instance-aware semantic segmentation. As shown in Figure 3, the system consists of four parts: an attribute generation module, a region proposal module, the SAL-Net, and a postprocessing module.

3.4.1. Candidate Region Proposal Module. This module automatically generates candidate regions from the input image based on the selective search [50] algorithm. In the training phase, this module generates negative samples. We calculate the IoU (intersection over union) score between each proposal and ground-truth regions. The proposal with a small maximum IoU, e.g., smaller than 0.1, is taken as a "background" sample. In the testing phase, this module generates testing samples which are fed into the SAL-Net.

3.4.2. Attribute Generation Module. It generates positive samples for training SAL-Net, as shown in Figure 1. If a training image contains N_+ objects, then N_+ groups of positive samples can be obtained, where each group contains three samples $\{A, B, C\}$.

3.4.3. SAL-Net Module. The structure is similar to Figure 2, except N_- negative samples being utilized for training along with N_+ groups of positive samples. We add a background class into the category description space; that is, the category feature P_{cate} becomes a $(K_c + 1)$-dimensional vector. It is worth noticing that negative samples are not for training on attributes and segmentation tasks. Equation (18) is modified with the following given formula:

$$\Theta^*, \Phi^* = \underset{\Theta,\Phi}{\operatorname{argmin}} \frac{1}{N_+}\sum_{i=1}^{N_+}[\mathscr{L}_a(\Theta) + \mathscr{L}_m(\Theta) + \mathscr{L}_{\text{cate}}(\Theta) + \mathscr{L}_c(\Phi)] + \frac{1}{N_-}\sum_{i=1}^{N_-}[\mathscr{L}_{\text{cate}}(\Theta) + \mathscr{L}_c(\Phi)]. \tag{19}$$

In the testing phase, we use the category description P_{cate} as the basis for distinguishing the "object/background." If the maximum probability of P_{cate} appears on the background class, that means the testing sample belongs to the background; otherwise, the maximum probability of P_c is taken as the confidence of the sample, and the corresponding category is regarded as the recognition result.

3.4.4. Postprocessing Module. In the testing phase, this module filters the recognition and segmentation results of multiple test samples to obtain the final instance-aware semantic segmentation results. A fusion nonmaximum suppression algorithm is proposed to deal with repeated extraction of the same object. As described in Algorithm 1, we calculate the segmentation IoU of each pair of predictions with the same category. When the IoU is larger than a threshold (set to 0.2), we remove the prediction with lower confidence and fuse its segmentation into the other prediction.

4. Experiments

4.1. Datasets and Implementation Details. This section conducted a series of experiments on two public benchmarks, including CUB-2011 [10] and Pascal VOC [28].

(i) *CUB-2011* [10]. Wah et al. [10] established a dataset with 11,788 images from 200 bird categories. Each bird is annotated with a 312-dimensional attribute distribution. The standard training/testing split (i.e., 5994/5794) is adopted. In the training phase, each image is resized into 224×224, and the batch size is set to 64 for both SAL-Net and baseline models. All networks are trained 50,000 iterations with a learning rate of 0.001, and then 30,000 iterations with a learning rate of 0.0001, and finally 20,000 iterations with a learning rate of 0.00001.

(ii) *Pascal VOC* [28]. The Pascal VOC dataset is adopted to evaluate the effectiveness of both SAL-Net and the proposed multiobject system. We use 1464 images for training and 1449 images for testing. In the training phase, 3508 groups of positive samples and 1464 negative samples have been used. The batch size is set to the number of samples from 4 training images, and each cropped sample is resized into 160×160. The training for both SAL-Net and baseline models is performed in three stages. First, it performs 20,000 iterations with a learning rate of 0.001, then performs 12,000 iterations with a learning rate of 0.0001, and finally performs 8000 iterations with a learning rate of 0.00001. In order to verify the performance of SAL-Net, a bounding box is provided to extract each object in the testing image and use it as input. In order to verify the performance of the proposed system, the entire testing image is fed into the system.

We compute the average intersection over union (aIoU) among testing samples to evaluate the segmentation performance and the accuracy (Acc) to evaluate the classification performance. For instance-aware semantic segmentation, we calculate the mean average precision and recall among categories (mask AP and AR defined in [29]).

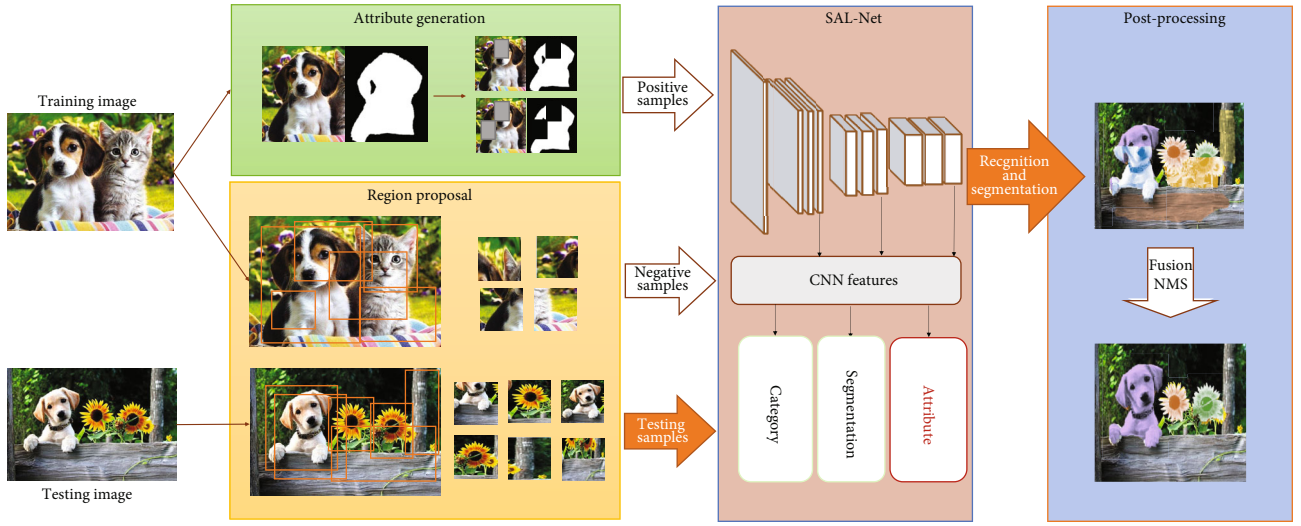


Figure 3: The proposed multiobject recognition and segmentation system pipeline. In the training phase, the input image is fed into the attribute generation module to obtain positive samples and the region proposal module to obtain negative samples. Then, the positive and negative samples are utilized to train the parameters of SAL-Net. In the testing phase, the input image is fed into the region proposal module to obtain testing samples, which are sent to the SAL-Net for prediction. Finally, the recognition and segmentation results are sent to the postprocessing module for filtering.

Input: Mask set M, category set C, confidence set P
Output: updated M, C and P
[1] Sort M, C and P according to the value in P from large to small;
[2] Initialize the number of predictions k.
[3] **for** $m_i \in M, c_i \in C, p_i \in P$ **do**
[4] **for** $j \in (i, k]$ **do**
[5] **if** $c_i == c_j$ **then**
[6] Calculate $IoU(m_i, m_j)$;
[7] **if** $IoU > T_{iou}$ **then**
[8] Fuse the prediction: $m_i = m_i \bigcup m_j$;
[9] Remove the prediction: $M = M - m_j, C = C - c_j, P = P - p_j$;
[10] $k = k - 1$;
[11] **end if**
[12] **end if**
[13] **end for**
[14] **end for**
[15] **return** Updated M, C and P

Algorithm 1: The proposed fusion nonmaximum suppression algorithm (Fusion NMS).

4.2. Comparative Methods. We compare the following degenerated models with the proposed SAL-Net. All comparative models are based on the ResNet50 [14], which is pretrained on the ImageNet dataset [49].

(i) *ST-cls (A Single-Task Model for Classification).* The output of Conv5_x is fed into a global pooling layer to obtain a 2048-dimensional feature vector, which is projected onto the category space with a fully connected layer.

(ii) *ST-seg (A Single-Task Model for Segmentation).* The output of Conv5_x is fed into a 1×1 convolutional layer and a deconvolutional layer to obtain the foreground confidence map.

(iii) *MT-SF (A Multitask Model with Single-Layer Features).* This is a two-branch multitask network, where the classification task and the segmentation task share the CNN backbone. The output of Conv5_x is fed into a classification branch and a segmentation branch, which have the same structure as ST-cls and ST-seg, respectively.

(iv) *MT-MF (A Multitask Model with Multilayer Features).* The network has a similar structure to SAL-Net except for the auxiliary attribute. For the classification branch, the Conv5_x feature is fed into a global pooling layer to obtain a 2048-dimensional vector. For the segmentation branch, features from Conv3_x, Conv4_x, and Conv5_x are, respectively,

Table 1: Effectiveness of the proposed strategies, including multitask learning, features from multiple CNN layers, and joint learning with attributes.

Model	Strategies			CUB-2011 [10]		Pascal VOC [28]	
	Multitask	Multifeature	Attribute learning	Acc (%)	aIoU (%)	Acc (%)	aIoU (%)
ST-cls				64.67	—	67.46	—
ST-seg				—	79.76	—	73.81
MT-SF	✓			70.37	74.22	69.11	74.02
MT-MF	✓	✓		71.45	82.61	71.42	77.59
AFE-Net [27]	✓	✓	✓	74.37	85.07	—	—
SAL-Net (ours)	✓	✓	✓	77.55	86.66	75.00	78.19

fed into three segmentation projection models; each model contains a 1×1 convolutional layer and a deconvolutional layer. The outputs of the three segmentation models are fused into the final prediction.

(v) *AFE-Net* [27] *(Our Previous Attribute-Aware Feature Encoding Network)*. It projects the visual features onto attribute-wise feature maps before sending the features to the segmentation branch. At the same time, the classification branch adopts the strategy of smoothing the category space and combining attributes and category descriptions. Compared with SAL-Net, the only difference is that such a baseline applies supervision from manually annotated attributes, while SAL-Net is based on the proposed self-supervised attribute learning method.

4.3. Performance of SAL-Net

4.3.1. Comparisons with Baseline Models. The proposed SAL-Net is compared with baseline models in Table 1. Compared with the single-task model ST-cls, the multitask model MT-SF obtains better classification results. It is thanks to the segmentation supervision signal transferring the object contour information to the classification task. The extracted object region contains fewer background features' adverse effects to improve the stability of the classification model. However, MT-SF has limited advantages over the single-task model ST-seg in the segmentation task. The additional classification supervision motivates the CNN to extract semantic features rather than location features. The above problem can be solved by applying the multilayer feature strategy. Compared with MT-SF, MT-MF significantly improves the segmentation accuracy, which obtains a gain of 8.39% aIoU on the CUB-2011 dataset and 3.57% on the Pascal VOC dataset.

In our previous work [27], attribute knowledge is applied to constrain the feature extraction, which has made significant improvements in both classification and segmentation tasks on the Pascal VOC dataset. However, it is not suitable to Pascal VOC since there are no attribute annotations provided by the dataset. This paper proposes a self-learning attribute learning (SAL) method. Compared with MT-MF, SAL-Net produces a 4.05% aIoU gain and a 6.1% accuracy gain on the CUB-2011 dataset. On the Pascal VOC dataset, it obtains 0.6% aIoU gain and 3.55% accuracy gain. The reason is that SAL-Net introduces attribute relationships to constrain the feature extraction network and smooth the category space, which makes the extracted features more discriminative and robust. Meanwhile, concatenating the attribute features to the category description can further improve classification accuracy. Compared with [27], SAL-Net produces a segmentation gain of 1.59% aIoU and a classification gain of 3.18% accuracy on the CUB-2011 dataset. It can be explained by the advantages of latent attributes, where the attribute knowledge automatically mined from images are more suitable than manually annotated attributes for identifying and segmenting objects.

Table 2: Comparison with state-of-the-art fine-grained recognition methods on CUB-2011. Extra annotations are provided during the training phase, including attribute (Att), bounding box (BBox), and part location (Part).

Methods	Annotations			Acc (%)
	Att	BBox	Part	
ResNet50 [14]				82.0
Bilinear-CNN [53]				84.1
AutoBD [54]				81.6
SAL-Net (ours)				84.6
Liu et al. [51]	✓			82.5
VTRL [52]	✓			86.3
CAP-Net [55]	✓	✓		82.5
A3M [4]	✓	✓		86.2
PBC [56]		✓	✓	83.7
BBPL [57]		✓	✓	87.6
Liu et al. [51]	✓	✓	✓	85.5

4.3.2. Comparisons with State-of-the-Art Methods. In Table 2, SAL-Net is compared with state-of-the-art fine-grained recognition methods on the CUB-2011 benchmark. A widespread data augmentation approach is applied, where each input image is resized into 512×512 and rotated with a random angle. The randomly cropped patch is resized and flipped into a 448×448 input image. Based on the data augmentation, the standard ResNet50 achieves high accuracy on the CUB-2011 testing set, while our SAL-Net improves the accuracy from 82.0% to 84.6%. Existing attribute-based methods [4, 27, 51, 52] also learn the attribute attention to

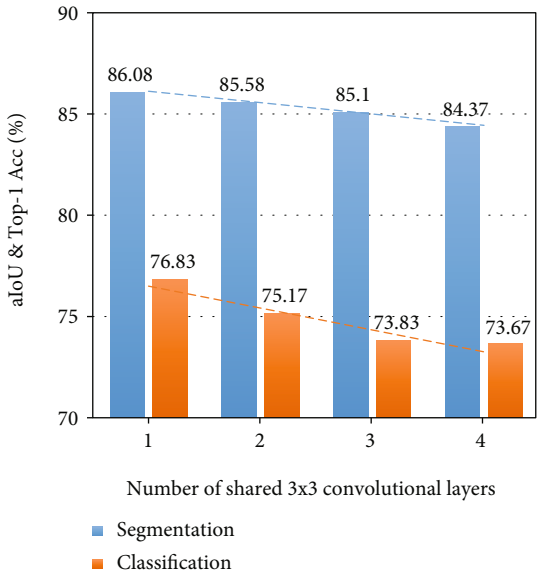


Figure 4: Classification and segmentation performance w.r.t. the number of shared 3×3 convolutional layers of the attribute feature coding module. The models are trained by randomly selecting 90% samples from the CUB-2011 training set and using the remaining 10% samples for validation.

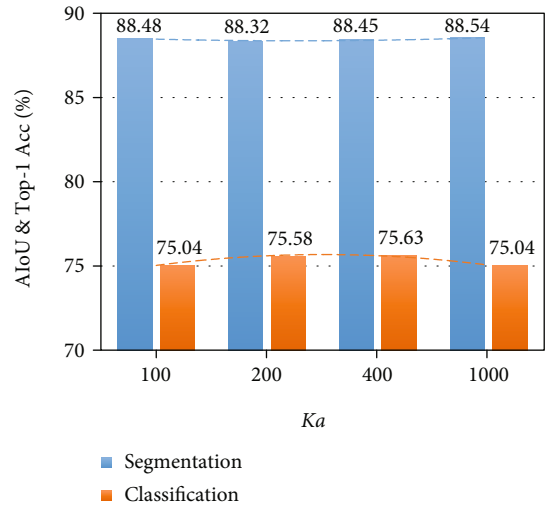


Figure 5: Classification and segmentation performance on CUB-2011 validation set w.r.t. the dimension of attributes.

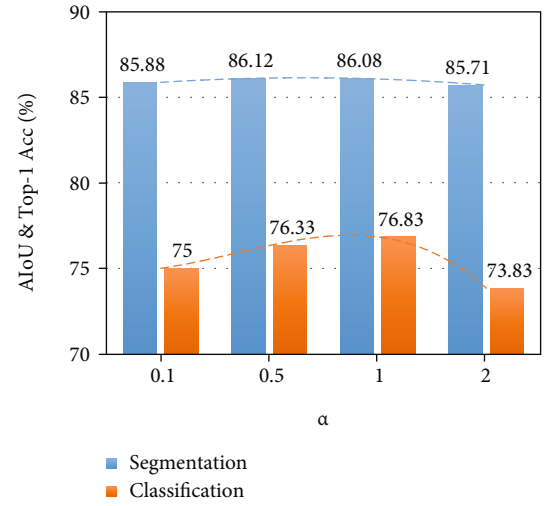


Figure 6: Classification and segmentation performance on CUB-2011 validation set w.r.t. the minimum margin between two attribute distributions (α in Equations (3) and (5)).

localize discriminative regions. These methods rely on the predefined attributes or textual descriptions provided by the dataset. In contrast, our SAL-Net automatically exploits latent attributes based on the self-supervised learning method, which does not need any extra annotations.

4.3.3. Effects of the Structure of Attribute Feature Coding Module. Figure 4 illustrates the classification and segmentation performance by using different attribute feature coding modules. The horizontal axis represents the number of shared 3×3 convolutional layers, which encode CNN features as K_a-dimensional attribute-wise feature maps. We can see that the performance of using one convolutional layer is relatively better. As the number of hidden layers increases, the supervision of the attributes imposes fewer constraints on the feature extraction. The result indirectly verifies the effectiveness of the proposed self-supervised attribute learning method for representation learning.

4.3.4. Effects of the Attribute Dimension. Figure 5 compares the classification and segmentation results by using different attribute dimensions. We can see that the change of the attribute dimension K_a has a limited effect on the segmentation task. The reason is that SAL-Net mainly refines the segmentation through the localization of latent attributes. When K_a is small, the area of each latent attribute is learned to be large, and when K_a is large, the area of each latent attribute is learned to be small. Thus, the fused segmentation result is stable. For the classification task, the network with a small K_a cannot learn enough attributes to describe the object. On the contrary, an excessively large K_a will weaken the category knowledge in the concatenated feature and decrease the recognition accuracy. In the Pascal VOC dataset, there are remarkable visual differences among the 20 categories. There are also noticeable visual differences between individuals from a common category, resulting in the entire dataset with diversified characteristics. Therefore, we select the value of $K_a = 1000$ to learn as many latent attributes as possible. In the fine-grained CUB-2011 dataset, 200 categories share attributes, and birds from a common category contain almost the same attribute distribution. Therefore, we choose $K_a = 400$, which is 88 higher than the predefined attributes provided by the dataset.

4.3.5. Effects of the Attribute Margin. Figure 6 illustrates the influence of the minimum margin α in Equation (3). We can see that when α is small, the occluded sample is closer to the original sample in the attribute space, which reduces the discrimination of the representation. When α is large, the attribute features of the occluded sample and the original sample

(a) (b) (c)

Figure 7: The visualization of each module in the proposed multitarget recognition and segmentation system: (a) region proposals generated by selective search algorithm; (b) SAL-Net predictions filtered by score threshold 0.6; (c) results processed by Fusion NMS.

are quite different, making it difficult for the concatenated features to be classified into the same category. Therefore, an appropriate $\alpha = 1$ is selected to learn discriminative attributes and ensure their occlusion invariance to category labels.

4.4. Performance of Instance-Aware Semantic Segmentation. Figure 7 shows the output of each module in the proposed multiobject recognition and segmentation system. It can be seen from Figure 7(a) that the region proposal module based on the selective search [50] can obtain multiple target candidates concentrated in the object region. Figure 7(b) illustrates the classification and segmentation results of SAL-Net. Each candidate region is fed into the network to obtain a binary segmentation mask and category confidence. Some background regions are filtered out with a threshold of 0.6 on the confidence score. However, there are still some multiple overlapping results for the same object. The proposed Fusion NMS approach is applied to process the above predictions. In Figure 7(c), we can see that most overlapping areas are filtered out, and the segmentation result obtained after fusion is more accurate.

Table 3 further demonstrates the effectiveness of the proposed Fusion NMS method. We can see that the baseline result without postprocessing has a higher recall and a lower precision. It is because many overlapping SAL-Net predictions are retained (as shown in Figure 7(b)). Mask NMS [58] can filter out overlapping results with lower scores, while the performance is sensitive to the accuracy of each candidate box. Based on the coarse boxes extracted by the SS algorithm, Mask NMS cannot obtain effective improvement. Matrix NMS [59] updates the confidence of all predictions instead of directly filtering out those low-score predictions, while its gain of AR and AP is limited. The

Table 3: Effectiveness of the proposed Fusion NMS on Pascal VOC.

Module	AP	AP_{50}	AP_{75}	AP_L	AR_L	AR_1	AR_{10}	AR_{100}
Without NMS (baseline)	2.1	6.0	0.8	2.9	20.0	6.6	14.0	14.7
Mask NMS [58]	2.5	8.2	0.7	3.5	11.1	6.6	8.4	8.4
Matrix NMS [59]	2.8	9.2	0.8	3.8	14.7	6.3	10.4	10.9
Fusion NMS (ours)	**7.7**	19.1	5.5	10.7	20.9	13.6	15.6	15.6

proposed Fusion NMS method is an advancement of Mask NMS [58]. We filter out the overlapping prediction with a lower score while fusing its segmentation into the retained prediction, making the final segmentation more accurate. As shown in Figure 7(c), even if the retained box does not contain the entire object, the segmentation after fusion is still satisfactory.

5. Conclusion

This paper proposes a self-supervised attribute learning method, which automatically generates latent attribute descriptions for training. Based on the proposed method, we design a multitask network SAL-Net, in which attributes constrain the feature coding network to learn more discriminative representations for recognition and segmentation. Experimental results on CUB-2011 and Pascal VOC datasets illustrate that the proposed SAL-Net can significantly improve object recognition and segmentation accuracy without extra annotations. It achieves state-of-the-art fine-grained recognition performance on CUB-2011, superior to existing methods based on predefined attributes. In addition, we naturally embed SAL-Net into a multiobject system to achieve instance-aware semantic segmentation. Qualitative and quantitative experiments on Pascal VOC verify the effectiveness of each module in the system.

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 they have no conflicts of interest.

Acknowledgments

This work was partially supported by the National Natural Science Foundation of China (Grant No.: 61620106002) and the Central Government Guides Local Science and Technology Development Fund Project of China (Grant No.: 2021ZY0004).

References

[1] V. Ferrari and A. Zisserman, "Learning visual attributes," in *Advances in Neural Information Processing Systems*, pp. 433–440, MIT Press, 2007.

[2] K. Duan, D. Parikh, D. Crandall, and K. Grauman, "Discovering localized attributes for fine-grained recognition," in *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3474–3481, Providence, RI, USA, June 2012.

[3] Y. Lu, L. Yang, K. Yang, and Y. Rui, "Mining latent attributes from click-through logs for image recognition," *IEEE Transactions on Multimedia*, vol. 17, no. 8, pp. 1213–1224, 2015.

[4] K. Han, J. Guo, C. Zhang, and M. Zhu, "Attribute-aware attention model for fine-grained representation learning," in *Proceedings of the 26th ACM international conference on Multimedia*, pp. 2040–2048, October 2018.

[5] C. H. Lampert, H. Nickisch, and S. Harmeling, "Learning to detect unseen object classes by between-class attribute transfer," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 951–958, Miami, FL, USA, June 2009.

[6] Z. Akata, F. Perronnin, Z. Harchaoui, and C. Schmid, "Label-embedding for image classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 7, pp. 1425–1438, 2015.

[7] B. Demirel, R. G. Cinbis, and N. Ikizler-Cinbis, "Attributes2Classname: a discriminative model for attribute-based unsupervised zero-shot learning," in *2017 IEEE International Conference on Computer Vision (ICCV)*, pp. 1241–1250, Venice, Italy, October 2017.

[8] A. Farhadi, I. Endres, D. Hoiem, and D. Forsyth, "Describing objects by their attributes," in *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1778–1785, Miami, FL, USA, June 2009.

[9] Y. Deng, P. Luo, C. C. Loy, and X. Tang, "Pedestrian attribute recognition at far distance," in *Proceedings of the 22nd ACM international conference on Multimedia*, pp. 789–792, Orlando, USA, November 2014.

[10] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, *The Caltech-UCSD Birds-200-2011 Dataset*, Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

[11] G. Patterson and J. Hays, "COCO attributes: attributes for people, animals, and objects," in *European Conference on Computer Vision*, pp. 85–100, Springer, 2016.

[12] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, https://arxiv.org/abs/1409.1556.

[13] C. Szegedy, W. Liu, Y. Jia et al., "Going deeper with convolutions," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, Boston, MA, USA, June 2015.

[14] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 770–778, Las Vegas, NV, USA, June 2016.

[15] C. Szegedy, V. Vanhoucke, S. Ioffe, J. Shlens, and Z. Wojna, "Rethinking the inception architecture for computer vision," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2818–2826, Las Vegas, NV, USA, June 2016.

[16] Y. Xing, L. Zhong, and X. Zhong, "An encoder-decoder network based FCN architecture for semantic segmentation," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 8861886, 9 pages, 2020.

[17] A. Derhab, A. Aldweesh, A. Z. Emam, and F. A. Khan, "Intrusion detection system for internet of things based on temporal convolution neural network and efficient feature engineering," *Wireless Communications and Mobile Computing*, vol. 2020, Article ID 6689134, 16 pages, 2020.

[18] M. Yan, Z. Li, X. Yu, and C. Jin, "An end-to-end deep learning network for 3D object detection from RGB-D data based on hough voting," *IEEE Access*, vol. 8, pp. 138810–138822, 2020.

[19] C. Jin, T. Zhang, S. Liu, and T. Yun, "Cross-modal deep learning applications: audio-visual retrieval," in *International Conference on Pattern Recognition Workshops*, pp. 301–313, Paris, France, 2021.

[20] C. Jin, Y. Tie, Y. Bai, X. Lv, and S. Liu, "A style-specific music composition neural network," *Neural Processing Letters*, vol. 52, no. 3, pp. 1893–1912, 2020.

[21] C. Jin, T. Wang, S. Liu et al., "A transformer-based model for multi-track music generation," *International Journal of Multimedia Data Engineering and Management*, vol. 11, no. 3, pp. 36–54, 2020.

[22] C. Jin, W. Zhao, and H. Wang, "Research on objective evaluation of recording audio restoration based on deep learning network," *Advances in Multimedia*, vol. 2018, Article ID 3748141, 13 pages, 2018.

[23] C. Doersch, A. Gupta, and A. A. Efros, "Unsupervised visual representation learning by context prediction," in *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1422–1430, Santiago, Chile, December 2015.

[24] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *European Conference on Computer Vision*, pp. 649–666, Springer, 2016.

[25] S. Gidaris, P. Singh, and N. Komodakis, "Unsupervised representation learning by predicting image rotations," in *International Conference on Learning Representations*, Vancouver, Canada, 2018.

[26] H. Lee, S. J. Hwang, and J. Shin, "Rethinking data augmentation: self-supervision and self-distillation," in *International Conference on Learning Representations*, Vienna, Austria, 2020.

[27] S. Yang, Y. Wang, K. Chen, W. Zeng, and Z. Fei, "Attribute-aware feature encoding for object recognition and segmentation," *IEEE Transactions on Multimedia*, p. 1, 2021.

[28] 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.

[29] T. Y. Lin, M. Maire, S. Belongie et al., "Microsoft COCO: common objects in context," in *European Conference on Computer Vision*, Springer, 2014.

[30] N. Kumar, A. Berg, P. N. Belhumeur, and S. Nayar, "Describable visual attributes for face verification and image search," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 10, pp. 1962–1977, 2011.

[31] H. Han, A. K. Jain, F. Wang, S. Shan, and X. Chen, "Heterogeneous face attribute estimation: a deep multi-task learning approach," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 11, pp. 2597–2609, 2018.

[32] M. Ehrlich, T. J. Shields, T. Almaev, and M. R. Amer, "Facial attributes classification using multi-task representation learning," in *2016 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 752–760, Las Vegas, NV, USA, June 2016.

[33] R. Ranjan, V. M. Patel, and R. Chellappa, "Hyperface: a deep multi-task learning framework for face detection, landmark localization, pose estimation, and gender recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 41, no. 1, pp. 121–135, 2019.

[34] S. Rui, H. Qiheng, F. Wei, and Z. Xudong, "Attributes-based person re-identification via CNNs with coupled clusters loss," *Journal of Systems Engineering and Electronics*, vol. 31, no. 1, pp. 45–55, 2020.

[35] Z. Shi, T. M. Hospedales, and T. Xiang, "Transferring a semantic representation for person reidentification and search," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4184–4193, Boston, MA, USA, June 2015.

[36] S. Chi, S. Zhang, J. Xing, W. Gao, and Q. Tian, "Deep attributes driven multi-camera person re-identification," in *European Conference on Computer Vision*, pp. 475–491, Springer, 2016.

[37] Y. Lin, L. Zheng, Z. Zheng et al., "Improving person re-identification by attribute and identity learning," *Pattern Recognition*, vol. 95, pp. 151–161, 2019.

[38] J. Zhang, L. Lin, J. Zhu, Y. Hu, and S. C. H. Hoi, "CSID: center, scale, identity and density-aware pedestrian detection in a crowd," 2019, https://arxiv.org/abs/1910.09188.

[39] S. Zheng, M.-M. Cheng, J. Warrell et al., "Dense semantic image segmentation with objects and attributes," in *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3214–3221, Columbus, OH, USA, June 2014.

[40] Z. Shi, Y. Yang, T. M. Hospedales, and T. Xiang, "Weakly-supervised image annotation and segmentation with objects and attributes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 12, pp. 2525–2538, 2017.

[41] M. D. Sulistiyo, Y. Kawanishi, D. Deguchi et al., "Attribute-aware semantic segmentation of road scenes for understanding pedestrian orientations," in *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pp. 2698–2703, Maui, HI, USA, November 2018.

[42] Y. Fu, T. M. Hospedales, T. Xiang, and S. Gong, "Learning multimodal latent attributes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 36, no. 2, pp. 303–316, 2014.

[43] P. Peng, Y. Tian, T. Xiang et al., "Joint semantic and latent attribute modelling for cross-class transfer learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 7, pp. 1625–1638, 2017.

[44] Y. Li, J. Zhang, J. Zhang, and K. Huang, "Discriminative learning of latent features for zero-shot recognition," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7463–7471, Salt Lake City, UT, USA, June 2018.

[45] Y. Xie, X. He, J. Zhang, and X. Luo, "Zero-shot recognition with latent visual attributes learning," *Multimedia Tools and Applications*, vol. 79, no. 37-38, pp. 27321–27335, 2020.

[46] D. G. Lowe, "Object recognition from local scale-invariant features," in *Proceedings of the Seventh IEEE International Conference on Computer Vision*, vol. 2, pp. 1150–1157, Kerkyra, Greece, September 1999.

[47] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference*

on Computer Vision and Pattern Recognition (CVPR'05), pp. 886–893, San Diego, CA, USA, June 2005.

[48] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: speeded up robust features," in *European Conference on Computer Vision*, pp. 404–417, Springer, 2006.

[49] O. Russakovsky, J. Deng, H. Su et al., "ImageNet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[50] J. R. R. Uijlings, K. E. A. van de Sande, T. Gevers, and A. W. M. Smeulders, "Selective search for object recognition," *International Journal of Computer Vision*, vol. 104, no. 2, pp. 154–171, 2013.

[51] X. Liu, J. Wang, S. Wen, E. Ding, and Y. Lin, "Localizing by describing: attribute guided attention localization for fine-grained recognition," in *AAAI Conference on Artificial Intelligence*, pp. 4190–4196, San Francisco, USA, 2017.

[52] X. He and Y. Peng, "Fine-grained visual-textual representation learning," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 2, pp. 520–531, 2020.

[53] T. Y. Lin, A. Roychowdhury, and S. Maji, "Bilinear cnn models for fine-grained visual recognition," in *2015 IEEE International Conference on Computer Vision (ICCV)*, pp. 1449–1457, Santiago, Chile, December 2015.

[54] H. Yao, S. Zhang, C. Yan, Y. Zhang, J. Li, and Q. Tian, "Autobd: automated bi-level description for scalable fine-grained visual categorization," *IEEE Transactions on Image Processing*, vol. 27, no. 1, pp. 10–23, 2018.

[55] S. Yang, Y. Wang, Y. Shi, and Z. Fei, "Can categories and attributes be learned in a multi-task way?," *IEEE Transactions on Multimedia*, vol. 21, no. 12, pp. 3194–3204, 2019.

[56] C. Huang, H. Li, Y. Xie, Q. Wu, and B. Luo, "Pbc: polygon-based classifier for fine-grained categorization," *IEEE Transactions on Multimedia*, vol. 19, no. 4, pp. 673–684, 2017.

[57] J. Chen, J. Hu, and S. Li, "Learning to locate for fine-grained image recognition," *Computer Vision and Image Understanding*, vol. 206, p. 103184, 2021.

[58] Y. Dai, H. Zheng, Y. Gao et al., "Fused text segmentation networks for multi-oriented scene text detection," in *2018 24th International Conference on Pattern Recognition (ICPR)*, pp. 3604–3609, Beijing, China, August 2018.

[59] X. Wang, R. Zhang, T. Kong, L. Li, and C. Shen, "SOLOv2: dynamic, faster and stronger," 2020, https://arxiv.org/abs/2003.10152.