Research Article

Multimodal Multiobject Tracking by Fusing Deep Appearance Features and Motion Information

Liwei Zhang¹, **Jiahong Lai¹**, **Zenghui Zhang¹**, **Zhen Deng¹**, **Bingwei He¹**,
and **Yucheng He²**

¹*School of Mechanical Engineering and Automation, Fuzhou University, Fuzhou, China*

²*The T Stone Robotics Institute, Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong, China*

Correspondence should be addressed to Zhen Deng; zdeng@fzu.edu.cn

Received 6 August 2020; Revised 17 August 2020; Accepted 13 September 2020; Published 25 September 2020

Academic Editor: Guang Li

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

Multiobject Tracking (MOT) is one of the most important abilities of autonomous driving systems. However, most of the existing MOT methods only use a single sensor, such as a camera, which has the problem of insufficient reliability. In this paper, we propose a novel Multiobject Tracking method by fusing deep appearance features and motion information of objects. In this method, the locations of objects are first determined based on a 2D object detector and a 3D object detector. We use the Nonmaximum Suppression (NMS) algorithm to combine the detection results of the two detectors to ensure the detection accuracy in complex scenes. After that, we use Convolutional Neural Network (CNN) to learn the deep appearance features of objects and employ Kalman Filter to obtain the motion information of objects. Finally, the MOT task is achieved by associating the motion information and deep appearance features. A successful match indicates that the object was tracked successfully. A set of experiments on the KITTI Tracking Benchmark shows that the proposed MOT method can effectively perform the MOT task. The Multiobject Tracking Accuracy (MOTA) is up to 76.40% and the Multiobject Tracking Precision (MOTP) is up to 83.50%.

1. Introduction

The objective of Multiobject Tracking (MOT) is to track multiple objects at the same time and estimate their current states, such as locations, velocities, and sizes, while maintaining their motion identifications. Hence, the MOT is one of the most important abilities of autonomous systems, but it remains challenging because the target objects may be obscured, or it may be interfered by objects of similar shape. Owing to the rapid development of object detectors, several tracking-by-detection methods [1–5] have been widely proposed to address the MOT problem. Typically, the existing tracking-by-detection methods involve two main computational steps: object detection and tracking. These methods first detect the location of objects and then compute the trajectories of the objects based on the results of object detection [6–8]. The accuracy of object tracking is highly related to the performance of object detection. Hence,

the important thing about the MOT is to track the new targets that appear at any time and find lost tracking target objects from detections and associate again. However, most of the tracking-by-detection methods are based on vision-based object detections. In the case of occlusion and overexposure, vision-based object detection may lead to false association with existing trajectories. For example, Figure 1(a) shows the failure of vehicle detection on the image with the occlusion of humans. Figure 1(b) shows the camera is disabled when overexposure.

The scene of autonomous driving may contain multiple objects, and the states of the objects are usually uncertain [9, 10]. In this case, the vision-based object detections are susceptible to occlusion or overexposure, which will easily lead to false checks or loss of target tracking. Besides, one major challenge of the MOT is how to reduce incorrect identity switching. Because the tracked objects often have high similarities, it is

FIGURE 1: Examples of object detection in the case of occlusion and overexposure.

challenging to track objects correctly and perform correct Re-Identification(RE-ID).

Multimodal data fusion has the potential to improve the stability and accuracy of the MOT. However, a majority of traditional methods use the camera, LiDAR, or radar. These methods need to design hand-crafted features [11]. However, the hand-crafted features are often not of high precision, and it is difficult to guarantee the tracking performance. Hence, it is necessary to design a feature learning method that can automatically learn appearance features from raw visual data. Moreover, in autonomous driving systems, since the objects are moving rather than stationary, the motion information of objects should be integrated with the appearance features to achieve the MOT tasks. In addition, some MOT methods include depth information in the tracking process by using depth camera in order to improve tracking performance. For example, Mehner et al. [12] used an ordinary camera to obtain 2D information of objects and used a depth camera to obtain depth information to assist in locating the objects in world coordinates. Although it can improve the accuracy, the depth camera has a small field of view, high noise, and is easily affected by sunlight, so it is not effective as LiDAR. Moreover, they only use Kalman Filter for tracking, which does not work well in complex scenarios.

In this paper, we propose a multimodal MOT method by fusing the motion information and the deep appearance features of objects. This paper employs a 2D object detector, i.e., You Only Look Once (YOLOv3) [3] and a 3D object detector, i.e., PointRCNN [5] to process the RGB image and laser point cloud, respectively. The combination of 2D detection and 3D detection is helpful to improve the robustness of object detection. Then, the MOT is achieved by associating the motion information and the deep appearance features of the target object. A set of experiments on the KITTI Tracking Benchmark is performed to demonstrate the effectiveness of the proposed MOT method. Our contributions are summarized as follows:

- (1) The 2D object detection based on the image and the 3D object detection-based laser point cloud are combined to detect the location of objects, which is robust against light changes and occlusion.
- (2) We apply CNN that is pretrained to discriminate vehicles on a large-scale vehicle Re-Identification dataset to automatically extract the deep appearance features of the target object without manually designing features.

- (3) A multimodal MOT method is proposed by fusing the motion information and deep appearance features of the object to achieve the MOT task. In addition, the proposed method obtains competitive qualitative and quantitative tracking results on the KITTI tracking benchmark.

The rest of the paper is organized as follows. Section 2 introduces related works. Section 3 presents the proposed multimodal MOT method. Experiments and their results are presented in Section 4. Finally, the conclusion and future work are summarized in Section 5.

2. Related Works

This section provides an overview of the two related research topics: multiobject tracking and object detection.

2.1. Multiobject Tracking. The problem of the MOT first appeared in the tracking of object trajectory. For example, tracking of multiple enemy aircraft or passing missiles. With the development of computer vision, researchers have proposed several MOT methods from different aspects in the past few decades. For example, the single-object tracking method is extended to support multiple objects. According to the data association, the existing MOT methods can be divided into two categories: offline and online MOT methods. In offline methods [13–16], the detection of all frames in the sequence is combined to obtain the object trajectory robustly. These methods need to construct a global graph structure, which leads to high computational complexity. However, in the online MOT method [17–20], the target detector is only associated with the existing trajectories frame by frame. Hence, online methods are more suitable for real-time tracking.

Most of the existing MOT methods rely on motion information produced from Kalman Filter [21], Hungarian algorithm with Kalman Filter [17], Particle Filter [22], or probability hypothesis density filter [23]. However, in autonomous driving systems, due to the uncertainty of the scene, it is impossible to track objects stably only by using motion information. Therefore, more recent methods combine the motion features with the appearance features to improve the re-identification of target objects. Traditionally, the appearance features of objects are manually designed [24], which cannot provide reliable features, especially, in complex scenes. Owing to the rapid development of deep learning, deep convolutional networks [9, 25, 26] have been widely used to extract the appearance features from raw

visual data. For example, Wojke et al. [17] used CNN to extract the pedestrian image features and measure the distance between features for human detection.

2.2. Object Detection. Most of the existing 2D object detection methods are based on CNNs, which can be divided into two-stage detectors and one-stage detectors. In the two-stage detectors, such as RCNN [27], Fast RCNN [28], Faster RCNN [1], and FPN [29], they use Region Proposal Networks (RPN) to generate the candidate regions and then perform bounding-box classification and regression. For example, RCNN starts with the extraction of a set of object proposals by the selective search. Then, each proposal is rescaled to a fixed size image and fed into a CNN model that is trained on ImageNet. In this way, the presence of an object within each region is predicted and its category is recognized. Although the two-stage detectors have made great progress, their main drawback is that the redundant feature calculation of a large number of overlapping schemes results in a very slow detection speed.

The One-stage detectors have YOLO [3, 30, 31], Single Shot MultiBox Detector (SSD) [2], and RetinaNet [32]. These detectors do not need the RPN. They directly generate the categories' probability and bounding boxes of the objects. These methods only use one-stage calculation to get the final detection results. For example, the YOLO applies a single neural network to the whole image. This network divides the images into regions and predicts the bounding boxes and the probabilities for each region simultaneously. Compared with the two-stage detectors, the one-stage detectors have a higher detection speed.

Because the point-cloud data contains richer geometric features, 3D object detection has attracted more and more attention. Compared with 2D object detection, 3D object detection is more challenging because it needs to process the point clouds of the scene. Chen et al. [33] projected point cloud to the bird's view and used 2D CNNs to learn the features of point cloud for 3D boxes' generation. Song and Xiao [34, 35] divided the point cloud into equally spaced 3D voxels and used 3D CNNs to learn the features of voxels to generate 3D boxes. Shi et al. [36] used PointNet++ [37] to process the point-cloud inputs for 3D boxes' generation. Besides, some methods [38, 39] estimate 3D bounding boxes based on images.

3. Method

This section introduces the proposed multimodal MOT method that tracks multiple objects at the same time and records their trajectories. The proposed MOT method includes the four main computations: object detection with Non-maximum Suppression, motion information extraction, learning deep appearance feature, and object tracking with data association. Figure 2 shows an overview of the proposed MOT method. We combine the result of 2D object detection and 3D object detection such that the location of the object can be detected robustly. Based on this, the motion information and appearance features of objects are computed respectively. Finally, the motion information and appearance features of objects are associated to track the target object.

3.1. Object Detection with NMS. The first task of the MOT is to detect the location of objects in the scene. In this paper, we propose to combine the results of 2D object detection and 3D object detection for robust object detection. We use the 2D detector, i.e., YOLOV3 [3] that is trained on the training set of the KITTI 2D object detection benchmark and uses the 3D detector, i.e., PointRCNN [5], that is trained on the training set of the KITTI 3D object detection benchmark.

The 2D detector processes the RGB image. The output of 2D object detection is a set of detections $D_{2d}^t = \{D_{2d}^1, D_{2d}^2, \dots, D_{2d}^{n_1}\}$, where n_1 is the number of objects at frame t . The 3D detector processes the point clouds that were collected from a LiDAR. The output of 3D object detection is $D_{3d}^t = \{D_{3d}^1, D_{3d}^2, \dots, D_{3d}^{n_2}\}$, where n_2 is the number of objects at frame t . For further calculation, we project the LiDAR point in the 3D space into the 2D space according to combine camera and LiDAR calibration:

$$y = P_{\text{rect}} * R_{\text{rect}} * T_{\text{proj}} * x, \quad (1)$$

where y is the projected point in the RGB image. x denotes the 3D LiDAR point. P_{rect} and R_{rect} are the intrinsic camera parameters. The P_{rect} is the camera matrix, and the R_{rect} is the rectification matrix to make the image co-planar. T_{proj} projects the point X in the LiDAR coordinates onto the camera coordinate system. Both the intrinsic and extrinsic parameters are available in the KITTI dataset [40]. Figure 3 shows an example of point projections.

After the 3D point clouds are projected onto the image, two overlapping boxes will appear on the same object. This paper further uses the Nonmaximum Suppression (NMS) algorithm to get rid of the extra boxes. The NMS sorts all detection boxes on the basis of their scores and selects box M with the highest score. All other detection boxes with the large overlapping area with M are suppressed by using a predefined threshold N_t :

$$S_i = \begin{cases} S_i, & \text{IOU}(M, b_i) < N_t, \\ 0, & \text{IOU}(M, b_i) \geq N_t, \end{cases} \quad (2)$$

$$\text{with } \text{IOU}(M, b_i) = \frac{\text{area}(M) \cap \text{area}(b_i)}{\text{area}(M) \cup \text{area}(b_i)},$$

where b_i is the detection box to be screened, when $\text{IOU}(M, b_i)$ is greater than N_t , b_i will be removed. In our experiment, N_t is set to 0.7. Figure 4 shows a comparison result by the detection method without NMS and with NMS.

3.2. Learning Object Appearance Features. Before implementing the MOT, we need to extract the appearance features of the object. This paper employs CNN to automatically learn the deep appearance features of objects from raw visual data. The CNN is trained on a large-scale benchmark dataset [41]. The dataset contains over 50,000 images of 776 vehicles captured by 20 cameras. Figure 5 shows several samples in this dataset.

Table 1 illustrates the architecture of CNN used in this paper. The CNN model is inspired by the wide residual

FIGURE 2: An overview of the proposed multimodal MOT method.

FIGURE 3: Result of point projections, where the 3D LiDAR points are projected onto the RGB image.

network [40, 42] that consists of two convolution layers and six residual blocks. The Dense layer 10 extracts a 128-dimensional global feature. The final batch and l_2 -norm layer projects feature to a unit hypersphere. By resizing the tracked vehicle image to 224×224 , then inputting it into the network. Finally, we get a 128-dimensional feature vector that is used as the deep appearance features of the object.

3.3. Extraction of Object Motion Information. Since the objects are usually moving rather than stationary, it is necessary to extract the motion information of objects for the MOT. This paper employs the Kalman filter to predict the state of the object and then extract its motion information. We use eight parameters $x_k = (a, b, \gamma, h, \dot{a}, \dot{b}, \dot{\gamma}, \dot{h})$ to describe the tracking state at frame k , where (a, b) is the bounding box center position, γ is the aspect ratio, h is the height of the bounding box, and $(\dot{a}, \dot{b}, \dot{\gamma}, \dot{h})$ represents the corresponding velocity in the image coordinate system.

Because the interval of time between each frame is very short, it can be regarded as a linear model of constant-velocity motion. We get the predicted object state at the next frame and calculate the error covariance matrix P_k^- between the predicted state and the true state:

$$\begin{aligned} x_k^- &= Ax_{k-1}, \\ P_k^- &= AP_{k-1}A^T + Q, \end{aligned} \quad (3)$$

where x_k^- is the predicted object state at frame k . A is a state transition matrix, and x_{k-1} is the object state at frame $k-1$. And Q is the covariance matrix of the predict noise. Then, we can get the Kalman gain matrix K and calculate the estimated state x_k :

$$\begin{aligned} K_k &= P_k^- H^T (H P_k^- H^T + R)^{-1}, \\ x_k &= x_k^- + K_k (z_k - H x_k^-), \end{aligned} \quad (4)$$

where z_k is the measured value and H is the conversion matrix from x_k^- to z_k . R is the covariance matrix of the measurement noise. Finally, update the covariance matrix P_k :

$$P_k = (I - K_k H) P_k^-. \quad (5)$$

3.4. Object Tracking Based on Data Association. The next is to associate the deep appearance features and the motion information of the object for the MOT. First, this paper uses the Mahalanobis distance to compare the motion correlation between the predicted state of the Kalman Filter and the newly detected bounding boxes:

$$d^{(1)}(i, j) = (d_j - y_i)^T S_i^{-1} (d_j - y_i), \quad (6)$$

where d_j denotes the j th bounding box detection, y_i and S_i represent the mean and covariance of the i th predicted bounding box. A threshold can be adjusted to control the minimum confidence of the motion information association between objects i and j . We denote this decision with an indicator $b_{i,j}^{(1)}$, as shown in equation (7). The indicator will be equal to 1 if the Mahalanobis distance is smaller or equal to a threshold $t^{(1)}$, which is set to 9.4877 for our four-dimensional measurement space:

$$b_{i,j}^{(1)} = \begin{cases} 1, & d^{(1)}(i, j) \leq t^{(1)}, \\ 0, & d^{(1)}(i, j) > t^{(1)}. \end{cases} \quad (7)$$

Next, the above method is only a suitable related measurement index when motion uncertainty is very low. However, in the image space, only using the Kalman filter framework is a rough prediction. Therefore, this paper also adopted the second metric. It measures the smallest cosine distance of the appearance features between the i th track and j th detection as follows:

$$d^{(2)}(i, j) = \min(1 - r_j^T r_k^{(i)}), \quad (8)$$

FIGURE 4: The comparison results by the detection method without NMS (a) and with NMS (b).

FIGURE 5: Some samples in the VERI dataset. From top to bottom are cars, trucks, and buses.

TABLE 1: The architecture of the used CNN.

Name	Patch size/stride	Output size
Conv 1	$3 \times 3/1$	$32 \times 128 \times 64$
Conv 2	$3 \times 3/1$	$32 \times 128 \times 64$
Max pool 3	$3 \times 3/2$	$32 \times 64 \times 32$
Residual 4	$3 \times 3/1$	$32 \times 64 \times 32$
Residual 5	$3 \times 3/1$	$32 \times 64 \times 32$
Residual 6	$3 \times 3/2$	$64 \times 32 \times 16$
Residual 7	$3 \times 3/1$	$64 \times 32 \times 16$
Residual 8	$3 \times 3/2$	$128 \times 16 \times 8$
Residual 9	$3 \times 3/1$	$128 \times 16 \times 8$
Dense 10	—	128
Batch and l_2 normalization	—	128

where r_j is the appearance feature vector of detection d_j and $r_k^{(i)}$ represents the feature vector of the i th tracked object at the most recent frame k . In our experiment, parameter k is set to a maximum number of 100 available vectors. In addition, in order to determine whether the appearance features are related, we introduce a binary indicator, as shown in equation (9). A threshold $t^{(2)}$ is set for this indicator on a VeRi dataset:

$$b_{i,j}^{(2)} = \begin{cases} 1, & d^{(2)}(i, j) \leq t^{(2)}, \\ 0, & d^{(2)}(i, j) > t^{(2)}. \end{cases} \quad (9)$$

Then, the Mahalanobis distance determines whether the prediction position of the Kalman filter is related to the new detection, which is especially useful for short-term prediction. And the cosine distance considers the appearance of tracking objects, which is especially useful for recovering identity after a long period of occlusion. Therefore, this paper combines the two metrics using a weighted sum:

$$d_{i,j} = \lambda d^{(1)}(i, j) + (1 - \lambda) d^{(2)}(i, j), \quad (10)$$

$$\text{with } b_{i,j} = \prod_{m=1}^2 b_{i,j}^{(m)},$$

where we call an association admissible if $b_{i,j}^{(1)} = 1$ and $b_{i,j}^{(2)} = 1$. The hyperparameter λ is used to control the influence of each metric on the combined association. For example, when there is substantial object motion, the prediction of the constant-velocity motion model becomes less effective. Thus, the appearance metric becomes more significant by reducing the value of λ ; on the contrary, when there are limited vehicles on the road without long-term partial occlusions, increasing the value of λ can improve the importance of distance metric.

Finally, in our implementation, the maximum number of frames allowed to lose the target $A_{\max}$ is considered. In order to avoid redundant computations, if a tracked object is not re-identified in the most recent $A_{\max}$ frames passed since

its last instantiation, it will be assumed that it has left the scene. If the object is seen again, a new ID will be assigned to it. The judgement of a new track is that an object in the result of detection can never be associated with the existing MOT methods. If the prediction of the object position can be correctly correlated with the detection in the consecutive $F_{\min}$ frames, we can confirm that a new track target has appeared.

4. Experiment

This section introduces the dataset, evaluation metric, training parameters, and experimental evaluation results in the experiments on the KITTI Tracking Benchmark.

4.1. Dataset. The proposed method was evaluated on the KITTI tracking benchmark [43]. The KITTI dataset was collected under 4 different scenarios, including city, residential, road, and campus. Some samples of the KITTI dataset are shown in Figure 6. The dataset consists of 21 training sequences and 29 test sequences. In each sequence, LIDAR point clouds, RGB images, and calibration files were provided. In the training sequences, eight different classes were labeled, including car, pedestrian, and cyclist. The objects in images were annotated with 3D and 2D bounding boxes between different frames and had a unique ID. In this work, we used all 29 testing sequences for modal validation and only used on the car subset for model evaluation because it had the most instances of all object types.

4.2. Evaluation Metric. The indexes used to evaluate the performance of the proposed MOT method were as follows:

- (1) Mostly Tracked (MT): objects are successfully tracked to at least 80% of their trajectories during their life span.
- (2) Mostly Lost (ML): objects are successfully tracked to less than 20% of their trajectories during their life span.
- (3) Identity Switches (IDS): the number of times objects' identities have changed during their life span.
- (4) Fragmentation (Frag): due to the missing detection, the number of times a trajectory is interrupted.
- (5) FP and FN: the total number of false positives and false negatives (missed targets).
- (6) Multiobject Tracking Accuracy (MOTA): it combines three error sources, i.e., FP, FN, and IDS as follows [44]. Equation (11) shows the computation of the MOTA, where t is the index of the frame and G is the number of the ground truth:

$$\text{MOTA} = 1 - \frac{\sum_t (\text{FP}_t + \text{FN}_t + \text{IDS}_t)}{\sum_t G_t}. \quad (11)$$

- (7) Multiobject Tracking Precision (MOTP): the alignment accuracy between the annotated and the predicted bounding boxes [44].

4.3. Training Parameters. This paper trained the 2D detector, i.e., the YOLOv3, on the training set of the KITTI 2D object detection benchmark [5], and trained the 3D detector, i.e., the PointRCNN, on the training set of the KITTI 3D object detection benchmark [36]. The IOU threshold N_t of the NMS module was set to 0.7. The minimum number of matched frames required to create a new trajectory $F_{\min}$ is set to 3 and the maximum number of frames allowed to lose the target $A_{\max} = 30$. And because the prediction results of Kalman Filter is rough and there are many scenes with long-term partial occlusions in the KITTI dataset, we set $\lambda = 0.1$.

4.4. Qualitative Evaluation. We evaluated the proposed tracking method qualitatively by using the KITTI test sequence. Different scenarios including occlusions, clutter, parked vehicles, and false positives from detectors were considered in the qualitative evaluation.

Figure 7 shows an example of the test sequence 0 in the test set. Each vehicle was assigned a tracking ID as a reference. Despite the compact and messy parking of the vehicle, the proposed MOT method can continuously detect and track the vehicles. Moreover, from this figure, we can see that, since the image is easily affected by the environment, such as illumination changes and partial occlusion, the shape of the detected target will change. In addition, the scale of the target object may be very different. In this case, the proposed MOT method still obtained a relatively high tracking performance. The experimental results show that our method can locate each car well even in the cluttered and strong lighting scene and maintain the ID of the car unchanged.

Figure 8 shows another example from the test sequence 1. Figure 8(a) shows that the object detector produces a false detection result, and Figure 8(b) shows the false positive of the detector is overcome by data association. In the case of transient errors in object detection, the proposed MOT method can still track the target stably. Hence, these experimental results demonstrated the robustness of the proposed MOT method.

4.5. Benchmark Results. We further evaluated the proposed MOT method on the KITTI Tracking Benchmark. In this evaluation, we considered some published online MOT methods for comparison. The results are presented in Table 2. It can be seen that the proposed MOT method is very competitive. In particular, the proposed MOT method returns the fewest number of identity switches, while maintaining competitive MOTA scores, MOTP scores, and track fragmentations. The tracking accuracy is mainly affected by a large number of false positives. Given their overall impact on the MOTA score, the combination of the 2D and 3D object detection results can significantly improve the performance of the MOT. Besides, because we set the maximum allowed trackage and associate the object motion information and appearance features, the proposed MOT method has the fewest number of identity switches. Therefore, the proposed MOT method can generate a relatively stable trajectory of the target object.

FIGURE 6: Some samples in the KITTI dataset. (a) City. (b) Residential. (c) Road. (d) Campus.

FIGURE 7: Result of the qualitative evaluation with the test sequence 0. Each vehicle is assigned a tracking ID as a reference.

FIGURE 8: Result of the qualitative evaluation with the test sequence 1. (a) The detector generates an error detection and disappears in the next frame. (b) The proposed MOT method can track objects stably.

4.6. Ablation Study. The ablation study was to evaluate the effects of hyperparameters on the performance of the proposed MOT method. Table 3 shows the results of the ablation study on the KITTI benchmark. The hyperparameter N_t is the threshold of IOU, and the $F_{\min}$ denotes the minimum number of matching frames required to create a new trajectory. From the table, we can be seen that when $N_t = 0.6$, this may miss some

correct detection results. That is because the number of detected objects is reduced. When $N_t = 0.8$, this may result in some wrong detection results, which is also the reason why it has the most IDS. $F_{\min} = 1$ means that track immediately when a new target is detected, which leads to more IDS and FRAG. The $F_{\min} = 5$ makes the minimum IDS, but MOTA is lower. Therefore, we finally set $N_t = 0.7$ and $F_{\min} = 3$.

TABLE 2: Comparison on KITTI tracking benchmark.

Method	MOTA (%)↑	MOTP (%)↑	MT (%)↑	ML (%)↓	IDS↓	FRAG↓
SASN-MCF nano [45]	70.86	82.65	58.00	7.85	443	975
SSP [46]	72.72	78.55	53.85	8.00	185	932
CIWT [12]	75.39	79.25	49.85	10.31	165	660
Complexer-YOLO [47]	75.70	78.46	58.00	5.08	1186	2092
DSM [48]	76.15	83.42	60.00	8.31	296	868
Ours	76.40	83.50	47.38	14.00	147	608

TABLE 3: Ablation study on the training set of KITTI tracking benchmark.

Parameter	MOTA (%)↑	MOTP (%)↑	MT (%)↑	ML (%)↓	IDS↓	FRAG↓
$N_t = 0.6$	71.27	82.92	43.73	8.42	59	409
$N_t = 0.8$	70.09	83.11	49.64	6.81	289	559
$F_{\min} = 1$	72.05	82.70	46.01	5.02	161	548
$F_{\min} = 5$	68.13	83.30	32.97	13.26	35	229
Ours	72.36	83.94	47.31	7.36	94	395

5. Conclusion

This paper proposed a multimodal MOT method by fusing the motion information and the deep appearance feature of objects. In this method, we use a Nonmaximum Suppression algorithm to combine a 2D object detector and a 3D object detector for robust object detection. Then, the deep appearance features of objects are learned by a CNN, and the motion information of objects is computed by the Kalman Filter. The MOT task is achieved by associating the appearance features and the motion information of the target object. The effectiveness of the proposed MOT method was demonstrated in a set of experiments. The proposed MOT method can track objects stably in crowded scenes and effectively avoid false detection. In the KITTI tracking benchmark, the proposed method also shows competitive results.

Although 3D object detection is used in the proposed MOT method, it is only used as the auxiliary information for 2D object detection. 3D object detection can provide accurate position and size estimation for automatic driving. Therefore, our future work will be towards the direction of 3D multitarget tracking that can adapt to a more complex environment.

Data Availability

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

Conflicts of Interest

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

Acknowledgments

This work was supported by the National Natural Science Foundation of China (Project no. 61673115). This work was also partly funded by the German Science Foundation (DFG) and National Science Foundation of China (NSFC) in

project Cross Modal Learning under contract Sonderforschungsbereich Transregio 169.

References

- [1] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *Neural Information Processing Systems*, vol. 39, pp. 91–99, 2015.
- [2] W. Liu, "SSD: single Shot MultiBox detector," in *Proceedings of the European Conference on Computer Vision*, pp. 21–37, Amsterdam, Netherlands, October 2016.
- [3] R. Joseph and A. Farhadi, "Yolov3: an incremental improvement," 2018, <https://arxiv.org/pdf/1804.02767.pdf>.
- [4] C. Zhu, Y. He, and M. Savvides, "Feature selective anchor-free module for single-shot object detection," in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 840–849, Hubballi, India, December 2019.
- [5] D. Maturana and S. Scherer, "VoxNet: a 3D convolutional neural network for real-time object recognition," in *Proceedings of the Intelligent Robots and Systems*, pp. 922–928, Hamburg, Germany, September 2015.
- [6] S. Sharma, J. A. Ansari, J. K. Murthy, and K. M. Krishna, "Beyond pixels: leveraging geometry and shape cues for online multi-object tracking," in *Proceedings of the International Conference on Robotics and Automation*, pp. 3508–3515, Brisbane, Australia, May 2018.
- [7] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier et al., "Online multiperson tracking-by-detection from a single, uncalibrated camera," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 9, pp. 1820–1833, 2011.
- [8] H. Zhou, W. Ouyang, J. Cheng, X. Wang et al., "Deep continuous conditional random fields with asymmetric inter-object constraints for online multi-object tracking," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 4, pp. 1011–1022, 2019.
- [9] W. Luo, B. Yang, and R. Urtasun, "Fast and furious: real time end-to-end 3D detection, tracking and motion forecasting with a single convolutional net," in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 3569–3577, Salt Lake City, UT, USA, June 2018.
- [10] N. Smolyanskiy, A. Kamenev, and S. Birchfield, "On the importance of stereo for accurate depth estimation: an

- efficient semi-supervised deep neural network approach,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 1007–1015, Salt Lake City, UT, USA, June 2018.
- [11] A. Asvadi, P. Girao, P. Peixoto, and U. Nunes, “3D object tracking using RGB and LIDAR data,” in *Proceedings of the IEEE International Conference on Intelligent Transportation Systems*, Rio de Janeiro, Brazil, November 2016.
 - [12] A. Osep, W. Mehner, M. Mathias, and B. Leibe, “Combined image- and world-space tracking in traffic scenes,” in *Proceedings of the International Conference on Robotics and Automation*, pp. 1988–1995, Singapore, June 2017.
 - [13] E. Levinkov, J. Uhrig, S. Tang, M. Omran et al., “Joint graph decomposition & node labeling: problem, algorithms, applications – supplement –,” in *Proceedings of the IEEE Conf. Comput. Vis. Pattern Recognit*, pp. 1904–1912, Honolulu, HI, USA, July 2017.
 - [14] M. Keuper, S. Tang, B. Andres, T. Brox et al., “Motion segmentation & multiple object tracking by correlation co-clustering,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 1, pp. 140–153, 2020.
 - [15] B. Wang, G. Wang, K. L. Chan, and L. Wang, “Tracklet association with online target-specific metric learning,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 1234–1241, Columbus, OH, USA, June 2014.
 - [16] C. Wang, H. Liu, and Y. Gao, “Scene-adaptive hierarchical data association for multiple objects tracking,” *IEEE Signal Processing Letters*, vol. 21, no. 6, pp. 697–701, 2014.
 - [17] N. Wojke, A. Bewley, and D. Paulus, “Simple online and realtime tracking with a deep association metric,” in *Proceedings of the International Conference on Image Processing*, pp. 3645–3649, Beijing, China, September 2017.
 - [18] L. Zhang and L. V. Der Maaten, “Structure preserving object tracking,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 1838–1845, Cancun, Mexico, December 2013.
 - [19] S. H. Bae and K. J. Yoon, “Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning,” in *Proceedings of the Computer Vision and Pattern Recognition (CVPR)*, Columbus, OH, USA, June 2014.
 - [20] F. Poiesi, R. Mazzon, A. Cavallaro, and I. U. Cviu, “Multi-target tracking on confidence maps: an application to people tracking,” *Computer Vision and Image Understanding*, vol. 117, no. 10, pp. 1257–1272, 2013.
 - [21] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, “Simple online and realtime tracking,” in *Proceedings of the International Conference on Image Processing*, pp. 3464–3468, Las Vegas, NV, USA, January 2016.
 - [22] B. Liu, S. Cheng, and Y. Shi, “Particle filter optimization: a brief introduction,” in *Proceedings of the International Conference on Swarm Intelligence*, pp. 95–104, Springer, Cham, Switzerland, 2016.
 - [23] S. A. Goli, B. H. Far, and A. O. Fapojuwo, “An accurate multi-sensor multi-target localization method for cooperating vehicles,” in *Information Reuse and Integration*, pp. 197–217, Springer, 2016.
 - [24] B. D. Lucas and T. Kanade, “An iterative image registration technique with an application to stereo vision,” in *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 674–679, Vancouver, Canada, August 1981.
 - [25] L. Chen, H. Ai, R. Chen, and Z. Zhuang, “Aggregate tracklet appearance features for multi-object tracking,” *IEEE Signal Processing Letters*, vol. 26, no. 11, pp. 1613–1617, 2019.
 - [26] X. Su, X. Qu, Z. Zou, P. Zhou et al., “K-reciprocal harmonious attention network for video-based person Re-identification,” *IEEE Access*, vol. 7, pp. 22457–22470, 2019.
 - [27] R. Girshick, J. Donahue, T. Darrell, and J. Malik, “Rich feature hierarchies for accurate object detection and semantic segmentation,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 580–587, Columbus, OH, USA, June 2014.
 - [28] R. Girshick, “Fast R-CNN,” in *Proceedings of the International Conference on Computer Vision*, pp. 1440–1448, Santiago, Chile, December 2015.
 - [29] T. Lin, P. Dollar, R. Girshick, K. He, B. Hariharan, and S. Belongie, “Feature pyramid networks for object detection,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 936–944, Honolulu, HI, USA, July 2017.
 - [30] J. Redmon, S. K. Divvala, R. Girshick, and A. Farhadi, “You only Look once: unified, real-time object detection,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 779–788, Las Vegas, NV, USA, June 2016.
 - [31] J. Redmon and A. Farhadi, “YOLO9000: better, faster, stronger,” in *Proceedings of the computer vision and pattern recognition*, pp. 6517–6525, Honolulu, HI, USA, July 2017.
 - [32] T. Lin, P. Goyal, R. Girshick, K. He, and P. Dollar, “Focal loss for dense object detection,” in *Proceedings of the International Conference on Computer Vision*, pp. 2999–3007, Venice, Italy, July, 2017.
 - [33] X. Chen, H. Ma, J. Wan, B. Li, and T. Xia, “Multi-view 3D object detection network for autonomous driving,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 6526–6534, Honolulu, HI, USA, July 2017.
 - [34] S. Song and J. Xiao, “Deep sliding shapes for amodal 3D object detection in RGB-D images,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 808–816, Las Vegas, NV, USA, June 2016.
 - [35] Y. Zhou and O. Tuzel, “VoxelNet: end-to-end learning for point cloud based 3D object detection,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 4490–4499, Salt Lake City, UT, USA, June 2018.
 - [36] S. Shi, X. Wang, and H. Li, “PointRCNN: 3D object proposal generation and detection from point cloud,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 770–779, Long Beach, CA, USA, June 2019.
 - [37] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, “PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space,” in *Proceedings of the NIPS*, Long Beach, CA, USA, December 2017.
 - [38] J. Ku, A. D. Pon, and S. L. Waslander, “Monocular 3D object detection leveraging accurate proposals and shape reconstruction,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 11867–11876, Long Beach, CA, USA, June 2019.
 - [39] Z. Deng and L. J. Latecki, “Amodal detection of 3D objects: inferring 3D bounding boxes from 2D ones in RGB-depth images,” in *Proceedings of the Computer Vision and Pattern Recognition*, pp. 398–406, Honolulu, HI, USA, July 2017.
 - [40] S. Zagoruyko, N. J. Komodakis, and P. Recognition, “Wide residual networks,” in *Proceedings of the BMVC*, New York, NY, USA, September 2016.
 - [41] X. Liu, W. Liu, T. Mei, and H. Ma, “PROVID: progressive and multimodal vehicle reidentification for large-scale urban surveillance,” *IEEE Transactions on Multimedia*, vol. 20, no. 3, pp. 645–658, 2018.
 - [42] N. Wojke and A. Bewley, “Deep cosine metric learning for person Re-identification,” in *Proceedings of the Workshop on Applications of Computer Vision*, pp. 748–756, Vancouver, Canada, September 2018.

- [43] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Providence, RI, USA, June 2012.
- [44] K. Bernardin, R. Stiefelhagen, and V. Processing, "Evaluating multiple object tracking performance: the CLEAR MOT metrics," *EURASIP Journal on Image and Video Processing*, vol. 2008, 10 pages, 2008.
- [45] G. Gunduz and TAJITOI Vehicles, *Efficient Multi-Object Tracking by Strong Associations on Temporal Window*, IEEE, Newyork, NY, USA, 2019.
- [46] P. Lenz, A. Geiger, and R. Urtasun, "FollowMe: efficient online min-cost flow tracking with bounded memory and computation," in *Proceedings of the International Conference on Computer Vision*, pp. 4364–4372, Santiago, Chile, December 2015.
- [47] M. Simon, "Complexer YOLO: real-time 3D object detection and tracking on semantic point clouds," in *Proceedings of the Computer Vision and Pattern Recognition*, p. 0, Long Beach, CA, USA, June 2019.
- [48] D. Frossard and R. Urtasun, "End-to-end learning of multi-sensor 3D tracking by detection," in *Proceedings of the International Conference on Robotics and Automation*, pp. 635–642, Brisbane, Australia, May 2018.