CNN-Based Multi-Object Tracking Networks with Position Correction and IMM in Intelligent Transportation System

Chengjuan Ren[1], Guangnan Zhang[2,*], Dongwon Jeong[3], Lele Hao[4]

1, Guangdong Atv Academy For Performing Arts, Guangdong, China
2, Department of Computer Science, Baoji University of Arts and Sciences, Baoji, China
3, Software Convergence Engineering Department, Kunsan National University, Gunsan, Korea
*: Correspondence
Email: renchengjuan163@163.com

Abstract In multi-object tracking of Intelligent Transportation System, there are objects of different sizes in images or videos, especially pedestrian and traffic lights with low resolution in the image. Meantime, objects are subject to occlusion or loss in object tracking. All of the above-mentioned situations may lead to unsatisfactory multi-object tracking results. Attracted by the effect of deep convolution neural networks, the paper proposes a multi-object tracking network, CNN-Based Multi-Object Tracking Networks with Position Correction and IMM (CNN_PC_IMM) to solve those problems. Our proposed method consists of object detection module and object tracking module. Compared to other networks, our proposed network has several main contributions that play an essential role in achieving state-of-the-art object tracking performance. In the detection phase, the feature fusion technique is used. We add a scale branch to the YOLOv3 network to increase the accuracy of small object prediction and import a residual structure to enhance gradient propagation and avoid gradient disappearance and explosion for the whole network. In addition, we determine the size of the anchor box based on the size of the object in the dataset to better detect and track the objects. In the tracking phase, IMM is used to calculate the motion state information of the object at a certain moment. Next, the optimization algorithm is proposed to fine-tune object position when the tracking object is occluded due to dense multi-object in traffic scenes or lost due to incomplete object information. Finally, experimental results and analysis are performed on the MOT16 benchmark dataset with several popular tracking algorithms used to compare the performance with the proposed algorithm in the paper. It is demonstrated that the proposed network has better performance on MOPA, MOTP, ML.

Keywords Intelligent Transportation System; multi-object tracking; deep convolution neural networks; YOLOv3; IMM

1. Introduction

With the further improvement of people's material and cultural levels in modern society, vehicles have gradually developed into a consumer product needed by the public in daily life. However, along with the yearly increase in the total number of vehicles come a series of serious traffic safety and traffic congestion problems in cities. For the prospective planning and development of each city, to reduce the occurrence of traffic accidents and traffic congestion, experts and scholars in industry and academia have invested a lot of human and material resources in the exploration of Intelligent Transportation Systems.

Multiple Object Tracking (MOT) also known as Multi-Object Tracking (MTT) is a computer vision task that seeks to optimize the analysis of video to identify and track objects belonging to more than one category [1, 2]. Multi-object tracking is an important part of the unmanned system perception algorithm. It not only provides data information of multiple spatio-temporal points and positions for multi-object motion objects, which gets the specific running state trajectory of each object. It also supplies highly informative data for object scene understanding. For example, the object motion contains information such as the acceleration and time of the motion of the corresponding object, the start-stop, and duration information of the acceleration in

the trajectory which indicates when the object has entered or left the scene. The implicit information of the trajectory can also indirectly express the behavioral motility and behavioral psychological characteristics of each object, which can offer significant data for high-level computerized motion visual recognition such as behavioral feature analysis and behavioral feature prediction. In addition, the rich message can be obtained by analyzing the motion state of each object, such as acquiring the number of objects that exist in a specific area over a while, the relationship between each object, etc. Therefore, multi-object tracking technology has important research and application value in Intelligent Transportation System. Traditional multi-object tracking involves three main elements, appearance model, motion model, and online update mechanism [3]. The implementation stage of traditional multi-object tracking includes constructing the initial appearance model, in which the initial area is obtained by object detection methods, etc.. And then, the corresponding features and the model are obtained by using the motion model to predict and analyze the area where the object is likely to appear. Third, the candidate area is determined and matched. Finally, the appearance model is adjusted based on the information from the previous parts to calculate the object area for the current frame. Traditional multi-object tracking methods usually use texture, shape, color, SIFT, SURF, and other features for object discrimination [4, 5]. Most of these features are designed for specific application scenarios, and their robustness is weak, resulting in a limited range of use.

In 2012, a milestone time of the deep learning era, a group of researchers from the University of Toronto achieved impressively with results in the ImageNet competition, overcoming all other teams in the first place by a significant margin[6]. Deep learning has subsequently been utilized in a broad scope of applications, speech recognition, automatic machine translation, self-driving cars, face recognition, etc. Multi-object tracking methods based on deep learning have also emerged in recent years. Usually, multi-object tracking algorithms are based on convolutional neural networks to detect objects and employ tracking models to accomplish multi-object tracking and achieve many excellent results. Yu et al. [7] developed a revised Faster R-CNN that contained skip-pooling and multi-region characteristics and fine-tuned it on various pedestrian detection datasets. They were able to enhance the performance of the proposed model by more than 30% with this structure, attaining state-of-the-art performance on the MOT16 dataset. Zhang et al.[8] interviewed SSD with Faster R-CNN and R-FCN in the context of their pig tracking and expressed that its performance is better on their data. They used an online tracking method named Discriminative Correlation Filters (DFC) with HOG and Colour Names features to detect the object, and the output of the DCF tracker was applied to refine the bounding boxes in case of tracking failure. Zhao et al.[9] then used SSD to replace pedestrians and vehicles in the scenario, but they made use of a CNN-based correlation filter to enable the SSD to create far more accuracy in the bounding boxes. Wang et al.[10] developed a new RGBT object tracking with short-term historical information in correlation filter tracking to solve the issue of RGBT and RGB tracking in a difficult environment by booting multimodal datasets. CNN and object bounding box were used to obtain object features in the whole framework which achieved results compared to state-of-the-art algorithms on three RGBT datasets. To improve the technique to get better performance of the bounding box once the objects encounter serious deformation, Xie et al. [11] combined the deepest layer feature of the CNN model and affine transformation as a new information model which was based on region CNN. RoI pooling and NMS were applied in the model and the proposed model has gained promising results.

The object scale changes a lot and the actual motion pattern is complicated, which makes it difficult for a single model to describe the motion pattern of the object. In this paper, we propose a multi-object tracking algorithm named CNN_PC_IMM that integrates the improved YOLOv3, named YOLOv3_I, and interactive multi-model, position correction optimization algorithm and can also automatically adjusts the model parameters according to the size of the target in the database. The model finally is tested and analyzed on the MOT16 dataset. The main contributions of this paper are: a new model for multi-scale detection of objects based on YOLOv3 is proposed and experimentally proven to be feasible for object detection. The detection results of the previous step are used as input for the subsequent tracking. And the motion state of the object is recorded with the interactive multi-model for object matching. A position correction optimization algorithm is proposed for the multi-object to detect the error rate by fine-tuning the position of the detection and prediction results when an object is lost or covered. The reliability of the proposed algorithm is verified and analyzed on the MOT16 and KITTI datasets in comparison with other algorithms.

The rest of the research is described as follows. Section 2 introduces the related work of multi-object tracking in Intelligent Transportation System. Then, Section 3 is dedicated to our approach to implementing multi-object tracking. Section 4 covers the comparison of the experimental results and analysis. Finally, the conclusion is wrapped up in Section 5.

2. Related work

Intelligent Transportation System (ITS) provides intelligent guidance for relieving traffic jams and reducing environmental pollution. The development of Intelligent Transportation System has been progressing rapidly. Meanwhile, Intelligent Transportation System has been encouraging much research in various fields such as vehicle

detection, congestion detection, vehicle counting, and multi-object tracking in recent years. Detection and tracking of traffic objects are an indispensable part of Intelligent Transportation System. The following gives the development of object tracking in Intelligent Transportation System with deep learning techniques or traditional methods。

2.1. Object tracking in ITS with deep learning techniques

Video-based car detection is considered as a component of Intelligent Transportation System, due to its accessibility to non-intrusive and data acquisition abilities of holistic car behavior. Inspired by Harris-Stephen corner detector, Chintalacheruvr et al. [12]designed a vehicle detection system that set the number and pace of vehicles on arterial roads and highways. This system has no complex calibration required, is robust against change, and gets greater performance on low-resolution video. In the field of ITS. Hinz et al.[13] proposed the first multi-object tracking model based on vision sensors for the neural vision system. Capabilities of the system were tested on real dynamic vision data of a highway bridge scenario. Liang et al. [14] employed the YOLO model and multi-object tracking algorithm to calculate the number of vehicles in the various traffic environment. In the paper, a real vehicle dataset was obtained from highway surveillance cameras. The experiment results expressed the proposed new method was feasible to be applied to real-life scenarios of vehicle computing. According to discuss the possibility of applying deep learning to low-resolution 3D LiDAR sensors, Pino et al. [15] devised a LIDAR-based system that performed point-to-point vehicle detection with PUCK data by CNN and MH-EKF to guess the real position and speed of the detected vehicle. The results showed that the claimed low-resolution DL algorithm could successfully perform the vehicle detection task with better performance than the geometric baseline approach. Furthermore, it was observed that the system realized tracking performance at the close range to that of the high-end HDL-64 sensor. On the other hand, at long distances, the detection was restrained to half the distance of the high-end sensors. Liu et al.[16] proposed the HSAN model to boot the ReID performance and obtained the robust features of the various objects. To distinguish objects, the attention mechanism method and the posed information were employed. The Market-1501, CUHK03 and DUKE ReID, and MOT were compared on HSAN. Abbas et al. [17]designed a V-ITS system to predict or track vehicles and driver's activities during highway driving. This modified V-ITS system enabled automated traffic regulation and thus reduced traffic accidents. To develop the V-ITS system, a pre-trained convolutional neural network model with 4 layers was used and the illegal behavior was identified with a deep belief network model. Vehicle counting has a vital role in Intelligent Transportation Systems as it assists in the creation of autonomous driving systems and better road planning. The author [18] suggested an effective object counting system and evaluated its capabilities with 20 different video datasets.

2.2. Multi-object tracking in ITS with traditional methods

Nizar et al. [19] used HOG to extract object features, SVM to classify objects, KLT tracker to compute the number of objects in order to detect traffic situations by computer vision. The developed system got 95.2% accuracy. Tian et al. [20] proposed a new tracking method to address the association of the object involving the presence of motion noise or extended occlusions by bringing together information from the expanded structural and spatial domains. In this approach, the detections are firstly combined into small traces depending on the meta-measurements of object proximity. The task of correlating small tracking is settled by structural information of the motion patterns based on their interactions. This work [21] is a broad overview of the MDP framework for MOT introduced by Xiang et al. with some additional crucial extensions. Firstly, the authors tracked objects with various cameras and sensor modalities by merging object candidate proposals. Secondly, the objects were tracked directly in the real world, which is different from the other methods. This allowed autonomous available to navigation and related tasks on tracking. Vasic et al. [22] used a collaborative fusion for extending the GM-PHD to track vehicles, which relied on the problems of clutter and occlusion. Emami et al. [23] presented a utility MOT framework that merged trajectories from a new video MOT neural architecture devised for low-power edge devices with trajectories from commercially accessible traffic radars. The proposed architecture implemented efficient spatio-temporal object re-identification by depth-separable convolution for joint predictive object detection and dense grid features at a single scale. Due to the complex interaction and representation of road participants (e.g., vehicles, bicycles, or pedestrians) and road context information (e.g., traffic lights, lanes, and regulatory elements), behavior prediction in multi-intelligent bodies and dynamic environments is essential in smart vehicle environments. Gómez–Huélamo et al.[24] described a novel SmartMOT, a powerful and simple model that introduced semantic information and the mind of tracking-by-detection to predict the next trajectories of the obstacles by supposing CTRV structure. The system pipeline was provided by the monitored lanes around the self-vehicle, which were accounted for by the planning layer, the status of the self-vehicle, including its mileage and speed and the corresponding bird's eye view (BEV) detection.

3.Our approach

The scale of traffic objects varies greatly, and the actual motion of the objects in traffic scenes is complex, so it is difficult to describe the motion pattern of the objects using only one model. Therefore, to address the above problems, the article proposes a multi-object tracking algorithm based on the improved YOLOv3 with interactive multi-model [25,

26] and object position correction algorithm. The algorithm uses the current state-of-the-art "detection + tracking" idea (shown in Figure 1). First, the algorithm uses the improved YOLOv3 model (YOLOv3_I) for multi-scale detection of the objects. Feature fusion technique is used and a scale branch is added to the YOLOv3 network to increase the accuracy of small object prediction. In addition, we also import a residual structure to enhance gradient propagation and avoid gradient disappearance or explosion for the whole network. The result of the object detection is taken as the input for the subsequent tracking, and the object detection frame and features are mainly adopted for the later tracker's matching calculation. Image pre-processing includes the usual operations such as data normalization, flipping, and refining the results by removing the objects with confidence less than 0.7. Non-maximum suppression is also used to get more accurate results. Second, in order to accommodate the complexity of moving objects, the interactive multi-model is used to calculate the motion state information of the object at a certain moment. An optimization algorithm corrects the position of objects to find an object detection frame for each predicted object. If it is found, it is judged as a tracker; otherwise, it is decided that the object has been lost and needs to be fine-tuned in the processing of the detection and prediction results. Then, the position correction algorithm is used to correct the object position. If the object detection box cannot find the corresponding predicted box that means that a new object has appeared. Finally, the objects are matched with the tracker,and the feature set is updated.

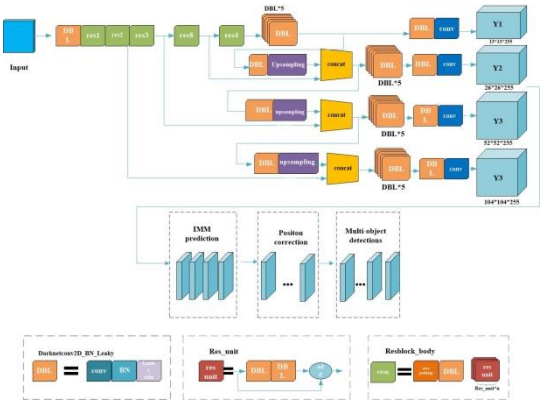


Figure 1. The network architecture of our approach

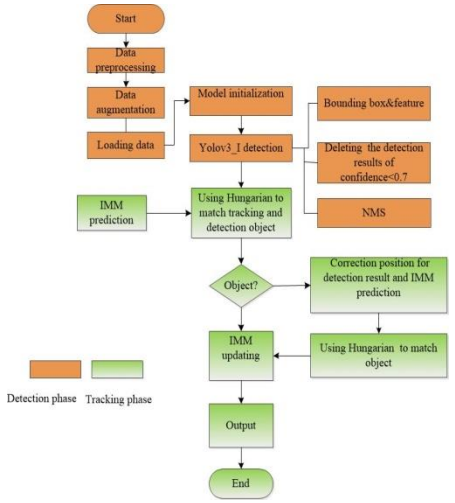


Figure 2. General flow chart of our multi-object tracking model

Figure 2 gives the flow chart of our model. It mainly contains two phases.

Detection phase:

Step 1: Split the original image into S*S cells or grids; each cell produces K bounding boxes according to the number of anchor boxes

Step 2: Employ the convolutional neural network to get features and predict the $b=[b_x, b_y, b_h, b_c]^T$, and the $class=[C_1,C_2,C_3,\ldots,C_c]^T$

Step 3: Compare the maximum confidence IoU_{thres}^{truth} of the K bounding boxes with the threshold IoU_{thres}, if $IoU_{thres}^{truth} > IoU_{thres}$, the bounding box has the object. Else, the bounding boxes do not contain the object.

Step 4: Select the class with the highest probability as the object category. Adopt NMS to operate a maximum part search for suppressing redundant boxes, output the results of object detection.

Tracking phase:

Step 1: Use IMM to predict tracks or Bbox for the current frame. If objects are confirmed detection results and prediction tracks begin correlation and matching. Else, the position correction algorithm is used for unmatched tracks and detections.

Step 2: Update the tracked Bbox after the matching is completed.

Step 3: After updating, the current frame is predicted, the next frame is observed, and updated; then predicted again. The next frame is observed and updated etc.

3.1. Improved yolov3 network

In traffic scenes, people, traffic lights, or cars in the input image have low resolution and belong to small objects, while the perceptual field of the convolutional layer at the end of the YOLOv3 backbone is very large. So it is a difficult task to detect the accurate objects with YOLOv3. In computer vision tasks, invariance and equivalent transformations are two important properties in image feature representation. Classification aims to learn high-level semantic information

and therefore invariant feature representations are required. The goal of object localization is to distinguish between position and scale changes, so it requires equivalence transformations. Object detection consists of two subtasks, object identification, and localization. While learning invariance and equivalence transformations are the keys to detection. The YOLOv3 is composed of a series of convolutional and pooling layers. The deep-level features have stronger invariance, but their equivalence transformation is weak. Although this is beneficial for classification recognition, it suffers from low localization accuracy in object detection. On the contrary, the shallow layer features are not conducive to semantic learning, but it contains more edge and contour information, which helps in object localization. The combination of multi-scale features can increase the global and local feature information in the model to improve the accuracy of object detection and increase the accuracy of subsequent object tracking. Therefore, to boot the detection performance of the model for small objects in traffic scenes, this paper uses the feature fusion technique to enhance the prediction of YOLOv3 by further integrating deep and shallow features in the model, adding a one-dimensional scale to strengthen small object prediction, and adding a residual structure in the corresponding scale branch to effectively control the gradient propagation and prevent the gradient from disappearing, degrading the network, and not easily converging, which brings about unfavorable network training. The network is easy to converge. This makes the training of the deeper network less difficult. The structure of the improved Yolov3 network is shown in Figure 3. The detailed fusion process can be described as follows:

$$ff m_{ip=} act_{conv}\left(act_{FC}\left(act_{US}\left(fm_{ir_{high}}\right), fm_{ir_{low}}\right)\right) \tag{1}$$

$$\mathrm{fm_{ir}} = \begin{cases} \mathrm{Action(f_i)}, & \mathrm{r} = 0 \\ \mathrm{Action(fm_{ir-1})}, & \mathrm{r} = \text{others} \end{cases} \tag{2}$$

Where $\mathrm{fm_{ir}}$ indicates the r_th layer feature map of $\mathrm{f_i}$, Action belongs to $\mathrm{act_{conv}}, \mathrm{act_{BN}}, \mathrm{act_{LR}}, \mathrm{act_{Add}}$ and stands for the action of convolution, BN, ReLu, and tensor addition. $\mathrm{ffm_{ip}}$ represents the p_th fused feature map. $\mathrm{fm_{ir_{high}}}$ and $\mathrm{fm_{ir_{low}}}$ show the high-resolution feature map and low-resolution feature map, respectively. $\mathrm{act_{US}}$ is the action of upsampling. $\mathrm{act_{FC}}$ is the action of full connection and $\mathrm{act_{conv}}$ represents the action of convolution. To describe the model later, the imporved model is abbreviated as YOLOv3_I.

The model is similar to YOLOv3, with only convolution layers. The size of the output feature map is controlled by adjusting the convolution step, so there is no special restriction on the input image size. Also drawing on the pyramid feature map idea, the small size feature map is applied to detect large-size objects, while the large size feature map detects small size objects. Finally, according to analysis to the dataset, a total of 4 feature maps are output, the first feature map is down-sampled 32 times, the second feature map 16 times, the third 8 times, and the fourth 4 times. Four detections, each corresponding to a different field of perception, 32 down-sampling has the largest field of perception, suitable for detecting large objects, 16 for objects of average size, 8 and 4 for the smallest field of perception, suitable for prediction small objects, and even smaller objects. Because objects have different pixels, we use different scales of anchor boxes to match. The anchor size of each cell is shown in table 1 according to the object size of the MOT16. The whole network, which draws the essence of Resnet [27], Densenet [28], and FPN [29], incorporates the current industry effective techniques of object detection to detect the object.

Table 1. Model anchor parameters

Feature map	13*13	26*26	52*52	104*104
Receptive field	large	medium	less	the smallest
A prior box	（156*198）	（30*61）	（10*13）	（10*11）
	（156*198）	（62*45）	（16*30）	（19*15）
	（371*326）	（59*119）	（33*23）	（25*21）

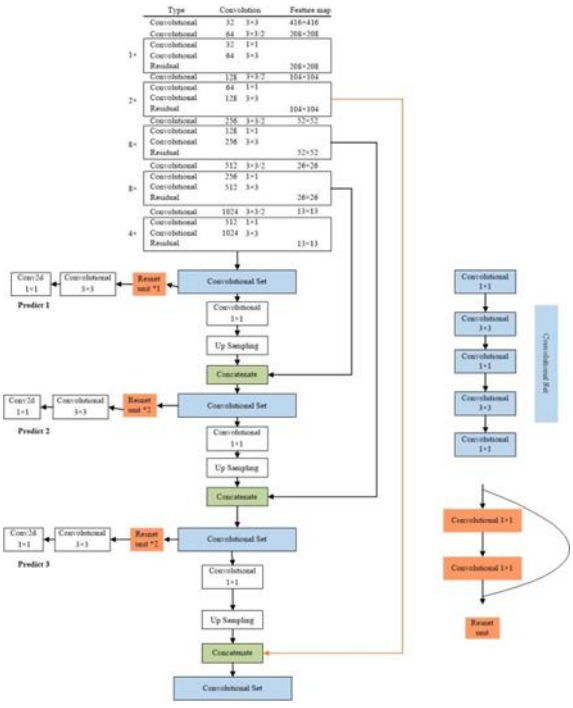


Figure 3. The structure of YOLOv3_I network

3.2. Interactive multi-model algorithms

Due to the complexity of the actual motion of the objects in the traffic scene, it is difficult to describe the motion pattern of the objects using only one model. Therefore, the algorithm in this paper adopts the Interacting Multiple Model (IMM) [30] to estimate the motion patterns of multiple objects (as shown in Figure 4). Its main feature is the ability to approximate the state of dynamic systems with multiple patterns of behavior that can be switched from one behavior pattern to another. In particular, the IMM evaluator can be a self-tuning variable bandwidth filter, which allows to naturally track maneuvering objects. IMM handles multi-object motion models in the Bayesian framework with a model set containing different motion sub-models, each corresponding to a filter. To solve the uncertainty of the object motion, each filter performs parallel operation computation and switches between models according to the updated weights. During object tracking, the fit of each sub-model of the system to the current object depends on the probability of the model. So, the interaction between models can be performed according to this principle, and finally, the probabilistic output results of each sub-model are fused.

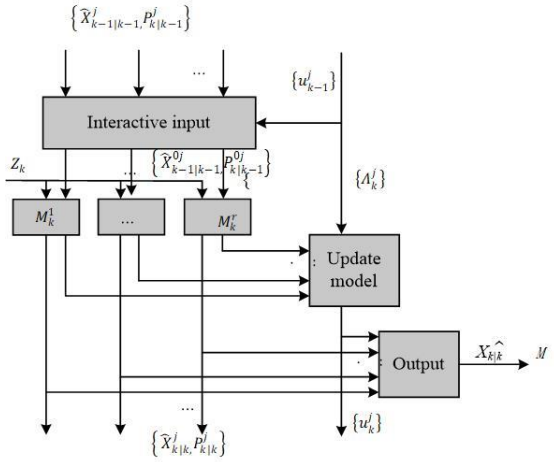


Figure 4. Flow chart of IMM algorithm.

(1). Model initialization: M_k^j denotes the j_th filter at the k_th frame, $j \in \{1, \ldots, r\}$.

Model initialization state $\hat{X}_{k-1|k-1}^{0j}$; Covariance matrix $P_{k-1|k-1}^{0j}$:

$$\hat{X}_{k-1|k-1}^{0j}=\sum_{i=1}^{r} \hat{X}_{k-1|k-1}^{i} u_{k-1|k-1}^{i|j} \quad (3)$$

$$P_{k-1|k-1}^{0j} = \sum_{i=1}^{r} u_{k-1|k-1}^{i|j} \left[P_{k-1|k-1}^{j} + \left(\hat{X}_{k-1|k-1}^{i} - \hat{X}_{k-1|k-1}^{0j}\right) \times \left(\hat{X}_{k-1|k-1}^{i} - \hat{X}_{k-1|k-1}^{0j}\right)^{T}\right] \quad (4)$$

Mixing probability $u_{k-1|k-1}^{i|j} = \bar{c}_j^{-1} p_{ij} u_{k-1}^{i}$; Normalization constants $\bar{c}_j = \sum_{i=1}^{r} p_{ij} u_{k-1}^{i}$; p_{ij} is the Markov transition probability of filter i to j.

(2). Model filtering estimation: in this paper, the standard IMM model is used, and the filtering phase has two main steps: prediction and correction. The prediction phase is responsible for calculating the a priori state estimates for each system in the current state. The correction phase is responsible for combining the actual measurements into each prior estimate to obtain the corresponding a posteriori state estimate. The motion and measurement models of the Kalman Filter are described as follows:

$$\hat{X}_{k-1|k-1}^{j}=A\hat{X}_{k-1|k-1}^{0j}+W_k^j \quad (5)$$

$$Z_k^j = H\hat{X}_{k|k-1}^j \quad (6)$$

$\hat{X}_{k|k-1}^{j}$ is M_k^j a priori state estimation. A, H are state transition matrice, measurement matrice respectively. Z_k^j represents the measured value at the k_th frame. W_k^j and V_k^j are the noise corresponding to the computational quantities obeying Gaussian distribution. Finally, the posterior state estimate is obtained.

$$\hat{X}_{k|k}^j=A\hat{X}_{k|k-1}^j+K_k^j(Z_k^j - H\hat{X}_{k|k-1}^j) \quad (7)$$

$Z_k^j - H\hat{X}_{k|k-1}^j$ is the residual of the motion model and the measurement model. K_k^j filtering gain is defined as:

$$K_k^j = P_{k-1|k-1}^J H^T (HP_{k|k-1}^j H^T + R)^{-1} \quad (8)$$

Where the a priori covariance matrix is:

$$P_{k|k-1}^j = AP_{k-1|k-1}^j A^T + Q \quad (9)$$

The posterior covariance matrix is updated as:

$$P_{k|k}^j = (I - K_k^j H) P_{k|k-1}^j \quad (10)$$

(3). Model probability updating. Calculating model probabilities:

$$u_k^i = \frac{1}{c} \Lambda_k^j \bar{\bar{c}}_j \quad (11)$$

$$c = \sum_{i=1}^{r} \Lambda_k^j \bar{\bar{c}}_i \quad (12)$$

Likelihood function Λ_k^j:

$$\Lambda_k^j = N\left(\tilde{Z}_k^j; 0, S_k^j\right) \quad (13)$$

The filter residual, $\tilde{Z}_k^j = Z_k^j - H\hat{X}_{k|k}^j$; $S_k^i = HP_{k|k-1}^j H^T + R$ is the corresponding filtering residual matrix. $N(\cdot)$ is Gaussian distribution. If the residuals are larger, it means that the model has a larger deviation from the object localization. Its weight decreases, and vice versa, the weight increases.

(4). Estimated fusion. Final state estimates and their covariances;

$$\hat{X}_{k|k}^j=\sum_{j=1}^{r} u_k^i \hat{X}_{k|k}^j \quad (14)$$

$$P_{k|k}=\sum_{j=1}^{r} u_k^j [P_{k|k}^j + (\hat{X}_{k|k}^j - \hat{X}_{k|k})(\hat{X}_{k|k}^j - \hat{X}_{k|k})^T] \quad (15)$$

3.3. Position correction optimization algorithm

In traffic scenes, traffic objects such as cars and pedestrians are usually very dense. Especially in the event of

traffic congestion, the objects are easily obscured and so on. When the object is obscured, the information of the object becomes incomplete and the object is easily lost. Therefore, in the paper, a position correction algorithm that fine-tunes the tracked object position is used in order to solve the lost-following phenomenon that occurs during the object tracking. The method corrects the object position based on the original predicted position, as shown in Figure 5. When the object is not found in the tracking area predicted by the IMM [31], the algorithm firstly considers that the object has been lost. Then, two points are selected from the location predicted by the IMM and the detection area of the detection algorithm. Eight-pixel points are added around them. Next, a fixed 125 threshold is set to compare the grayscale value of each point with the object's empirical grayscale value. The last, we get the best point from the ten points and determine whether the point belongs to the object position.

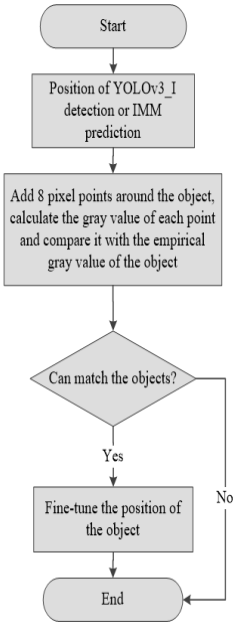


Figure 5. Flowchart of the position correction algorithm

3.4. Implement detail

The training parameters of the network are set as follows: the parameter of leak ReLu ai is 10; the hyper-parameters λnoobj, λobj, λcoord, λclass, λprior, of the loss function are 1,5,1,1 and 0.1; the N prior is 11200. The adaptive moment estimation is adopted to update the weights of the network; the momentum is 0.9, the decay is 0.0005, the batch size is 32; the initial learning rate is 0.001, and the learning rate on the 300th epoch and the 400th epoch is set to 0.1 times of the original.

4. Experiments

4.1. Experiment preparation

The experiments are operated on the computer with Intel Core i7-7700 CPU, 3.5GHz, 8G DDR4, 2666MHz memory, and Nvidia GTX1080Ti, with Ubuntu Linux 18.04 and Python3. Deep learning framework is tensorflow.

The performance of the proposed detection model is proven on MOT16 and KITTI datasets. The KITTI dataset is a dataset for autonomous driving, with cars and pedestrians. The MOT16 dataset is proposed in 2016 to measure standards for multi-object tracking detection and tracking methods. The performance of the proposed multi-object tracking algorithm is evaluated on the public MOT16 dataset [32]. The MOT16 benchmark evaluates the tracking performance of challenging test sequences, including forward-looking scenes with moving cameras and top-down surveillance settings. The dataset was divided into a training set and validation set. Finally, it is compared with other multi-object tracking algorithms on the test set of MOT16. The evaluation metrics are:

MOTA: Multi-Object Tracking Accuracy:

$$\text{MOT}=1-\frac{\text{FN}+\text{FP}+\text{IDs}}{\text{GT}} \tag{16}$$

GT is the number of Groundtruth. FN stands for the missing number. The number of frames t in which the object has no hypothetical position to match. FP is the number of object mismatches. The number of frames t in which the given hypothetical position has no tracking object to match it. IDs is a mismatch. The number of times the tracking object has an ID switch in frame t.

MOTP: Multi-Object Tracking Precise:

$$\text{MOTP}=\frac{\sum_{i,j} d_{i,j}}{\sum_t c_t} \tag{17}$$

c_t represents the number of matches between objects and hypotheses in frame t. $d_{i,j}$ denotes the distance between objects and their paired hypothesis positions in frame t.

4.2.The results and analysis of detection module

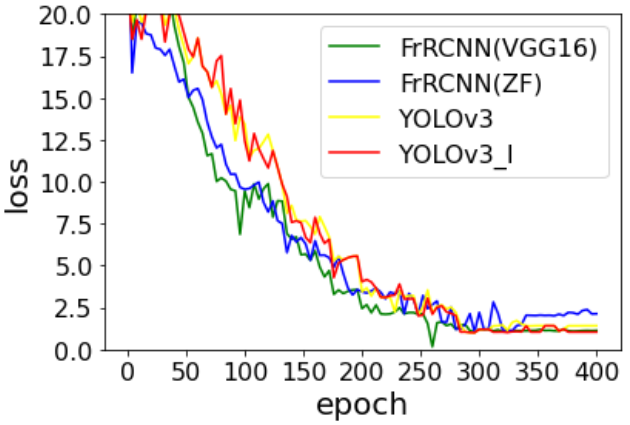
The evaluation metric of the object detection commonly contains the mean Average Precision (mAP), the Average Precision (AP), F1 score, Recall rate, and so on. The mAP is considered the average of AP of all object categories. Thus, we use mAP and F1 scores as the authoritative metric to evaluate the performance of our model. Tables 2 and 3 introduce typical and related evaluation results on MOT16 and KITTI datasets.

Table 2. Comparison performance of object detection model on mot16

Model (Detector)	Recall	Precision
FrRCNN(VGG16)	49.5%	77.5%
FrRCNN(ZF)	41.3%	72.4%
YOLOv3	48.2%	75.6%
YOLOv3_I	50.1%	77.4%

Table 3. Comparison performance of object detection model on KITTI

Model (Detector)	Recall	Precision
FrRCNN(VGG16)	63.9%	79.1%
FrRCNN(ZF)	60.1%	74.6%
YOLOv3	75.7%	78.9%
YOLOv3_I	79.8%	81.1%



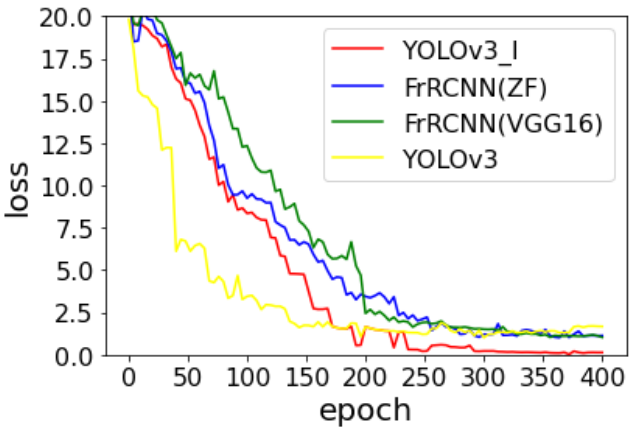


Figure 6. YOLOv3_I detection epoch vs loss on MOT16 (down) and KITTI (up)

Figure 7. YOLOv3_I detection results schematic diagram on MOT16

Figure 8. YOLOv3_I detection results schematic diagram on KITTI

Referring to the idea in SORT tracking algorithm, detection performance has a significant impact on tracking performance. A good detector can result in the best tracking accuracy [33]. Faster R-CNN is the popular and classical detection algorithm in the study. In this paper, we also hope to improve the tracking performance by improving the detector accuracy, as shown in Table 2 and 3, YOLOv3_I outperforms Faster R-CNN in terms of object detection effect and has more advantages in detection speed and accuracy, which is more beneficial to have better behavior of multi-object tracking. In the process of pedestrian detection, the paper uses the parameters obtained from training in PASCAL VOC for initialization. The Table compares two network architectures provided by FrRCNN, namely Zeiler and Fergus' architecture (FrRCNN(ZF)) [34] Simonyan and Zisserman's deeper architecture (FrRCNN(VGG16)) [35] and the paper's improved multi-scale fusion model based on YOLOv3, named YOLOv3_I which shows that the model used in this paper obtains better detection recall and accuracy. It can be facilitated later object tracking. Figure 6 gives the detection result of epoch vs loss on MOT16 and KITTI datasets. The proposed model compared with other models has good convergence speed. Figures 7 and 8 show the schematic diagrams of the detection output of the YOLOv3_I model, in which the corresponding confidence, classification, and localization are given for the detected objects. In the final

larger objects can be examined in relatively smaller feature maps, which can the detection objects at different scales. tracking part of the experiments, only the objects with the detection probability confidence greater than 50% are passed to tracking model in the paper.

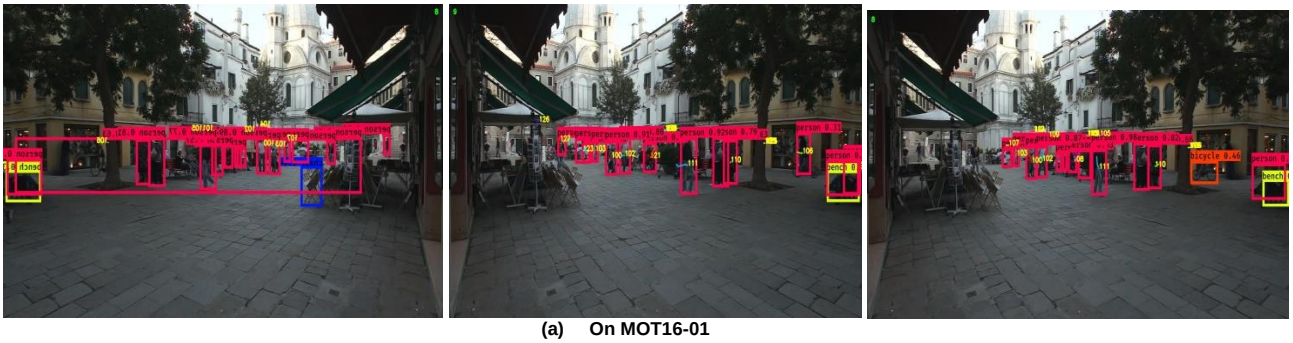
4.2. Analysis of experimental results of multi-object tracking model

The proposed method is compared with several existing object tracking methods on the MOT2016 to test the various performances of the algorithm, as shown in Table 4. All benchmark testing methods and our approach use the same publicly available test results for a fair comparison. The proposed method outperforms previous approaches in terms of MOTP, ML, and Frag. The offline tracking method can access all future detection results and reasons on the data association step. The results show that the MOTP of the proposed method is better than that of the offline method with about 3 percentage points higher performance. The technique using object correction can find the correct object after occlusion or drift. Therefore, our experiment results have higher MT and lower ML, but the IDS is higher. When the objects are occluded, the proposed method in the paper may incorrectly assign them to other detection objects. However, when the objects reappear, the methods proposed may re-match them with the correct detection results. Such a process can lead to a large number of switches. But the Frag metric is still high.

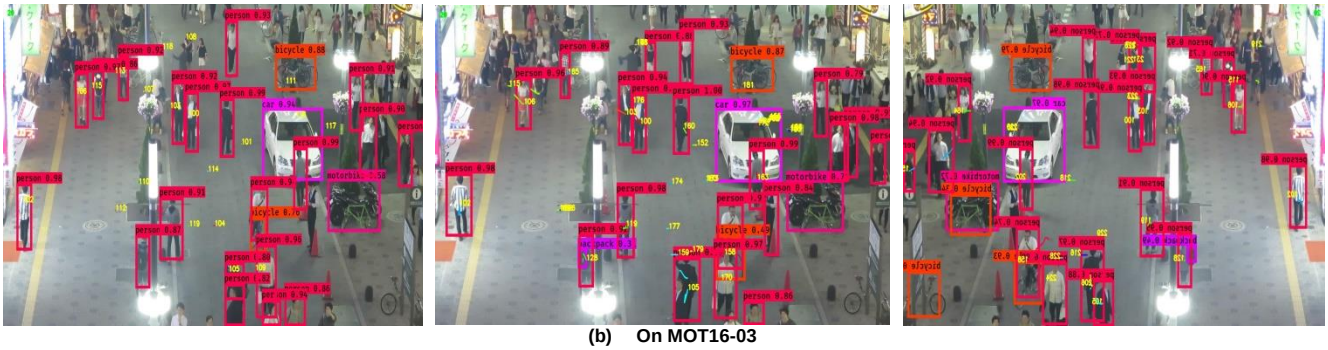
The red boxes represent the person and give confidence to the corresponding objects. The yellow numbers give the ID of the pedestrian tracking (only the pedestrian ID is given because the objects are too dense) in Figure 9. In this figure, cars or pedestrians move from far to near or near to far, showing the same class or the scale size of the object change over time, which reflects the different scale changes of the object. Correspondingly, YOLOv3_I uses 4 scale features for fusion to obtain detection results in feature maps of different depths. Smaller objects can be detected in larger feature maps, while

Table 4. Benchmark tracking performance comparison on MOT16

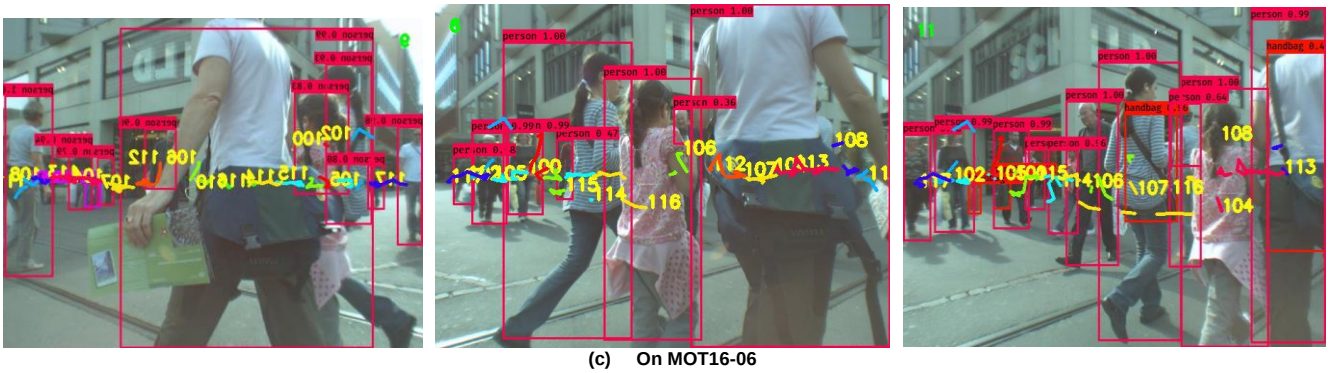
Method	Mode	MOTA ↑	MOTP ↑	MT ↑	ML ↓	IDs ↓	Frag ↓
MCjoint[36]	Offline	47.1	76.3	20.4	46.9	370	598
SMOT[37]	Offline	29.7	72.5	5.3	47.1	3108	4483
NOMT[19]	Offline	46.4	76.6	18.3	41.4	359	504
LMP[38]	Offline	48.8	79.0	18.2	40.1	481	595
OVBT[39]	Online	38.4	75.4	7.5%	47.3%	1321	2140
EAMTT[40]	Online	38.8	75.1	7.9%	49.1%	965	1657
oICF[41]	Online	43.2	74.3	11.3%	48.5%	381	1404
SORT[33]	Online	59.8	79.6	25.4	22.7	1101	1664
Deep SORT[42]	Online	61.4	79.1	32.8	18.2	781	1023
Our	Online	61.7	80.6	23.5	16.3	837	1705



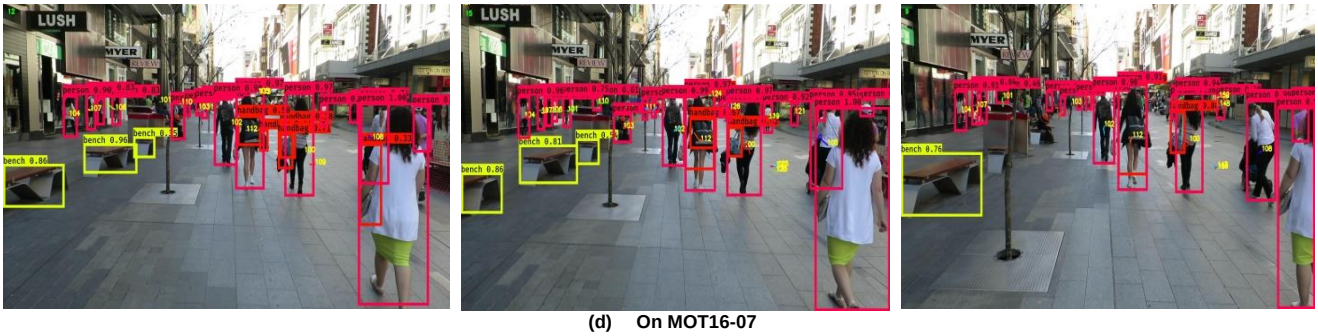
(a) On MOT16-01



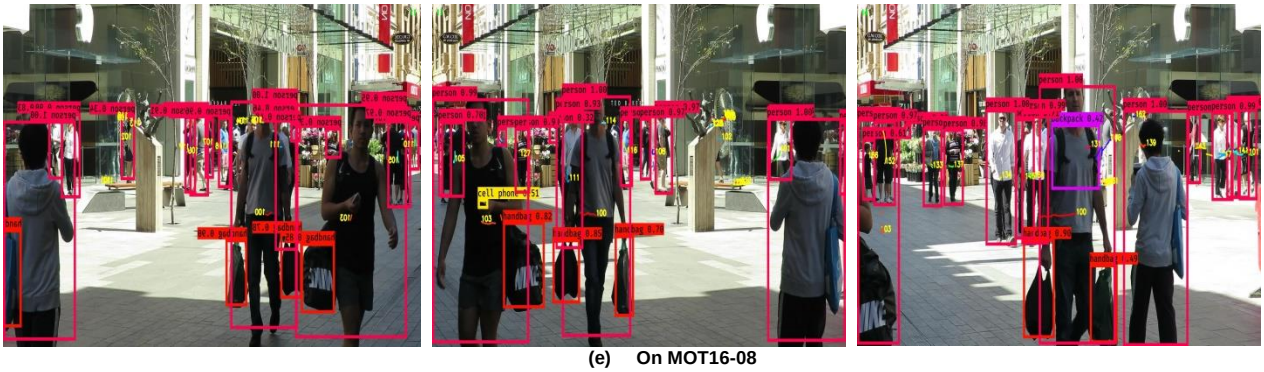
(b) On MOT16-03



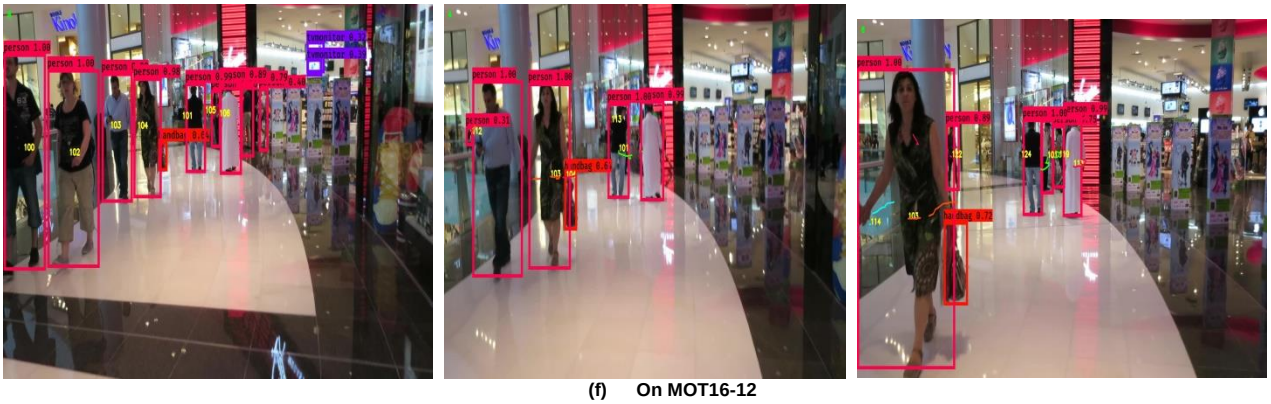
(c) On MOT16-06



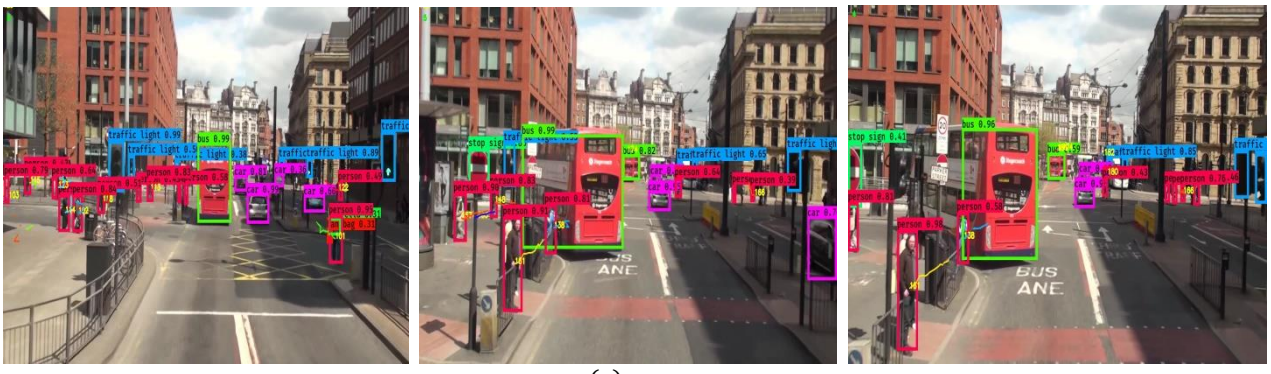
(d) On MOT16-07



(e) On MOT16-08



(f) On MOT16-12



(g) On MOT16-14

Figure 9.The results of our method on MOT16 (a,b,c,d,e,f,g)

5. Conclusion

The traffic scenes are complex and changeable, which brings great difficulty to implement the Intelligent Transportation System. In this paper, we propose a multi-object tracking method for traffic scenes. In multi-object detection, simple objects can be distinguished by using shallow features, while deeper features can identify more complex objects, so this paper uses multi-scale feature fusion technology to adapt to the changes of different scales of multi-object objects. A scale prediction is added to the original YOLOv3 network so that the accuracy of small object prediction can be increased, and the residual structure is also introduced to enhance the gradient propagation and avoid gradient disappearance or explosion. Next, the traffic scene is dense with objects, and occlusion between objects tends to occur. When the tracked object is occluded, the information of the object becomes incomplete and thus the phenomenon of the loss happens. This paper develops a position correction algorithm to fine-tune the position of the tracked object. And when the object reappears, the occluded object can be matched with the correct detected object again. The IMM algorithm is also dedicated to the tracking stage to adapt to the complex object changes in the traffic scene and improve the tracking accuracy.

The proposed multi-object tracking method in this paper is mainly based on the mainstream idea of "detection + tracking", which achieves a certain tracking effect and has a certain application value. However, for complex traffic scenarios, the method still has some shortcomings, so the follow-up work will be carried out. To improve the overall detection and tracking performance for multiple objects in a scene, the work can be considered to study the detection and tracking system in terms of the inter-influence relationship

between objects, or the interconnectedness between traffic objects. Another research direction is to combine the object detection output results with the tracking model to further investigate detection-based end-to-end tracking algorithms.

REFERENCES

[1]. Xu, Y.; Zhou, X.; Chen, S.; Li, F. Deep learning for multiple object tracking: a survey. IET Computer Vision, 2019, 13, 4, 355-368.

[2]. Ciaparrone, G.; Sánchez, F.L.; Tabik, S.; Troiano, L.; Tagliaferri, R.; Herrera, F. Deep learning in video multi-object tracking: A survey. Neurocomputing, 2020, 381, 61-88.

[3]. Li, X.; Wang, K.; Wang, W.; Li, Y. A multiple object tracking method using Kalman filter. The 2010 IEEE international conference on information and automation, IEEE, 2010. 1862-1866.

[4]. Henriques, J.F.; Caseiro, R.; Martins, P.; Batista. High-speed tracking with kernelized correlation filters. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2014, 37, 3, 583-596.

[5]. Li, X.; Hu, W.; Shen, C.; Zhang, Z.; Dick, A.; Hengel. A survey of appearance models in visual object tracking. ACM transactions on Intelligent Systems and Technology (TIST), 2013, 4, 4, 1-48.

[6]. LeCun, Y.; Bengio, Y.; Hinton, G. Deep learning. nature, 2015, 521, 7553, 436-444.

[7]. Yu, F.; Li, W.; Li, Q.; Liu, Y.; Shi, X.; Yan, J. POI: Multiple Object Tracking with High Performance Detection and Appearance Feature, Springer International Publishing, Cham, 2016. 36-42.

[8]. Zhang, L.; Gray, H.; Ye, X.; Collins, L.; Allinson, N. Automatic individual pig detection and tracking in pig farms. Sensors,2019, 19, 5, 1188.

[9]. Zhao, D.; Fu, H.; Xiao, L.; Wu, T.; Dai, B. Multi-Object Tracking with Correlation Filter for Autonomous Vehicle. Sensors, 2018, 18, 7, 2004.

[10]. Wang, Y.; Wei, X.; Tang, X.; Shen, H.; Zhang, H. Adaptive Fusion CNN Features for RGBT Object Tracking. IEEE Transactions on Intelligent Transportation Systems ,2021.

[11]. Xie, Y.; Shen, J.; Wu, C. Affine geometrical region CNN for object tracking. IEEE Access, 2020, 8, 68638-68648.

[12]. Chintalacheruvu, N.; Muthukumar, V. Video based vehicle detection and its application in intelligent transportation systems. Journal of transportation technologies, 2012, 2, 04, 305.

[13]. Hinz, G.; Chen, G.; Aafaque, M.; Röhrbein, F.; Conradt, J.; Bing, Z.; Qu, Z.; Stechele, W.; Knoll, A. Online multi-object tracking-by-clustering for intelligent transportation system with neuromorphic vision sensor, Joint German/Austrian Conference on Artificial Intelligence (Künstliche Intelligenz), Springer, 2017. 142-154.

[14]. Liang, H.; Song, H.; Li, H.; Dai, Z. Vehicle counting system using deep learning and multi-object tracking methods. Transportation research record, 2020, 2674, 4, 114-128.

[15]. Pino, I.d.; Vaquero, V.; Masini, B.; Sola, J.; Moreno-Noguer, F.; Sanfeliu, A.; Andrade-Cetto, J. Low resolution lidar-based multi-object tracking for driving applications, Iberian Robotics conference, Springer, 2017. 287-298.

[16]. Liu, Y.; Li, X.; Bai, T.; Wang, K.; Wang, F.Y. Multi-object tracking with hard-soft attention network and group-based cost minimization. Neurocomputing ,2021, 447, 80-91.

[17]. Abbas, Q. V-ITS: Video-based intelligent transportation system for monitoring vehicle illegal activities. Int. J. Adv. Comput. Sci. Appl, 2019, 10, 3, 202-208.

[18]. Dirir, A.; Adib, M.; Mahmoud, A.; Al-Gunaid, M.; El-Sayed, H. An Efficient Multi-Object Tracking and Counting Framework Using Video Streaming in Urban Vehicular Environments, 2020 International Conference on Communications, Signal Processing, and their Applications (ICCSPA), IEEE, 2021. 1-7.

[19]. Nizar, T.N.; Anbarsanti, N.; Prihatmanto, A.S. Multi-object tracking and detection system based on feature detection of the intelligent transportation system, 2014 IEEE 4th International Conference on System Engineering and Technology (ICSET), IEEE, 2014. 1-6.

[20]. Tian, W.; Lauer, M.; Chen, L. Online multi-object tracking using joint domain information in traffic scenarios. IEEE Transactions on Intelligent Transportation Systems ,2019, 21, 1, 374-384.

[21]. Rangesh, A.; Trivedi, M.M. No blind spots: Full-surround multi-object tracking for autonomous vehicles using cameras and lidars. IEEE Transactions on Intelligent Vehicles, 2019, 4, 4, 588-599.

[22]. Vasic, M.; Martinoli, A. A collaborative sensor fusion algorithm for multi-object tracking using a Gaussian mixture probability hypothesis density filter, 2015 IEEE 18th International Conference on Intelligent Transportation Systems, IEEE, 2015. 491-498.

[23]. Emami, P.; Elefteriadou, L.; Ranka, S. Long-Range Multi-Object Tracking at Traffic Intersections on Low-Power Devices. IEEE Transactions on Intelligent Transportation Systems ,2021.

[24]. Gómez–Huélamo, C.; Bergasa, L.M.; Gutiérrez, R.; Arango, J.F.; Díaz, A. SmartMOT: Exploiting the fusion of HDMaps and Multi-Object Tracking for Real-Time scene understanding in Intelligent Vehicles applications, 2021 IEEE Intelligent Vehicles Symposium (IV), IEEE, 2021. 710-715.

[25]. Redmon, J.; Farhadi, A. Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767 ,2018.

[26]. Faure, F.; Duriez, C.; Delingette, H.; Allard, J.; Gilles, B.; Marchesseau, S.; Talbot, H.; Courtecuisse, H.; Bousquet, G.; Peterlik, I. Sofa: A multi-model framework for interactive physical simulation, Soft tissue biomechanical modeling for computer assisted surgery, Springer2012. 283-321.

[27]. Wu, Z.; Shen, C..; Hengel, A. Wider or deeper: Revisiting the resnet model for visual recognition. Pattern Recognition, 2019, 90, 119-133.

[28]. Iandola, F.; Moskewicz, M.; Karayev, S.; Girshick, R.; Darrell, T.; Keutzer, K. Densenet: Implementing efficient convnet descriptor pyramids. arXiv preprint arXiv:1404.1869, 2014.

[29]. Ghiasi, G.; Lin, T.-Y..; Le, Q.V. Nas-fpn: Learning scalable feature pyramid architecture for object detection, Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019. 7036-7045.

[30]. He, R.; Lee, W.S.; Ng, H.T.; Dahlmeier, D. An interactive multi-task learning network for end-to-end aspect-based sentiment analysis. arXiv preprint arXiv:1906.06906, 2019.

[31]. Welch, G.F. Kalman filter. Computer Vision: A Reference Guide,2020, 1-3.

[32]. Dendorfer, P.; Rezatofighi, H.; Milan, A.; Shi, J.; Cremers, D.; Reid, I.; Roth, S.; Schindler, K.; Leal-Taixé, L. Mot20: A benchmark for multi object tracking in crowded scenes. arXiv preprint arXiv:2003.09003 ,2020.

[33]. Fu, H.; Wu, L.; Jian, M.; Yang, Y.; Wang, X., MF-SORT: Simple online and Realtime tracking with motion features, International Conference on Image and Graphics, Springer, 2019. 157-168.

[34]. Zeiler, M.; Fergus, R. Visualizing and understanding convolutional networks. arXiv 2013. 2019.

[35]. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 ,2018

[36]. Keuper, M.; Tang, S.; Zhongjie, Y.; Andres, B.; Brox, T.; Schiele, B. A multi-cut formulation for joint segmentation and tracking of multiple objects. arXiv preprint arXiv:1607.06317 ,2016.

[37]. Choi, W., Near-online multi-object tracking with aggregated local flow descriptor, Proceedings of the IEEE international conference on computer vision, 2015. 3029-3037.

[38]. Tang, S.; Andriluka, M.; Andres, B.; Schiele, B. Multiple people tracking by lifted multicut and person re-identification, Proceedings of the IEEE conference on computer vision and pattern recognition, 2017. 3539-3548.

[39]. Ban, Y.; Ba, S.; Alameda-Pineda, X.; Horaud, R. Tracking multiple persons based on a variational bayesian model, European Conference on Computer Vision, Springer, 2016. 52-67.

[40]. Sanchez-Matilla, R.; Poiesi, F.; Cavallaro, A. Online multi-object tracking with strong and weak detections, European Conference on Computer Vision, Springer, 2016. 84-99.

[41]. Kieritz, H.; Becker, S.; Hübner, W.; Arens, M. Online multi-person tracking using integral channel features, 2016 13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), IEEE, 2016. 122-130.

[42]. Wojke, N.; Bewley, A.; Paulus, D. Simple online and realtime tracking with a deep association metric, 2017 IEEE international conference on image processing (ICIP), IEEE, 2017. 3645-3649.