A temporal-based deep learning method for multiple objects detection in autonomous driving

Yaran Chen, Dongbin Zhao, Haoran Li, Dong Li, and Ping Guo

Abstract— **This paper proposes a novel vision-based object detection method in autonomous driving, which introduces the temporal information into the deep learning-based detection method for moving object detection. Vision-based object detection is a critical technology for autonomous driving. The objects in the real world such as driving cars, don't have great changes in their positions and velocities. So the position change of objects between two consecutive frames is not large. This is usually ignored by traditional works, which usually use object detection methods on still-images to detect moving objects. Considering the relationship among consecutive frames (temporal information), we present a robust and real-time tracking method following image detection to refine the object detection results. Based on the three key attributes (distances, sizes and positions), the tracking method aims to build the association between the detected objects on the current frame and those in previous frames. The proposed object detection with temporal information dramatically improves the performance of existing object detection algorithms based on still-image. With the proposed method, we won the champion in the preceding vehicle detection task in 2017 intelligent vehicle future challenge(2017 IVFC)[1].**

I. Introduction

Recently, autonomous driving has been widely studied and shown great promising. Object detection is a crucial technology of autonomous driving. It has attracted increasing attention of researchers. In the real traffic, the preceding vehicle detection aims to detect the moving vehicles, which have no great changes in terms of their positions. The preceding vehicle detection belongs to the video detection. Due to the moving objects, video object detection is more difficult than still-image detection, but it also shows more promising. 2017 Intelligent Vehicle Future Challenge (2017 IVFC) has a task to detect preceding vehicles from videos.

Currently, most works use still-image detection algorithms for the preceding vehicle detection. As the success of deep convolutional neural networks(CNN) [1], [2], [3], the CNN-based object detection algorithms also have significantly improved the detection performance, such as SSD [4], Faster R-CNN [5] and Mask R-CNN [6]. However, these detection algorithms are specifically designed for still-images without considering the temporal information, and not suitable for video detection.

Due to the bad poses, motion blur, and similar sizes of moving vehicles, the result of the still-image detection may have some false negatives and false positives. To solve this problem, this paper introduces the temporal information among video frames into the still-image detection algorithm. In a video, the positions and appearances of objects are temporal consistent, namely these objects don't unexpectedly appear or disappear. Based on these attributes, we can recover the missing object and remove the wrong detected objects. For example, a certain object disappears in the current frame, but it still exists in the adjacent frames. Then we can make up this frame by propagating the last frame detection result. If an object appears in this frame, but it doesn't exist in the adjacent frames. Then we can consider the object is a wrong detection result and remove it.

In order to consider the temporal information, this paper proposes a novel tracking method to link all the detected results from a target detector and generate the trajectory of the detected target. In video detection, there are many vehicles in each frame. The major challenge of tracking is how to build the associations among detected objects with the tracking targets, namely how to judge which target the newly detected object belongs to. The common tracking methods can be classified into two types: off-line tracking and online tracking methods. For the off-line tracking, many works [7] are based on video sequences, using video frames from the previous and future time steps to solve the object association. However, in autonomous driving, the future frames are not available. Online tracking methods use a similarity function between the detect object and the tracked target to solve the association. The similarity function is modeled with some parameters relying on appearance, motion and location of an object [8]. For different objects, the parameters need to be fine-tuning. So these methods are also not suitable for autonomous driving due to poor generalization.

In order to solve the data associations among the detected objects with the tracking targets, we utilize three key attributes of detected objects: positions, sizes and distances. The position of the object in a frame shows a visual distance. Different visual distances in the frame represent different physical distances from the host car, especially the horizontal distances. The size of the object in a frame is also a key attribute. Different objects may have different sizes, such as a sedan and a bus. At the same time, for a driving vehicle with 60km/h, the change of its distance in two adjacent frames is

Y. Chen, D. Zhao, H. Li and D. Li are with The State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China. They are also with the University of Chinese Academy of Sciences, Beijing, China. (e-mail: chenyaran2013@ia.ac.cn; dongbin.zhao@ia.ac.cn; lihaoran2015@ia.ac.cn;lidong2014@ia.ac.cn)

P. Guo is with School of Systems Science, Beijing Normal University, Beijing, 100875, China

This work is supported by the Beijing Natural Science Foundation under Grant No. L172054, National Natural Science Foundation of China (NSFC) under Grants No. 61573353 and No. 61533017,and the National Key Research and Development Plan under Grant No. 2016YFB0101000.

[1]http://mp.weixin.qq.com/s/IDrTDlJqb2Qx360nhgCXDw

slight, about 0.2m. If there are two cars in the same lane with the host car and they are quite close to each other, the two cars have some overlap in the frame. So the positions and sizes of the two cars are similar. In this case, the distances of two cars in the real world are relatively different at least a length of a car. So the distance is also necessary for tracking targets.

Based on these key attributes, we propose a filter to tracking multiple vehicles, shown in Fig. 1. Specifically, we develop a neural network that learns the data associations based on the above attributes from several previous video frames and judges which target the newly detected object belongs to. A CNN-based object detection network is introduced to get positions and sizes of objects. We derive the relationship between the image coordinate system and the world coordinate system, transforming the point in the image into the world coordinate. Then we can achieve the detected object distance through coordinate transformation.

In this paper, we propose a robust approach to video detection for autonomous driving. It contains an image-based object detection and a multi-object tracking to refine the detection results. A CNN-based object detection method is used to get positions and sizes of objects in an image. For each detected object, we use a coordinate transformation to generate a top view of the road and regress the object distance, considering the object position and the camera parameter. Then, based on the positions, sizes and distances of objects, we train a tracking neural network to link the detected objects with the targets and refine the object positions. The proposed system is a temporal-based video object detection method, called TBVOD. The proposal TBVOD achieved the champion of the preceding vehicle detection in 2017 IVFC.

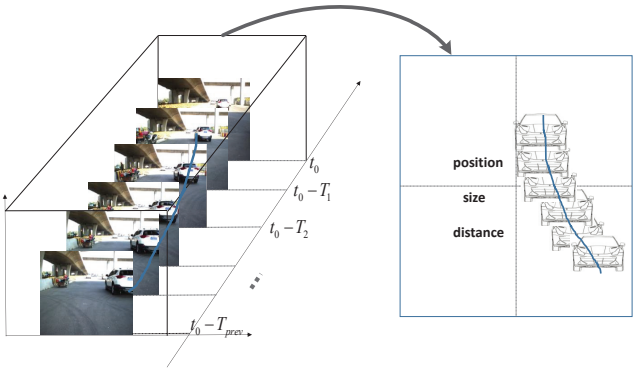

Fig. 1. Tracking-by-detection based on positions, sizes and distances of objects

II. Related Work

Preceding vehicle detection is a typical video detection problem, which is usually dealt with still-image detection methods. In this paper, we introduce the temporal information into the still-image detection method to improve the detection performance. The proposed tracking method learns data association from positions, sizes and distances of objects. The distance is usually obtained by the non-vision-based method and vision-based method. The vision-based method gets distance only from visual sensor. [9] uses the geometric relationship of the objects in images to calculate distance. Some works get the depth of images with deep CNN [1], [10], [11], while CNN needs large labeled images. Our method doesn't need lots of training images by using coordinate translation to get vehicle distances. The tracking problem is usually solved by tracking-by-detection in the computer vision field. There are two approaches of tracking-by-detection: online methods and off-line methods. The off-line methods analyze a batch of video sequences which contain video frames from the future time. While the future video frames are not available for autonomous driving. For the online methods, [12] proposes a probability-based association for linking detected objects, [13] uses a determinative association and [14] utilizes a greedy association. We propose a robust data association based on positions, sizes and distances of objects.

III. Video Object Detection Framework

The proposed video object detection method TBVOD is a deep convolutional neural network based video object detection framework that incorporates still-image detection with multi-objects tracking of which utilizes the temporal information of videos. The overall framework is shown in Fig. 2. In the following, we will introduce each major component in detail.

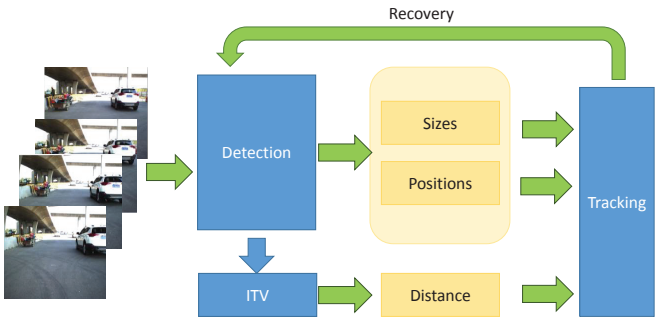

Fig. 2. The framework of the proposed video detection method TBVOD

A. Object detection

Nowadays, the performance of image detection has been improved with the successes of deep convolutional neural network models and the object detection methods. Among these methods, Faster R-CNN (Faster Region-based Convolutional Network method) [5] has a better performance of detecting small objects, which is suitable for detecting the vehicles far away on the road. In our still-image detection, we use Faster R-CNN as the detection framework to get object sizes and positions.

The Faster R-CNN method contains multiple convolutional layers ($\mathbf{M_{CNN}}$), region proposal network (PRN) and a lot of detectors ($\mathbf{M_{det}}$), shown in Fig. 3. An image $\mathbf{X}_{im}$ is input

into $\mathbf{M_{CNN}}$, which can extract object features and generate many feature maps f_{map}.

$$f_{map} = \mathbf{M_{CNN}}(\mathbf{X}_{im}, W_{cnn}), \tag{1}$$

where $\mathbf{W_{cnn}}$ represents the parameters of $\mathbf{M_{CNN}}$. In some feature maps, we set n default bounding boxes (B_{def}). Through classification, the RPN can generate the probabilities of these default bounding boxes containing objects. The default bounding boxes with high probabilities (noted as B_{pro} shown in Fig. 3) will be input into $\mathbf{M_{det}}$.

$$B_{pro} = \text{RPN}(B_{def}). \tag{2}$$

Each detector $\mathbf{M_{det}}$ contains a classification with a softmax layer and a position regression layer, to get which type the object in the default bounding box belongs to and where the object is.

$$(\mathbf{P_{class}}, [x, y, w, h]) = \mathbf{M_{det}}(B_{pro}), \tag{3}$$

where $\mathbf{P_{class}}$ is a probability matrix, meaning the classification probability distribution of the B_{pro}. (x, y) is the object position in the image and w, h respect the width and height of the object, separately.

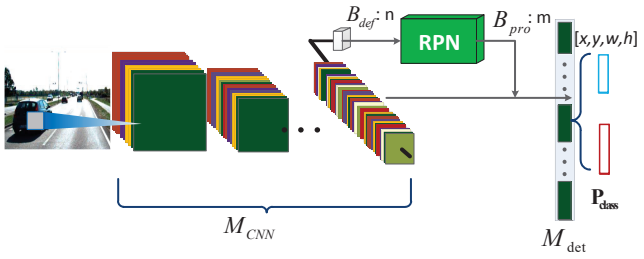

Fig. 3. The framework of Faster R-CNN object detection method.

Owing to strong capacity of CNN in feature transfer learning, the early multiple CNN layers usually inherit the CNN parameters which are trained on a large scale dataset, such as ImageNet dataset. It means that the early multiple CNN layers set initialization parameters with a trained CNN model. The early multiple CNN layers are called pre-training CNN model. As is well-known, increasing the number of CNN layers can improve the capability of learning object feature representation. Among these well-known CNN models, ResNet model [1] has many layers and owns a strong capability of extracting object features. So we choose ResNet model as the pre-training CNN model in the object detection.

The objects have a various of shapes and sizes in images. We gather all common objects such as vehicles pedestrian and so on, and cluster them into k groups based on their width and height. These objects mainly have k aspect ratios in images. In order to detect all shapes of objects, we set a group of default bounding boxes with different aspect ratios. These aspect ratios of default bounding boxes are the center points of the k clustered groups, which can cover all the shapes of objects.

For each image, we can get three hundred proposals and every proposal has a classification confidence (score), namely the probability of the proposal belongs to one class. These proposals are sorted by scores in descending order. The objects which have high detection scores are regarded as high confidences, and others are considered as low confidences. At last, we perform non-maximum suppression filter on these high-confidence objects to generate the final detection results.

B. Temporal based multi-object tracking

For video detection, the still-image detection methods have ignored the temporal information. It may cause several false negatives and false positives due to bad pose, motion blur, similar objects. In fact, the adjacent frames have a remarkably high correlation, so their detection results should also be highly correlated in positions and detection credibilities. For example, if a vehicle moves at a low speed with the camera, it would appear in nearby position in adjacent frames. So the temporal information can recover false negatives. On the other side, it's impossible that a vehicle or pedestrian suddenly appears in a close area in front of the host car. Then the temporal information also can recover the false positive. Considering the temporal information, we develop a multi-object tracking method to recover false positives and false negatives.

Through the still-image detection method, we can get object position and size. Then the proposed multi-object tracking associates these objects of video frames, and generates multiple object trajectories. The proposed multi-object tracking framework considers three key attributes (positions, sizes, and distances) of objects to build a neural network and explore temporal information.

Positions and sizes of objects: The position and size of an object in a video can not have dramatic changes, which are regular according to the object speed and orientation relative to the host car. For example, a preceding car driving at a certain speed and orientation. In the period of a car appearance and disappearance, the car positions in all the frames can be linked as time order, generating a smooth and continuous trajectory. If a newly detected car appears in this trajectory, it is highly probable that the newly detected car is the tracked car. In this paper, we can achieve the object position and size from the still-image detection framework.

Distances of objects: Another key attribute of building data association in tracking is the object distance. Even a certain object in a video sequence has a smooth and continuous trajectory. Other objects may also appear in this trajectory due to the limitation of the visual image perspective. For example, two preceding cars within a small distance have the nearby positions in the image, while the distance related to the host car is much different at least a length of a car. So it needs to locate the position of the object in the real world, namely the object distance.

In this paper, we use a coordinate transformation method to generate a top view map (TVM) of the road, in which each point can be calculated the distance related to the host car. The detected object can be projected into the TVM shown

in Fig. 6. In the figure, the red line is the intersection of the detected object with the road. Then we can get the object distance according to the red line.

In order to generate the TVM, we assume that the road is flat, and the camera parameters (focal length, optical center, pitch angle, yaw angle, and height aboveground) are available. We define a world coordinate $\{C_w\} = \{X_w, Y_w, Z_w\}$, a camera coordinate $\{C_c\} = \{X_c, Y_c, Z_c\}$ and an image coordinate $\{C_i\} = \{u, v\}$, shown in Fig. 4.

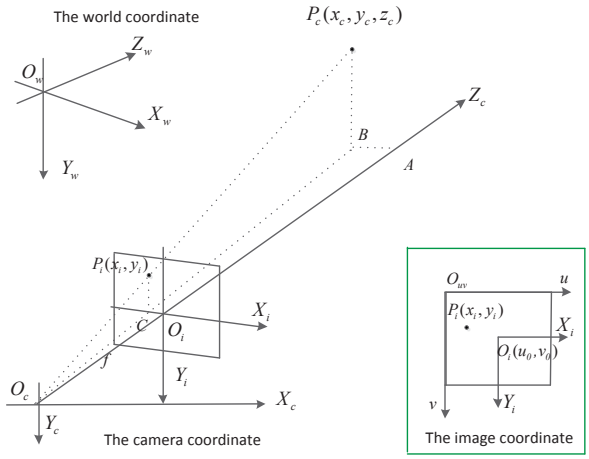

Fig. 4. The world coordinate, camera coordinate and image coordinate

The point $P_w = (x_w, y_w, z_w)^T$ in the world coordinate can be translated into the image coordinate under rotation and displacement:

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ \mathbf{0} & \mathbf{1} \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} = T_c^w \begin{bmatrix} x_w \\ y_w \\ z_w \\ 1 \end{bmatrix} \tag{4}$$

where $\mathbf{R}$ and $\mathbf{t}$ are the matrices for rotate, and translate, respectively. The point $P_c = (x_c, y_c, z_c)^T$ in the image coordinate can be calculated by $P_c = T_c^w P^w$. The point P_c is projected onto an image and generates the point $P_i = (x_i, y_i)^T$. Fig .4 shows two similar triangles: $\Delta ABO_c \sim \Delta O_i C O_c$ and $\Delta BO_c P_c \sim \Delta CO_c P_i$. According to the similarity triangle theory, we can get:

$$\frac{AB}{O_iC} = \frac{AO_c}{O_iO_c} = \frac{P_cB}{P_iC} = \frac{x_c}{x_i} = \frac{y_c}{y_i} = \frac{z_c}{f} \tag{5}$$

$$z_c \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} = \begin{bmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} \tag{6}$$

The coordinate (x_i, y_i) is translated into the image coordinate (u, v) by the following equation:

$$\begin{aligned} u &= x_i/dx + u_0 \\ v &= y_i/dy + v_0 \end{aligned} \tag{7}$$

where $\{u_0, v_0\}$ is the coordinate of the camera's optical center. If the physical units of the image is millimeter, dx and dy express that a pixel indicates dx millimeter in the direction of the u axis and dy millimeter in the direction of the v axis.

Using the above transformations, we can project a camera image into a ground plane, namely the top view map. Fig. 6 presents a TVM projection sample, where the left side is the original image, and the right side is the TVM image. We use a red line to tag the interaction of the car with the road, and the distance of red line corresponding to the distance of the car.

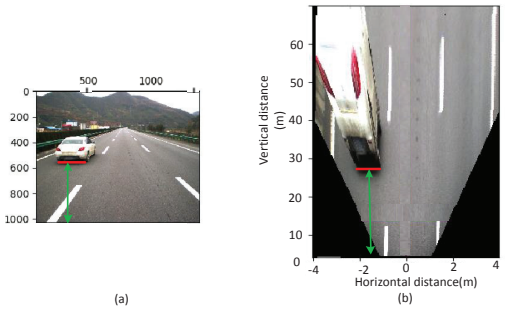

Fig. 6. An ITV sample of an image. (a) the raw image; (b) the ITV image

Trajectories of tracking objects: We develop a neural network to generate dense object trajectories by still-image detectors, called tracking-network. The tracking-network learns these associations among detected objects with tracking targets from positions, sizes and distances attributes, to identify which trajectories these newly detected objects in the current frame belong to. For a certain object, there is no a detected object appearing on its trajectory in this frame. While in the last frame, an object appears on this trajectory with a high confidence and its distance is greater than 2m and less than 60m. In this case, our multi-object tracking method can propagate the detection result in the last frame to this current frame, shown in Fig. 5. The red bounding boxes in Fig. 5 are got by still-image detection. At time T-1, a car is lost due to the blurry image, while after tracking we can recover this car which is noted with a green bounding box.

IV. Experiments

A. Dataset

2017 IVFC provides 100 annotated video clips and 40 no-annotated video clips ranging from 70 frames to 100 frames per clip. We divide 70 annotated clips as training set and the rest annotated clips as validation set. The test set contains 40 video clips. The ground truth annotations of the test clips are not released publicly. In the competition, the test set only can be used once, so we mainly show the experiments tested on the validation set. In the end, the paper reports the results of top-ranked teams in 2017 IVFC.

B. Evaluation metrics

2017 IVFC uses F-measure to evaluate the detection performance according to the Precision and Recall:

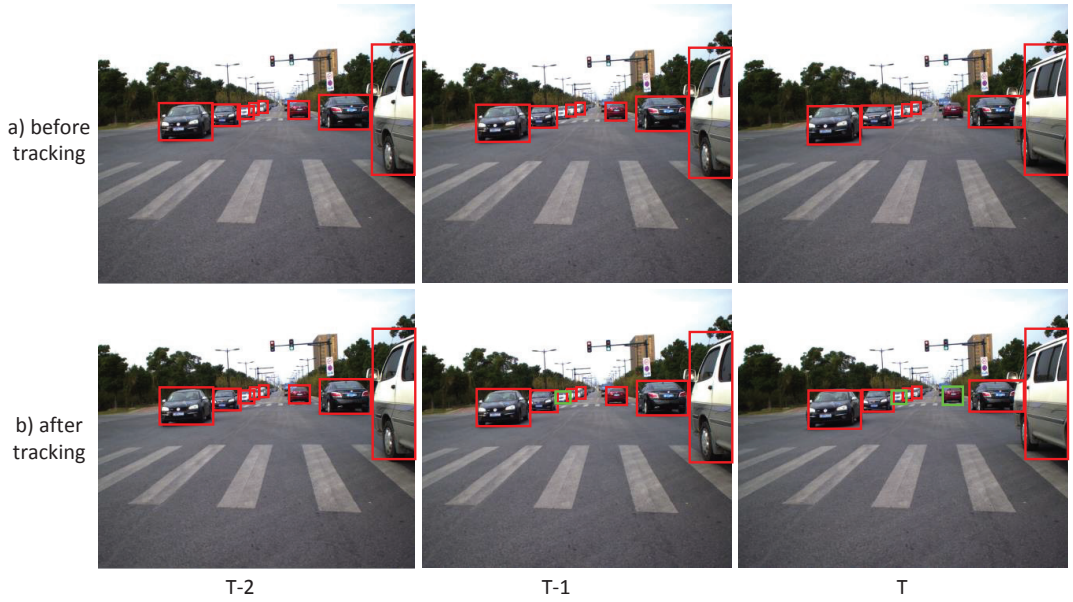

Fig. 5. The recover results by the proposed tracking method. (a) the results without tracking; (b) the results recovering by tracking

$$\text{Precision} = \frac{n_{TP}}{n_{TP} + n_{FP}} \tag{8a}$$

$$\text{Recall} = \frac{n_{TP}}{n_{TP} + n_{FN}} \tag{8b}$$

$$\text{F-measure} = \frac{2\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}, \tag{8c}$$

where m_{TP} and m_{FP} represent the number of correct detected objects and the number of wrong detected objects in all detected objects, namely true positives and false positives, respectively. m_{FN} denotes the number of missing objects, namely false negatives.

C. Network configuration

We investigate two still-image detectors: SSD [4], and Faster R-CNN [5], which are based on VGG model. To increase the speed of convergence, the model is pre-training on ImageNet DET task. TABLE I presents the detection results of SSD and Faster R-CNN. From the table, we can see that Faster R-CNN has a better performance than SSD, especially for the Recall. It may be due that SSD is not suitable for detecting small objects. For small objects, there are fewer default bounding boxes in SSD, which leads that the detection network doesn't have enough samples to be trained. Compared with SSD, Faster R-CNN contains region proposal network(PRN). PRN can choose from the default bounding boxes and retain the bounding boxes with high conference. The retaining bounding boxes have higher possibility to contain objects including small objects. So the performance of Faster R-CNN is better than SSD, and we use Faster R-CNN as the still-image detection framework in the competition.

We also do experiments with ResNet50 and ResNet101 in Faster R-CNN to investigate the impact of different number of CNN layers. The results are shown in TABLE I. The Faster R-CNN-ResNet50 and Faster R-CNN-ResNet101 mean the ResNet model [1] with 50 layers and 101 layers, respectively. From the detection results, we can see that the deeper network can extract more effective features and achieve better performance. In the following experiments, we choose the Faster R-CNN with ResNet101 as the detection framework.

D. Result

Fig. 7 shows the results of the proposed distance prediction model. For these objects within 30 m, the average error of predicting distances is less than 0.05 m, and the error becomes greater with distance. It may be due to the far

TABLE I
THE DETECTION PERFORMANCES OF DIFFERENT DETECTION FRAMEWORKS AND DIFFERENT CNN MODELS

Detection framework	Precision	Recall	F-measure
SSD-VGG	0.816	0.735	0.773
Faster R-CNN-VGG	0.854	0.775	0.813
Faster R-CNN-ResNet50	0.901	0.855	0.877
Faster R-CNN-ResNet101	**0.944**	**0.859**	**0.899**

TABLE II
THE PERFORMANCES OF AFTER TRACKING AND BEFORE TRACKING ON VALIDATION SET

Detection framework	Precision	Recall	F-measure
before tracking	0.944	0.859	0.899
after tracking	0.947	0.881	0.921

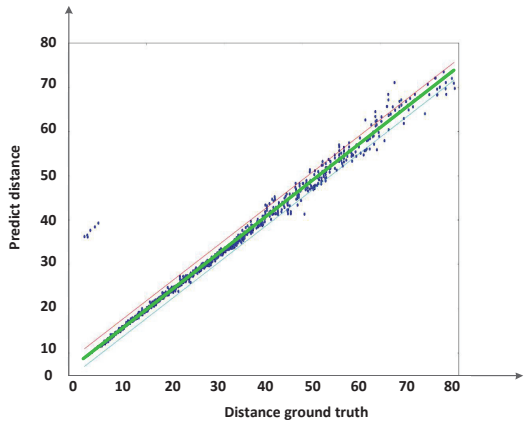

Fig. 7. The results of distance prediction. The horizontal axis is the distance ground truth and the vertical axis represents the predicting distance. The green line respects ground truth of distance, these blue nodes are the distances generated by the ITV map.

objects are blurry and small which increases the difficulty of detection. In the object detection stage, the small and blurry object will cause the object position error, which will cause the distance error in calculating distance stage.

Based on the positions, sizes, distances of objects, we can recover the detection results and achieve a better performance, shown in TABLE II. The first line of TABLE II is the result of the still-image detection, the second line is the result of the proposed video detection TBVOD. We can see that the performance of TBVOD, especially the recall has significant improvement. It may be due that in the real world traffic, some driving vehicles are blurry, and some far vehicles are small. It is difficult for these blurry and small vehicles to be detected. So it would decrease the detection recall. However, our proposed framework TBVOD considers the temporal information to recover the missing objects and get a better performance.

Our framework on video ranked 1st among these teams participated in the 2017 IVFC. The detailed results of the performance of the top 3 teams are shown in TABLE III.

V. Conclusion

In this paper, we propose a video detection framework TBVOD that incorporates the temporal information into still-image detection. A new tracking considering the positions, sizes and distances of objects learns the temporal information and association between objects among video frames to generate object trajectories. Based on these trajectories, the proposed TBVOD recovers the still-image detection results and improves the precision and recall of the video detection. The proposed TBVOD won the preceding vehicle detection task in 2017 IVFC and was very promising in practical applications.

TABLE III

THE DETECTION PERFORMANCES COMPARISON WITH OTHER TEAMS ON 2017 IVFC

Rank	Team name	Precision	Recall	F-measure
1	**CAS-IA(ours)**[1]	0.979	0.885	0.928
2	Cyber-Tiggo[2]	0.954	0.881	0.916
3	iFurture[3]	0.850	0.964	0.904

[1] Chinese Academy of Science, Institute of Automation
[2] Shanghai Jiao Tong University
[3] National University of Defense Technology

VI. Acknowledgment

We gratefully acknowledge the support of NVIDIA Corporation with the donation of the Titan Xp GPU used for this research.

References

[1] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun, "Deep residual learning for image recognition," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 770–778.

[2] Dongbin Zhao, Yaran Chen, and Le Lv, "Deep reinforcement learning with visual attention for vehicle classification," *IEEE Transactions on Cognitive and Developmental Systems,*, vol. 9, no. 4, pp. 356–367, 2016.

[3] Yaran Chen, Dongbin Zhao, Le Lv, and Chengdong Li, "A visual attention based convolutional neural network for image classification," in *12th World Congress on Intelligent Control and Automation*, June 2016, pp. 764–769.

[4] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng Yang Fu, and Alexander C. Berg, "Ssd: Single shot multibox detector," in *Proceedings of the European Conference on Computer Vision*, 2016.

[5] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," in *International Conference on Neural Information Processing Systems*, 2015, pp. 91–99.

[6] Kaiming He, Georgia Gkioxari, Piotr Dollr, and Ross Girshick, "Mask R-CNN," in *IEEE International Conference on Computer Vision*, 2017, pp. 2980–2988.

[7] Jerome Berclaz, Francois Fleuret, Engin Turetken, and Pascal Fua, "Multiple object tracking using k-shortest paths optimization," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 33, no. 9, pp. 1806–1819, 2011.

[8] Seung Hwan Bae and Kuk Jin Yoon, "Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1218–1225.

[9] Gideon Stein, Ofer Mano, and Amnon Shashua, "Vision-based acc with a single camera: bounds on range and range rate accuracy," *Information Visualization*, pp. 120–125, 2003.

[10] Yaran Chen, Dongbin Zhao, Lv Le, and Qichao Zhang, "Multi-task learning for dangerous object detection in autonomous driving ," *Information Sciences*, vol. 432, pp. 559–571, 2018.

[11] Yaran Chen, Dongbin Zhao, and Lv Le, "Multi-task learning with cartesian product-based multi-objective combination for dangerous object detection," in *International Symposium on Neural Networks*, 2017, pp. 28–35.

[12] Zia Khan, T. Balch, and F. Dellaert, "MCMC-based particle filtering for tracking a variable number of interacting targets," *IEEE Trans Pattern Anal Mach Intell*, vol. 27, no. 11, pp. 1805–19, 2005.

[13] James Munkres, "Algorithms for the assignment and transportation problems," *Journal of the Society for Industrial & Applied Mathematics*, vol. 5, no. 1, pp. 32–38, 1957.

[14] Michael D. Breitenstein, Fabian Reichlin, Bastian Leibe, Esther Koller-Meier, and Luc Van Gool, "Online multiperson tracking-by-detection from a single, uncalibrated camera," *IEEE Transactions on Pattern Analysis & Machine Intelligence*, vol. 33, no. 9, pp. 1820, 2011.