

Received February 11, 2020, accepted February 29, 2020, date of publication March 5, 2020, date of current version March 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2978539

Robust Multivehicle Tracking With Wasserstein Association Metric in Surveillance Videos

YANJIE ZENG¹, XINSHA FU¹, LEI GAO², JIAWEI ZHU³, HAIFENG LI³, (Member, IEEE), AND YUHENG LI⁴

¹School of Civil Engineering and Transportation, South China University of Technology, Guangzhou 510640, China

²Key Laboratory of Road and Traffic Engineering, Ministry of Education, Tongji University, Shanghai 410083, China

³School of Geosciences and Info-Physics, Central South University, Changsha 410083, China

⁴Technology Development Department, China Academy of Electronics and Information Technology, Beijing 100041, China

Corresponding author: Yuheng Li (yuhengli001@163.com)

This work was supported in part by the National Science Foundation of China under Grant 41571397, Grant 41501442, Grant 51778242, Grant 41871364, Grant 41501442, Grant 41861048, and Grant 41871302, and in part by the Natural Science Foundation of Hunan Province under Grant 2016JJ3144 and Grant 2016JJ2006.


ABSTRACT Vehicle tracking based on surveillance videos is of great significance in the highway traffic monitoring field. In real-world vehicle-tracking applications, partial occlusion and objects with similarly appearing distractors pose significant challenges. For addressing the above issues, we propose a robust multivehicle tracking with Wasserstein association metric (MTWAM) method. In MTWAM, we analyze the advantage of the 1-Wasserstein distance (WD-1) on partial occlusion and employ the WD-1 as the similarity criterion to measure the similarity between tracklets and detections. Moreover, for distinguishing different objects with a similar appearance, we improve the feature presentation of vehicles by developing target-specific feature sparse coding (TSSC). To demonstrate the validity of this method, we present a quantitative evaluation of both the UA-DETRAC dataset and our vehicle highway surveillance videos dataset (VecHSV). In both cases, our method achieves state-of-the-art performances.

INDEX TERMS Vehicle tracking, highway surveillance videos, Wasserstein association metric, target-specific sparse coding.

I. INTRODUCTION

Monitoring systems play an important role in the daily management of highways. Vehicle tracking based on surveillance videos, for which the goal is to provide a continuous trajectory to each target, is the main component of a monitoring system [1]–[4]. Although significant success has been achieved in detecting objects in static images, vehicle tracking based on surveillance videos remains challenging as a result of factors such as nonuniform continuous changes in the appearance of the vehicle target during movement, mutual obstructions between vehicles, motion blurring, and illumination changes [5]. The key to vehicle tracking in surveillance videos is accurately generating the object trajectories; that is, marking each object with its bounding box and class label while preserving its identity. Numerous vehicle-tracking methods based on surveillance videos formulated the task as a state estimation problem using filter-based strate-

gies, such as the Kalman filter [6]–[8] and the particle filter [9]–[11]. However, these approaches must assume a dynamic model a priori and have trouble distinguishing objects close to other targets. Therefore, these methods typically predict the states of targets in a short amount of time but do not perform well in complex scenarios. The vast majority of recent methods are based on the tracking-by-detection approach, which builds trajectories via associated detections. In general, these methods typically consist of the following components: a detection method and detection association method based on a similarity measure. The detection method finds bounding boxes enclosing instances' specific object categories and has recently provided reliable detections in complex scenes due to the development of deep learning techniques. Thus, the methods of detection association are crucial for this task. In detection association, input frame detections are linked to the short tracklets by trackers such as globally-optimal greedy (GOG) [12] and continuous energy minimization CEM [13] approaches, thereby, short tracklets are grown sequentially using frame-by-frame association up

The associate editor coordinating the review of this manuscript and approving it for publication was Mohammad Shorif Uddin .

to the current frame and eventually become long trajectories. However, these trackers [12]–[15] used in detection association exhibit a limited ability to identify vehicles with similar appearances, owing to the limited ability to characterize the intraclass fine-grained features. Moreover, partial occlusions also confuse these trackers, which can easily identify the occluded target as an occlusion target by mistake and result in tracking failure. To associate the tracklet and detection of the same object while distinguishing different objects with a similar appearance, it is crucial that the improvement of the feature presentation of vehicles not only involves semantic features but also retains detailed local features. Thus, we analyzed the characteristics of convolutional neural network (CNN) feature layers and found a suitable target feature representation manner: target-specific feature sparse coding (TSSC). In addition, to handle frequent partial occlusions, we analyzed the advantage of the 1-Wasserstein distance (WD-1) [16] for the feature similarity measure and employed the WD-1 as the similarity criterion to measure the similarity between tracklets and detections in consecutive frames.

Based on the above analysis, we propose a two-stage multivehicle tracking with Wasserstein association metric (MTWAM) method to track vehicles in highway surveillance videos. In the first stage, we generate vehicle proposals (VPs), which are the image patches for each vehicle in each surveillance video frame, by the faster region-convolutional neural network (faster R-CNN) model. Each VP contains a bounding box and confidence score but does not include the target identity. In the second stage, we develop a Wasserstein tracklet-detection association to identify and track vehicles. The Wasserstein tracklet-detection association includes VPs' fine-grained feature coding by TSSC, a 1-Wasserstein distance (WD-1) to measure the similarity of tracklet-detection pairs, and the Kuhn-Munkres algorithm to identify vehicles across neighboring frames. The contributions of this paper are as follows:

- 1) We propose a novel vehicle-tracking method called MTWAM for surveillance videos, which combines rich semantic and fine-grained features that are robust against appearance changes and possess sufficient discriminative power for similarity distractors. Compared to existing methods, our method provides state-of-the-art accuracy.

- 2) We analyze the properties of different level features in CNNs and select the features that not only involve semantic features but also retain detailed local features. Moreover, we develop TSSC for the selected features. This TSSC guarantees the ability to capture fine-grained vehicle features, while improving the computational efficiency in detection association.

- 3) We introduce the WD-1, which is used to measure the similarity of the vehicle features by TSSC across adjacent frames and is robust against partial occlusions. Experiments demonstrate that our method has discriminative power for vehicle-specific identification.

The remainder of this paper is organized as follows: Section II presents related works. Section III, part A describes

the proposal generation methods, while part B presents the Wasserstein tracklet-detection association by means of target-specific feature sparse coding, Wasserstein distance as the similarity measure and detection-tracklet association by the Kuhn-Munkres algorithm. Experimental results and comparisons are provided in section IV, and section V concludes the paper.

II. RELATED WORKS

A. VEHICLE PROPOSALS GENERATION

In recent years, object detection based on static images has achieved great successes. Numerous models based on handcrafted features [17]–[24] have been applied to static image object detection. Recent developments in state-of-the-art object detection methods are all based on deep CNNs [25]–[28]. Reference [29] proposed using a CNN during the stage of detection-proposal classification. Subsequently, [30] developed the spatial pyramid pooling (SPP) layer to overcome the fixed-size input constraint. Reference [31] improved the speed by proposing the Region of Interest (RoI) pooling layer, which shares the feature map of the entire image with each proposal. Reference [32] combined region proposals with object classification by developing the region proposal network (RPN) to accelerate proposal generation. Compared to [31], [32], which use a proposal stage and a classification stage, the method of you only look once (YOLO) [33] and its extension [34]–[36] exclude the proposal stage. In YOLO [33], the final feature map is divided into grid cells and then trained to detect objects in each cell. The YOLOv2 [34] adopts anchor boxes that are similar to faster R-CNN and uses features stacked from different layers to address object size variation. Moreover, [35] adopts default boxes for different feature layers at varying resolutions. Compared to faster R-CNN, these YOLO methods exhibit high speeds and low recall rates. In object proposal generation, faster R-CNN typically achieves high recall rates for individual frames, which is important to note because this rate is the upper bound of video object tracking performance.

B. MULTI-OBJECT TRACKING BY TARGET ASSOCIATION

Existing methods for multi-object tracking are mainly based on a tracking-by-detection approach [12]–[15], [37]–[42], which sequentially associates the detections of input frames with tracklets and builds trajectories. During this process, a robust association strategy plays an essential role.

Reference [13] formulated multitarget association by minimizing a continuous energy function. Similarly, [12] formulated the problem using a cost function and proposes a greedy algorithm that sequentially instantiates tracks using shortest path computations on a flow network. Reference [14] used motion dynamics as a cue to distinguish targets with a similar appearance. Moreover, [15] incorporated the relative motion network (RMN) model within the Bayesian filtering framework and the Kalman filter for online association.

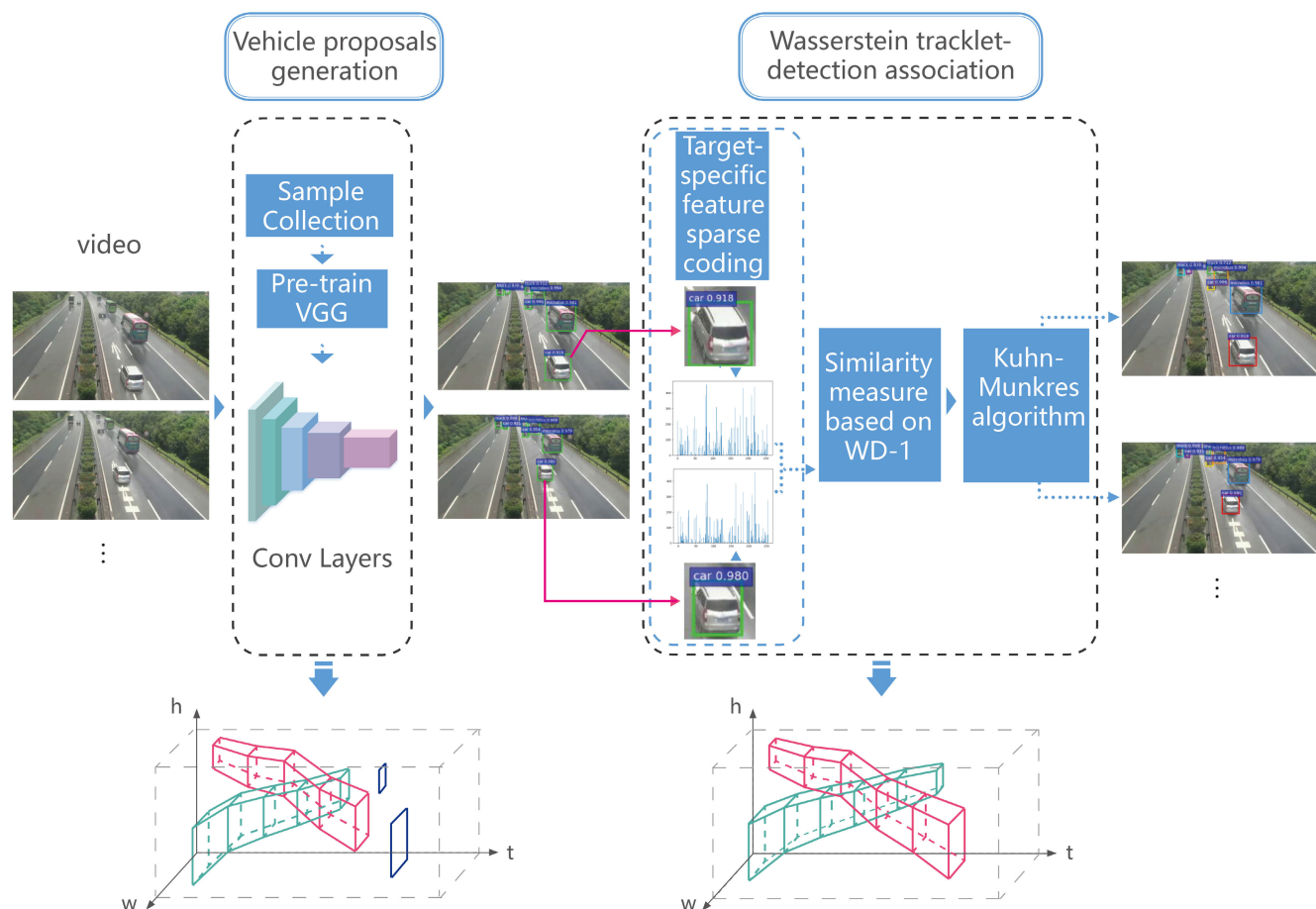


FIGURE 1. Overview of multivehicle tracking with Wasserstein association metric method. This method consists of two stages: vehicle proposals generation and Wasserstein tracklet-detection association. We adopt the faster R-CNN to generate vehicle proposals in vehicle proposals generation. In Wasserstein tracklet-detection association, we represent each detection and tracklet by means of TSSC, then we calculate each tracklet-detection pair similarity via WD-1 and introduce the Kuhn-Munkres algorithm to optimize the tracklet-detection association performance.

However, these methods do not perform as well when identifying vehicles with similar appearances due to the limitations of the object-specific appearance representations. Similar to our approach, [41] exploited a high-performance detector with a deep learning appearance feature. Reference [42] combined motion and appearance information through a deep association metric and achieved good performance. However, these do not have enough robustness on partial occlusions. In this paper, we aim to solve these challenges and propose our MTWAM method.

C. OTHER WD-BASED TRACKING METHODS

The Wasserstein distance (WD), also named Earth Mover’s Distance (EMD), is a natural distance metric for comparing two probability distributions. In the context of visual tracking, several EMD-based trackers that minimizes the EMD between the candidate and reference feature histograms have been proposed. Reference [43] employed color signatures with EMD and proposed Differential Earth Mover’s Distance (DEMD) algorithm, which was the first work using

the EMD and color signatures in visual tracking. Reference [44] combined the Gaussian Mixture Model (GMM) with the DEMD algorithm for visual object tracking. In [45], the Mean Shift tracker with EMD was proposed. In addition, [46] combined the Gyroscope information and presented gyro-aided iEMD algorithm. However, multi-object tracking has higher computational complexity than visual tracking. Therefore, WD is rarely applied to multi-object tracking due to its computation efficiency. In this paper, to improve computation efficiency, we propose target-specific sparse coding and introduce the iterative WD [47] to measure the similarity of the vehicle features.

III. METHODOLOGY

In this paper, we propose a method that decomposes the task of vehicle tracking in videos into two subproblems: target detection in each frame and target association between adjacent frames. Therefore, our method consists of two parts: vehicle proposals generation and Wasserstein tracklet-detection association, as illustrated in Fig. 1.

Vehicle Proposals Generation: We detect the vehicle in successive frames using an offline CNN model for the training set. Thereafter, the VPs are obtained, which contain bounding boxes and confidence scores but do not include target identities.

Wasserstein Tracklet-Detection Association: We associate each vehicle detection with a tracklet in the previous frame. First, we represent each detection and tracklet by means of TSSC; for simplicity, we use features of the tracklet’s terminal object to represent its features. Then, each tracklet-detection pair similarity is calculated via WD-1. To improve the accuracy of vehicle identification, we introduce the Kuhn-Munkres algorithm to optimize the tracklet-detection association performance.

A. VEHICLE PROPOSALS GENERATION

Vehicle proposals indicate the possible object locations in each frame and are crucial to object tracking performance. Considering that faster R-CNN exhibits high recall rates at each frame, in this work, we adopt the faster R-CNN framework to generate VPs. The faster R-CNN framework contains RPNs, which generate initial proposals, and a fast R-CNN detector for subcategory classification and bounding-box regression, which make detection results more reliable. In this section, we introduce the two-stage approach in faster R-CNN. In the RPN, the input image is fed into a CNN that was pretrained on a large-scale highway surveillance video dataset, as introduced in section IV part A, and forward propagated to generate feature maps. Then, a specific network is utilized over the feature maps’ output by the last convolutional layer in the CNN model to generate the initial proposal coordinates. Let $(x_i^T, y_i^T, w_i^T, h_i^T)$ denote the i -th initial box proposal at time T , where x, y, w, h represent coordinates of the box center and width and height of the box proposal, respectively.

Furthermore, in the fast R-CNN detection stage, the proposal features are RoI-pooled from the feature maps according to the initial box coordinates from the RPN and can be used for object classification and bounding-box regression. Therefore, we can obtain VPs $b = (X, Y, W, H, c)$ in each frame, where X, Y, W, H again represent the coordinates of the box center and the width and height of the box proposal, respectively, while c represents the class confidence score.

These VPs obtained from the fast R-CNN stage overlap with one another to a great extent; that is, redundancies exist because many proposals represent one object and result in inferior performance. To remove distractions, we can adopt nonmaximum suppression (NMS) based on the class confidence score c to avoid redundancy. Following the NMS process, the proposals with a class confidence score above the confidence threshold of 0.5 are selected for the association process.

B. WASSERSTEIN TRACKLET-DETECTION ASSOCIATION

Another difficulty in vehicle tracking tasks is vehicle identities with similar appearances, and the obstructions among

vehicle targets add to this challenge. We adopt the faster R-CNN algorithm, described in section III part A, to generate VPs. In this section, we propose a novel Wasserstein tracklet-detection association method, which contains the target-specific sparse feature coding, the similarities measure of tracklet-detection pairs based on WD-1 and the association optimization by the Kuhn-Munkres algorithm to achieve target identification and generate the trajectory.

We make the following observations: Although the target appearance exhibits nonuniform changes during the monitoring process, it interferes with factors such as partial occlusions, illumination changes, and motion blurring. The CNN model trained on the large-scale dataset can capture the target invariant features, which means that even when interfered with the above various factors, the similarity of features of the same target between adjacent frames is greater than that of different targets. Specifically, each target is represented by sparse coding in the CNN feature space and transforms the target-specific feature into the distribution. Thereafter, the associations are generated based on the distribution similarity measure across adjacent frames, as illustrated in Fig. 2. In view of this observation, we develop the Wasserstein tracklet-detection association method. We also present the pseudocode of Wasserstein tracklet-detection association.

1) WASSERSTEIN DISTANCE AS SIMILARITY MEASURE

As a measure of calculating similarity between tracklets and detections, WD-1 is robust to partial occlusion. In this chapter, we introduce WD-1 as a target-specific feature similarity measure and transform the feature similarity measure into a linear programming problem.

In this section, we briefly introduce the WD-1 as shown in Fig. 3. As mentioned above, we use features of the tracklet’s terminal object to represent its features. Given v_j^{T-1} representing the feature of vehicle j in frame $T - 1$ and v_i^T representing the feature of vehicle i in frame T , we have one-dimensional vectors for both of them. Taking the WD-1 measure, the similarity between v_j^{T-1} and v_i^T is defined by (1) to (5).

$$v_j^{T-1} = \{l_u\}_{u=1,\dots,k} \tag{1}$$

$$v_i^T = \{q_\varepsilon\}_{\varepsilon=1,\dots,k} \tag{2}$$

$$D^*(v_j^{T-1}, v_i^T) \triangleq \min_{f_{u\varepsilon}} \left(\sum_{u=1}^k \sum_{\varepsilon=1}^k d_{u\varepsilon} f_{u\varepsilon}(l_u, q_\varepsilon) \right) \tag{3}$$

subject to

$$\sum_{u=1}^k \sum_{\varepsilon=1}^k f_{u\varepsilon}(l_u, q_\varepsilon) = 1 \tag{4}$$

$$f_{u\varepsilon}(l_u, q_\varepsilon) \geq 0, \quad 1 \leq u \leq k, \quad 1 \leq \varepsilon \leq k \tag{5}$$

where D^* is the optimal solution to this linear programming problem by finding a flow $F = [f_{u\varepsilon}]$. $f_{u\varepsilon}(l_u, q_\varepsilon)$ is the flow from the u -th bin of v_j^{T-1} to the ε -th bin of v_i^T , and $d_{u\varepsilon}$ is the ground distance between the u -th bin of v_j^{T-1} to the ε -th bin

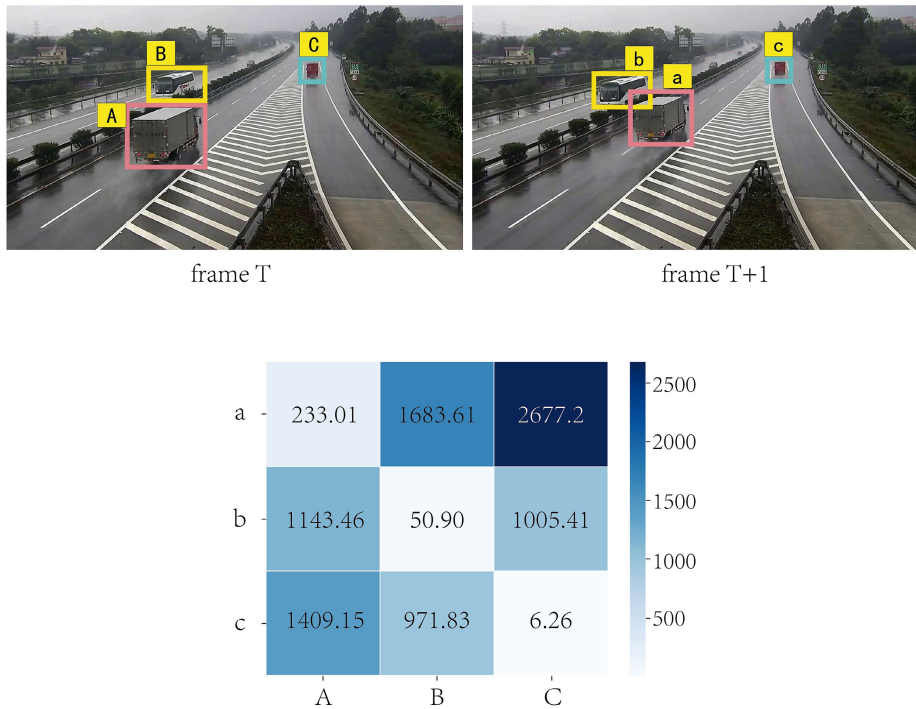


FIGURE 2. A, B, C are vehicles detected in frame T, and a, b, c are vehicles detected in the adjacent frame T+1. We compute the Wasserstein distance for each pairing and select the one with minimum WD-1 value in total. In this case, (A-a, B-b, C-c) is the successful association we generate.

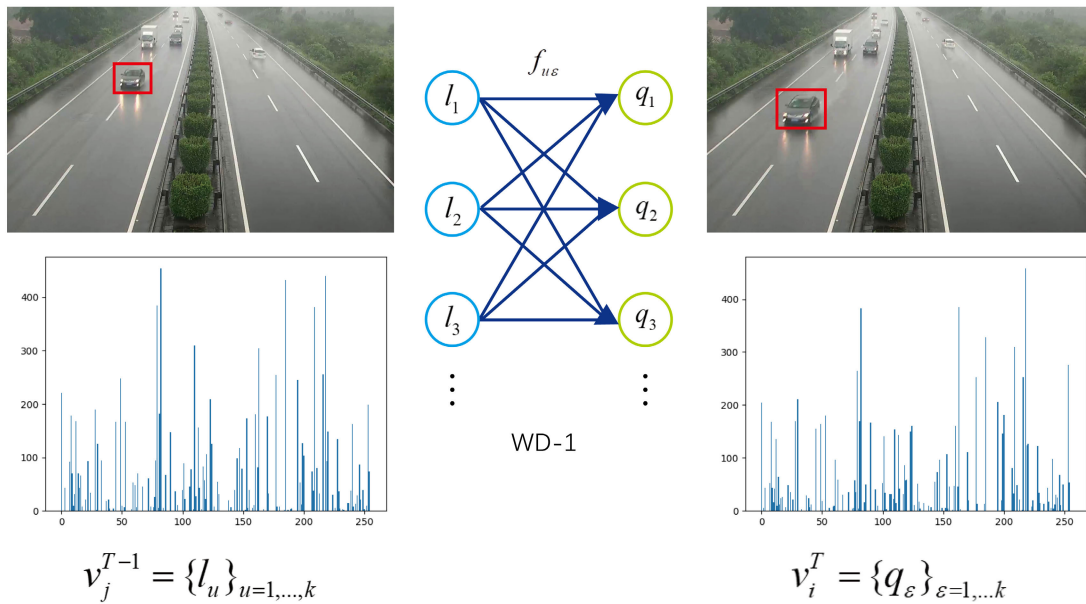


FIGURE 3. WD-1 comparison of two target-specific feature vectors.

of v_i^T . Here, we take a similar example to explain flow $f_{u\varepsilon}(l_u, q_\varepsilon)$ and ground distance $d_{u\varepsilon}$: given two distributions, one can be seen as a mass of earth properly spread in space, the other as a collection of holes in that same space. The WD-1 measures the least amount of work needed to fill the holes with earth; a unit of work corresponds to transporting

a unit of earth by a unit of ground distance. Here, the flow $f_{u\varepsilon}(l_u, q_\varepsilon)$ represents the volume of earth from the earth mass to the holes and the ground distance $d_{u\varepsilon}$ represents the work of transporting a unit of earth from the earth mass to the holes. Thus, WD-1 is known as the earth mover’s distance. Once we have found the optimal solution flow and the optimal flow F,

the WD-1 is defined by (6).

$$WD(v_j^{T-1}, v_i^T) = \frac{\sum_{u=1}^k \sum_{\varepsilon=1}^k d_{u\varepsilon} f_{u\varepsilon}(l_u, q_\varepsilon)}{\sum_{u=1}^k \sum_{\varepsilon=1}^k f_{u\varepsilon}(l_u, q_\varepsilon)} \quad (6)$$

Robustness to Partial Occlusion: Compared with the original appearance, when the target is partially occluded, its appearance features will cause interference and will have less similarity with each other. When this similarity is in the L2 distance or the L1 distance, the distances between the target features are linearly combined, while the Wasserstein distance provides a natural way to conduct set-to-set comparisons. For each element in a set, it considers only the distances with nearest neighbors in the other set. In other words, the more similar the distribution is, the more the difference will be minimized, while the less similar the distribution is, the less the property can be fully utilized. The WD-1 helps the system automatically discover a proper alignment between the part that is not obscured and the original appearance and transform the distribution of the unobscured part into the distribution of the original appearance using a minimal “cost”. Thus, a similarity measure based on the WD-1 can minimize the noise from partial occlusion and discriminate targets.

Additionally, we indicate that WD-1 is robust with respect to the similarity measure by two experiments in section IV part G.

Computation Complexity: The efficiency of the Wasserstein distance calculation is an important issue. The time complexity of the traditional algorithms for solving the Wasserstein distance is $O(n^3 \log(n))$, which is higher than other traditional measures. To solve the Wasserstein distance faster, we adopt a regularized Wasserstein distance [47] using an iterative algorithm, which is defined by (7):

$$D_S(v_j^{T-1}, v_i^T) = \min_{P \in \Pi(v_j^{T-1}, v_i^T)} \langle P, M \rangle_F - \frac{1}{\lambda} h(P) \quad (7)$$

where $h(P) = \sum_{u,\varepsilon} P_{u,\varepsilon} \log(P_{u,\varepsilon})$ is information entropy of P . The parameter λ determines the trade-off between the two terms. The entropy-regularized Wasserstein distance is also called the Sinkhorn distance and can be solved by iterating the Sinkhorn update, whose computational complexity is only $O(n^2)$.

2) TARGET-SPECIFIC FEATURE SPARSE CODING

In the previous section, we introduced WD-1 as a similarity measure, which exhibits strong robustness against partial occlusion. However, the main problem of WD-1 is its computational complexity. To overcome this limitation, we propose the target-specific feature sparse coding method based on the CNN feature layer, which aims to maintain the fine-grained features of vehicles for identifying a vehicle while improving the WD-1 computational efficiency. The method for target-specific feature sparse coding is as follows.

Given the proposals of bounding-box coordinates (X, Y, W, H) , according to (8) and (9), the bounding-box coordinates are mapped to the layer in the CNN, and the layer selection process takes place, as described in section IV part F.

$$(x, y, w, h) = (X/\theta, Y/\theta, W/\theta, H/\theta) \quad (8)$$

$$\theta = \Pi \beta_j \quad (9)$$

where β_j represents the strides of the layer in the CNN and θ is the product of all previous strides. Moreover, (X, Y, W, H) denotes the coordinates of the center point of the i th bounding box and the width and height of the bounding box in the input image, respectively. Likewise, (x, y, w, h) denotes the same properties in the selected feature layer. It should be emphasized that numerous channels exist in the selected feature layer. Specifically, each feature layer of the CNN contains multiple channels. For each channel in the selected feature layer, we obtain the matrix centered at coordinates (x, y) and with a width w and height h when projecting the proposed coordinates onto it. Then, we calculate the response value of each channel according to (10) and then concatenate the response value of each channel as the target feature. Moreover, we set the response values below the threshold to zero to make the target feature sparse owing to the significantly smaller response value in the channels. For all experiments, we set this threshold to 0.

$$l_u = \sum_{j=y-h/2}^{y+h/2} \sum_{i=x-w/2}^{x+w/2} \eta_{ij} \quad (10)$$

$$v = (l_1, l_2, l_3 \dots l_k)_{u=1, \dots, k} \quad (11)$$

where η_{ij} represents the value of coordinate (i, j) in a channel of the feature map, and l_u represents the response value of the u -th channel. Moreover, v represents the target-specific sparse feature concatenated by the response value of k channels.

3) DETECTION-TRACKLET OPTIMAL ASSOCIATION BY THE KUHN-MUNKRES ALGORITHM

The last section described the TSSC for coding each target feature and the similarity measure based on WD-1 of detection-tracklet pairs in two consecutive frames. Here, based on the results of the similarity measure, we introduce the Kuhn-Munkres algorithm [48] to optimize the detection-tracklet association. From the perspective of the similarity measure of all vehicle targets in two adjacent frames, it is intuitive that if all targets are correctly matched with their tracklets in a specific time interval, then the total distance of detection-tracklet pairs is at a minimum. Moreover, each target belongs to at most one tracklet; therefore, this detection-tracklet association problem is a typical assignment problem, and we introduce the Kuhn-Munkres algorithm to solve it. The Kuhn-Munkres algorithm is a combinatorial optimization algorithm that solves the assignment problem in polynomial time, and it is described as follows:

We assume here that d detections and t tracklets are given at frame T and We define S as the non-negative distance

matrix where each element of S represents the WD-1 distance between each detection-tracklet pair, defined as (12) (13):

$$S = [s_{ij}]_{d \times t} \quad (12)$$

$$s_{ij} = WD(v_i^T, v_j^{T-1}) \quad (13)$$

The element s_{ij} represents the WD-1 distance from the i -th detection target at frame T to the j -th tracklet that is the identified target at frame $T-1$, which is computed by (7). The calculation process of the Kuhn-Munkres algorithm is shown as follows:

step 0: Input the $d \times t$ matrix called the cost matrix.

step 1: For each row of the matrix, find the smallest element and subtract it from every element in its row. Go to Step 2.

step 2: Find a zero Z in the resulting matrix. If there is no starred zero in its row or column, star Z . Repeat for each element in the matrix. Go to Step 3.

step 3: Cover each column containing a starred zero. If $\min(d, t)$ columns are covered, the starred zeros describe a complete set of unique assignments. In this case, Go to DONE; otherwise, Go to Step 4.

step 4: Find a noncovered zero and prime it. If there is no starred zero in the row containing this primed zero, Go to Step 5. Otherwise, cover this row and uncover the column containing the starred zero. Continue in this manner until there are no uncovered zeros left. Save the smallest uncovered value and Go to Step 6.

step 5: Construct a series of alternating primed and starred zeros as follows. Let Z_0 represent the uncovered primed zero found in Step 4. Let Z_1 denote the starred zero in the column of Z_0 (if any). Let Z_2 denote the primed zero in the row of Z_1 (there will always be one). Continue until the series terminates at a primed zero that has no starred zero in its column. Unstar each starred zero of the series, star each primed zero of the series, erase all primes and uncover every line in the matrix. Return to Step 3.

step 6: Add the value found in Step 4 to every element of each covered row, and subtract it from every element of each uncovered column. Return to Step 4 without altering any stars, primes, or covered lines.

DONE: Assignment pairs are indicated by the positions of the starred zeros in the cost matrix. If s_{ij} is a starred zero, then the element associated with row i is assigned to the element associated with column j .

IV. EXPERIMENTS AND ANALYSIS

A. DATASETS

The appearance of vehicles in highway surveillance videos is affected by illumination changes, viewing angle variations, motion blurring, and partial occlusion. Considering that the highway surveillance video dataset is very sparse and few researchers have thoroughly considered the above factors, we collected surveillance data from many highways in Guangdong Province, China, produced a new highway surveillance video dataset. In addition to the monitoring scene of the various situations mentioned previously, the dataset,

Algorithm 1 Pseudocode of Wasserstein Tracklet-Detection Association

Input:

The t tracklets of previous T frames, which include each target's bounding box b_i^{T-1} , class scores c_i^{T-1} , trackID and its feature vector v_i^{T-1} . The d detections at frame T , which include its bounding box b_i^T and class scores c_i^T .

Output:

TrackID of each detection at frame T .

- 1: **begin**
- 2: **for** $i = 1$ to d **do**
- 3: calculate each detection's feature vectors v_i^T by TSSC.
- 4: **for** $j = 1$ to t **do**
- 5: Calculate the distance s_{ij} between i -th detection feature vectors v_i^T and j -th tracklet feature vector v_j^{T-1} by WD.
Add this distance s_{ij} into the distance matrix S .
- 6: **end for**
- 7: **end for**
- 8: Targets association with by Kuhn-Munkres algorithm based on distance matrix S .
- 9: **return** Each trackID of detection at frame T .

called the VecHSV dataset, with over 90,000 images and annotations, and a total of 0.7 million bounding boxes of 5,370 vehicles are labeled, including special scenes such as tunnels and interchanges and 80 videos that are selected from over 20 hours of image sequences at 18 different locations. Moreover, these videos are recorded at 25 frames per seconds (fps) with the JPG image resolution of 1920×1080 pixels and the mean length is about 1,150 frames. Certain sample images are illustrated in Fig. 4.

Moreover, we also experiment with our method on the UA-DETRAC benchmark [49] that consists of urban traffic surveillance with annotated vehicle tracks.

B. EVALUATION METRIC

We adopt the performance evaluation metrics of CLEAR MOT [50] for vehicle tracking, which are provided in the vast majority of the literature, including the multiple object tracking accuracy (MOTA), the number of ID switches (IDs), the percentages of mostly tracked (MT) and mostly lost (ML) out of all the tracks, the number of fragments (FM) and the multiple object tracking precision (MOTP). The IDs metric describes the number of times that the matched identity of a tracked trajectory changes, and FM is the number of times that trajectories are disconnected. Both the IDs and FM metrics reflect the accuracy of tracked trajectories. The ML metric measures the percentage of trajectories lost more than 80% of the time based on the ground truth. Similarly, the MT metric measures the percentage of tracked trajectories more than 80% of the time based on the ground truth. The MOTA

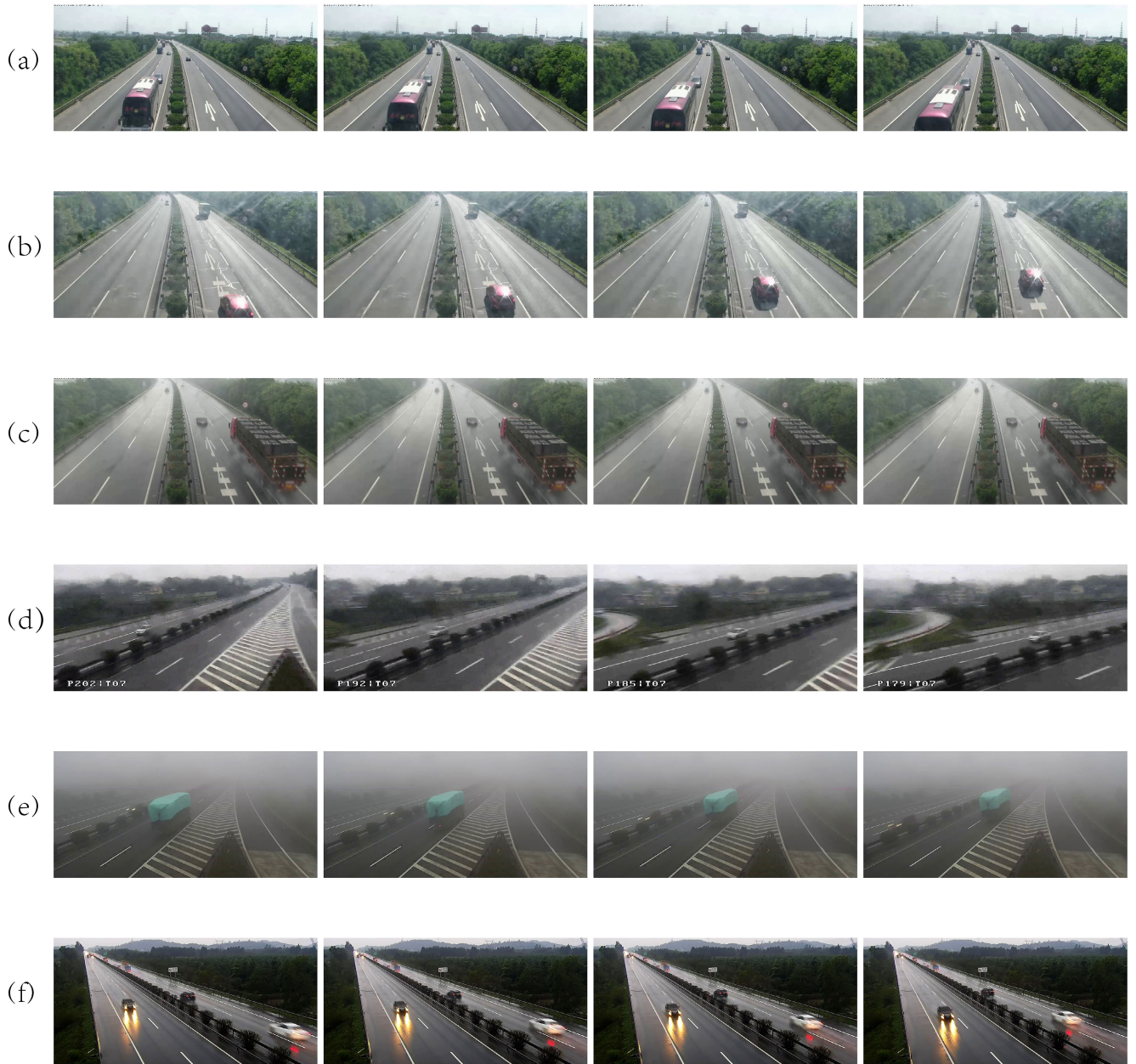


FIGURE 4. Certain scenarios in the VecHSV dataset, including (a) Partial occlusion; (b) Illumination changes; (c) Rainy days; (d) Camera motion; (e) Foggy day; (f) Dusk.

metric is defined as (14):

$$MOTA = 100 \times \left(1 - \frac{\sum_t (fn_t + fp_t + mm_t)}{\sum_t gt_t} \right) \quad (14)$$

where fn_t , fp_t , mm_t and gt_t are false negatives, false positives, mismatches and the ground truth at frame t , respectively.

Additionally, the MOTP metric is defined by (15):

$$MOTP = 100 \times \frac{\sum_{i,t} d_t^i}{\sum_t c_t} \quad (15)$$

where c_t is the number of matches found for time t , and d_t^i is the distance between detections and their ground truth.

C. QUANTITATIVE RESULTS

We apply our method to the VecHSV dataset and UAETRAC dataset with an NVIDIA GTX1080 GPU and an Intel Xeon E3-1230 CPU. For the experiment on the VecHSV dataset, in the training phase, we use 60 sequences to train a faster R-CNN for detecting vehicles, and we follow [32] to set the hyper-parameterstrain and train for 70,000 iterations using stochastic gradient descent (SGD) for optimization. Specifically, the models are initialized using weights pre-trained on ImageNet [51]. We finetune the network with a

TABLE 1. Evaluation of the vecsv dataset for different methods.

Methods	MOTA	IDs	MT	ML	FM	MOTP
CEM [13]	20.1	471	8.4	52.6	632	78.3
GOG [12]	53.4	851	47.3	7.5	1023	68.1
IHTLS [14]	66.7	741	69.1	3.2	1157	81.8
RMOT [15]	68.2	687	46.7	4.4	981	77.2
POI [41]	72.6	715	72.9	4.8	1001	84.7
Deep SORT [42]	71.4	691	71.7	4	904	84.6
Ours (L2)	70.2	703	72.5	4.1	873	83.7
Ours (WD))	72.4	670	73.7	3	845	85.3

learning rate of 0.001 for 50,000 iterations and then reduce the learning rate to 0.0001 for another 20,000 iterations. In addition, batch size is set to 8 and momentum of 0.9 and a weight decay of 0.0005 is used in our experiments. In the test phase, similar to other tracking-by-detection methods, we first use our trained faster R-CNN model after the training phase with the original nonmaximum suppression included to obtain the vehicle bounding boxes and class scores in each frame. Then, we sequentially build trajectories based on the frame-by-frame association with the present frame in the Wasserstein tracklet-detection association. Moreover, we reproduce the six state-of-the-art methods on VecHSV dataset and all results are presented in TABLE 1.

According to the results, our approach, regardless of whether using the L2 distance to measure the dissimilarity of each detection-tracklet pair or WD-1 distances, is a strong competitor to other online tracking methods in terms of MOTA. Moreover, we surpass the online state-of-the-art methods in terms of the number of ID switches. Additionally, our method achieves better performance on the MT score and the ML score, which also implies the effectiveness of our methods for maintaining consistent trajectories. At the same time, we also have a higher MOTP score, which proves that our method can precisely estimate target positions. Note that in our method, as a dissimilarity measure of detection-tracklet pairs in a consecutive frame, the WD-1 distance surpasses the L2 distance with a higher MOTA score and fewer IDs. This result also demonstrates the conclusion in the experiment of section G that the WD-1 distance helps to associate detection when targets are partially occluded.

We also apply our method to the UA-DETRAC dataset. The training strategy is the same as the training on the VecHSV dataset, and we use 52 sequences in the training set to train a faster R-CNN for detecting vehicles, and the other eight for testing1 (including sequence numbers: 39781, 40152, 40181, 40752, 41063, 41073, 63521, and 63525). We also reproduce the six state-of-the-art methods on UA-DETRAC dataset, all results are presented in TABLE 2.

In this experiment, our method is still competitive. In particular, we achieve the second best performance in terms of MOTA with 0.2 point below the best performing one and the fewest number of ID switches and fragments of all online methods. In addition, we achieve a much lower percentage of ML as well as a high percentage of MT. At the same time,

TABLE 2. Evaluation of the detrac dataset for different methods.

Methods	MOTA	IDs	MT	ML	FM	MOTP
CEM [13]	15.7	65	4.8	57.8	100	74.8
GOG [12]	49.1	328	44.9	9.6	472	66
IHTLS [14]	62.6	205	63.4	3.8	563	82.2
RMOT [15]	62.6	132	42.1	6.5	174	75.7
POI [41]	69.2	194	67.4	5.2	651	86.1
Deep SORT [42]	65.4	170	65.9	4.7	593	86.3
Our(L2)	64.1	175	65.7	4.8	527	86.2
Our(WD))	68.5	151	70.2	3.1	498	86.5

TABLE 3. Runtime record for different methods.

Methods	Runtime
CEM [13]	0.22 s
GOG [12]	0.002 s
IHTLS [14]	0.05 s
RMOT [15]	0.03 s
POI [41]	0.14 s
Deep SORT [42]	0.02 s
Our(L2)	0.24 s
Our(WD))	0.31 s

our approach also performs slightly better than other methods on the MOTP score.

D. COMPUTATIONAL COMPLEXITY ANALYSIS

Here, we take one tracklet-detection association operation as the unit for computational complexity. We assume that in a specific time interval with d detections and t tracklets, both detections and tracklet are represented as a sparse feature coding, which is a $1 \times k$ vector. The computational complexity is

$$O(dtk^2) + O(n^3) \quad (16)$$

where the first term computes the Wasserstein distance of each tracklet-detection pair. As mentioned in section III part B.1, the regularized Wasserstein distance is employed to accelerate the computational efficiency. The second term indicates the computational complexity for the assignment of detections-to-tracklets by the Kuhn-Munkres algorithm.

The runtime results of each method are presented in TABLE 3. Our approach achieves a runtime record of approximately 0.3 s per frame, which is slower than the other methods. However, there is space for improvement, e.g., by accelerating the computation of the WD-1. We will focus on these points in our future work to promote both accuracy and speed.

E. QUALITATIVE RESULTS

A portion of the qualitative results is illustrated in Fig. 5. We can draw the following conclusions. These bounding boxes proposed by the faster R-CNN exhibit no obvious false positives, and when facing partial occlusion, vehicle tracks are consistent across successive frames, benefitting from the Wasserstein tracklet-detection association.

F. FEATURE SELECTION

It is well known that CNNs offer powerful capabilities for learning feature representations, and these feature representations are the key to helping CNNs yield remarkable results. In [52], it was found that different CNN layers

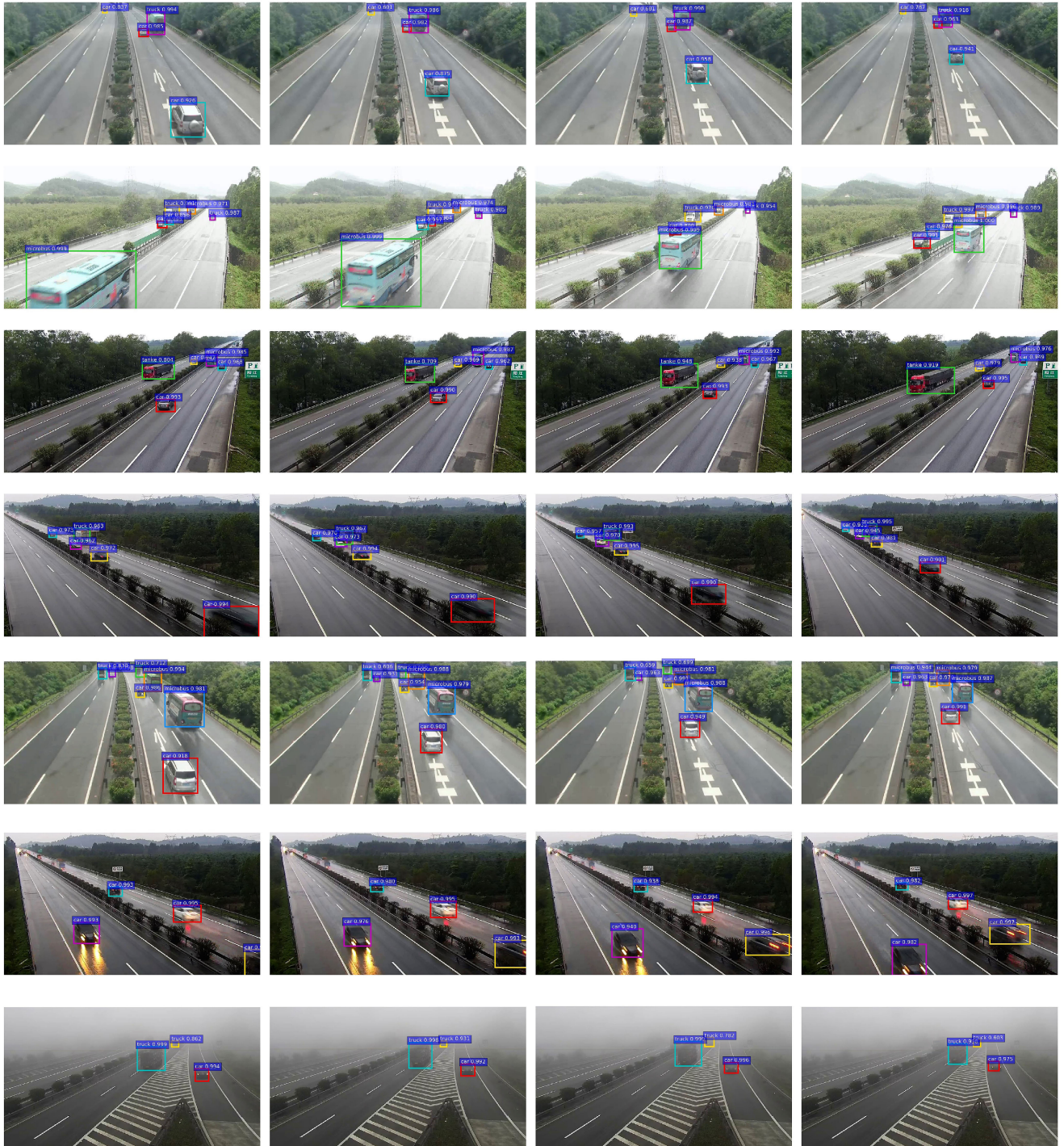


FIGURE 5. Qualitative results for the highway monitoring video datasets. Bounding boxes of the same color represent the same vehicle across successive frames.

encode different features. The higher layers capture more semantic information and serve as a category detector, while the lower layers encode more fine-grained features and can better discriminate the target from distractors with similar appearances. This observation is verified using the highway surveillance videos dataset, as illustrated in Fig. 6. Therefore, it is important to select CNN feature layers that not only

capture fine-grained features but also encode less noisy and irrelevant information for characterizing target vehicles.

In this section, we design the experiments for selecting the feature layer in the Visual Geometry Group (VGG) [25] model. First, we prepare one subset from the VecHSV and UA-DETRAC datasets that has 30 video clips that contain some similarity appearance samples. Then, we build upon

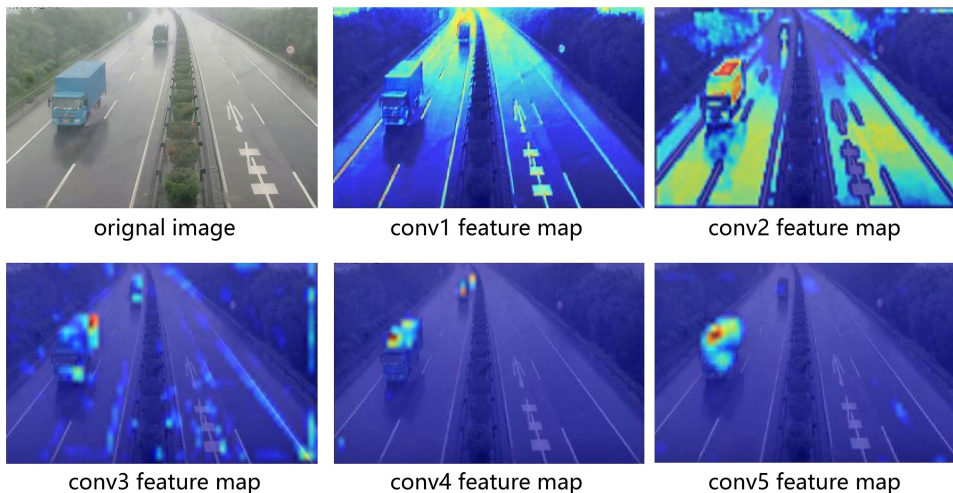


FIGURE 6. Heat maps using a feature map of the VGG network.

TABLE 4. Target-specific affinity metric on different feature maps with.

Feature layer	IDs
Conv3	102
Conv4	86
Conv5	97

the target-specific features in the different feature layers to reduce the interference of target detection errors. The target features are encoded in different feature layers according to the ground truth coordinates instead of the proposal coordinates generated by the faster R-CNN. Moreover, we adopt the WD-1 as a similarity measure and associate detections to its tracklets by the Kuhn-Munkres algorithm.

In this experiment, we choose the number of ID switches (IDs) as the evaluation metric because it is sensitive to ID mismatches, and the results are provided in TABLE 4.

In the VGG_M_1024 model, the performance of the target-specific feature coding by the conv4 layers, compared to the conv3 and conv5 feature layers, is more sensitive to discriminating intraclass vehicles with similar appearances, which could be attributed to the noisier and more irrelevant information contained in the conv3 layer feature maps and the lack of fine-grained information in the conv5 layer feature maps.

G. ROBUSTNESS ANALYSIS OF THE WASSERSTEIN DISTANCE

To measure the similarity between detections and tracklets across adjacent frames, it is crucial to select suitable similarity measures. In this section, we verify the robustness of the WD-1 by means of two experiments and compare it with traditional similarity measures. We also use features of the tracklet’s terminal object to represent its features in the similarity measure process.

1) RELATIVE DISTANCE COMPARISON FOR TARGET PAIR

A suitable similarity measure should be capable of minimizing the noise effects and maintaining a large “margin” between the positive target pair and negative target pair. In this experiment, we experiment on ten video sequences, and each target feature is coded by the TSSC method. Then, we called this margin the relative distance, which is defined as follows.

We assume these are the h target pairs in two adjacent frames. Given a positive similarity cost r_i^p computed between a positive target pair (a pair of detection responses belonging to the same vehicle i) and a negative similarity cost r_{ij}^n computed between a negative target pair (a pair of detection responses belonging to different vehicles; one belongs to vehicle i at frame T , the other one belongs to vehicle j at frame $T - 1$), we have the following as shown in (17) to (19):

$$r_i^p(M, i) = M(v_i^T, v_i^{T-1}) \tag{17}$$

$$r_{ij}^n(M, i, j) = M(v_i^T, v_j^{T-1}), i \neq j \tag{18}$$

$$\tilde{r}_i^n(M, i) = \frac{1}{h-1} \sum_{j=1, j \neq i}^h r_{ij}^n(M, i, j) \tag{19}$$

where $M(\cdot)$ is the unified presentation of the similarity cost between feature vectors, for example, the computation by (7) when using WD-1. v_i^T and v_i^{T-1} are the feature vectors in two adjacent frames from the same vehicle; otherwise, v_i^T and v_j^{T-1} represent different vehicles when i is not equal to j . \tilde{r}_i^n represents the vehicle i ’s mean value of all negative similarity costs.

$$RD(M, i) = -\log(r_i^p / \tilde{r}_i^n) \tag{20}$$

$$M^* = \arg \max_M RD(M, i) \tag{21}$$

where $RD(M, i)$ represents the relative distance of vehicle i between the positive similarity cost and the mean value of

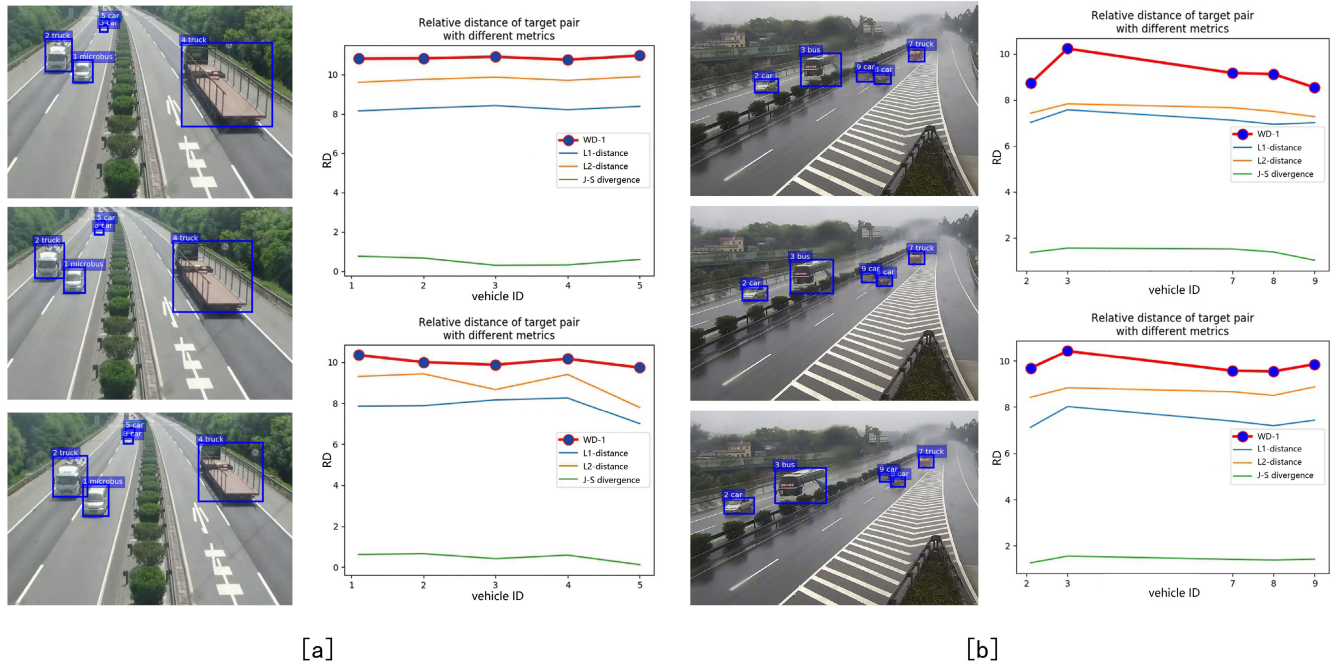


FIGURE 7. Relative distance of target pairs with different metrics. In each subgraph in this figure, the x-axes represent the vehicle ID, while the y-axis is the relative distance of this vehicle in a particular measure. From each subgraph, we can see that the relative difference in the WD-1 is the largest, followed by the L2 and L1 distances, with the smallest difference under the J-S divergence. This result implies that the WD-1 is more suitable for measuring the target-specific similarity across adjacent frames.

all negative similarity costs under measure M . We select the similarity measure by maximizing the relative distance.

As illustrated in Fig. 7, we compare the WD-1 with the L1 and L2 distances and the Jensen-Shannon (J-S) divergence [53], which are widespread uses of the distribution similarity measure. When using the WD-1 as a similarity measure, the relative distance $RD(i)$ is at a maximum, which means that the WD-1 has a more powerful robustness in maintaining discrimination across consecutive frames.

2) WASSERSTEIN DISTANCE PERFORMANCE IN PARTIAL OCCLUSIONS

Partial occlusion is a major problem in surveillance because the appearance information of the occluded target is interrupted. In the Methods section, we have described that the important advantage of the WD-1 over other measures on target-specific feature similarity is its robustness to partial occlusion; this conclusion is proved by the experiments in this section. We compare the robustness of various measures on the similarity measure for a target that is partially occluded before and after.

We define a given continuous frame in which one car obstructs another. Specifically, vehicle B is gradually occluded by vehicle A. Here, o^p represents the distance between the feature of vehicle B in frame T and the feature of vehicle B in frame $T - 1$, while o^n represents the distance between the feature of vehicle B in frame T and the feature of vehicle A in frame $T - 1$, both of which are a function

of the occlusion ratio or . We select thirty sequences that have partial occlusions and use ΔR to represent the average relative distance between these two distances, which is similar to but not the same as the $RD(M, i)$ defined in the last experiment and is intended to represent the performance in discriminating against occlusion target A, as per the following (22) to (24):

$$o^p(or) = M(v_B^T, v_B^{T-1}) \quad (22)$$

$$o^n(or) = M(v_B^T, v_A^{T-1}) \quad (23)$$

$$\Delta R(or) = \frac{1}{n_{se}} \sum_{i=1}^{n_{se}} \frac{o_i^n - o_i^p}{o_i^n} \quad (24)$$

where $M(\cdot)$ is also a unified presentation of the similarity cost between feature vectors, which is the same as that in the last experiment, n_{se} is the number of video sequences that have partial occlusions. As the occlusion ratio gradually increases, o^p increases and o^n decreases; thus, ΔR gradually decreases, which means that the discrimination between targets becomes weaker. As illustrated in Fig. 8, compared to other measures, the WD-1 can effectively distinguish the occlusion target from the occluded target in various contexts. For example, when the occlusion ratio is less than 0.2, the WD-1 does not change substantially, but the divergences of the L1 and L2 distances and the J-S divergence decrease significantly. When the occlusion rate reaches 0.5, the WD-1 maintains a relatively strong performance, while the other measures, particularly the J-S divergence, lose the ability to distinguish between the occlusion target and the occluded target.

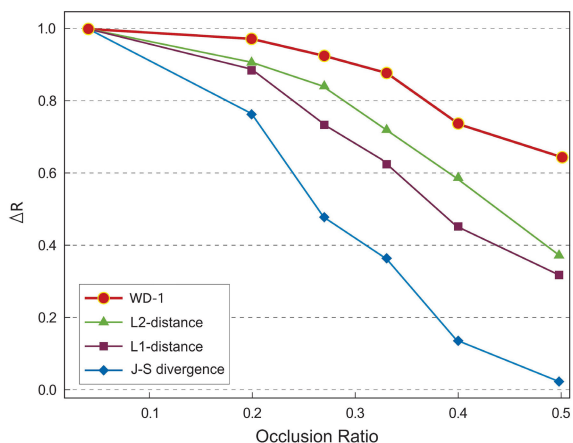


FIGURE 8. Average relative distance with different metrics under occlusion.

V. CONCLUSION

In this work, we propose a robust deep learning method for multivehicle tracking in surveillance videos. The method combines a popular still-image detector with Wasserstein tracklet-detection association. Due to the Wasserstein tracklet-detection association, our method can track targets with a similar appearance and achieves robustness with respect to partial occlusion. Experiments demonstrate the effectiveness of our proposed method. In the future, we will combine our work with graph convolutional network to analysis and predict traffic flow [54].

REFERENCES

- [1] S. Sivaraman and M. M. Trivedi, "Looking at vehicles on the road: A survey of vision-based vehicle detection, tracking, and behavior analysis," *IEEE Trans. Intell. Transp. Syst.*, vol. 14, no. 4, pp. 1773–1795, Dec. 2013.
- [2] S. Gupte, O. Masoud, R. F. K. Martin, and N. P. Papanikolopoulos, "Detection and classification of vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 3, no. 1, pp. 37–47, Mar. 2002.
- [3] S.-H. Yu, J.-W. Hsieh, Y.-S. Chen, and W.-F. Hu, "Automatic traffic surveillance system for vehicle tracking and classification," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 2, pp. 175–187, Jun. 2006.
- [4] A. Jazayeri, H. Cai, J. Y. Zheng, and M. Tuceyan, "Vehicle detection and tracking in car video based on motion model," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 2, pp. 583–595, Jun. 2011.
- [5] D. Acunzo, Y. Zhu, B. Xie, and G. Baratoff, "Context-adaptive approach for vehicle detection under varying lighting conditions," in *Proc. IEEE Intell. Transp. Syst. Conf.*, Sep. 2007, pp. 654–660.
- [6] J. Black, T. Ellis, and P. Rosin, "Multi view image surveillance and tracking," in *Proc. Workshop Motion Video Comput.*, Dec. 2002, pp. 169–174.
- [7] L. Marcenaro, M. Ferrari, L. Marchesotti, and C. S. Regazzoni, "Multiple object tracking under heavy occlusions by using Kalman filters based on shape matching," in *Proc. Int. Conf. Image Process.*, vol. 3, Sep. 2002, p. 2.
- [8] W. F. Leven and A. D. Lanterman, "Unscented Kalman filters for multiple target tracking with symmetric measurement equations," *IEEE Trans. Autom. Control*, vol. 54, no. 2, pp. 370–375, Feb. 2009.
- [9] J. Giebel, D. M. Gavrilu, and C. Schnörr, "A Bayesian framework for multi-cue 3D object tracking," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2004, pp. 241–252.
- [10] K. Okuma, A. Taleghani, N. De Freitas, J. J. Little, and D. G. Lowe, "A boosted particle filter: Multitarget detection and tracking," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2004, pp. 28–39.
- [11] Vermaak, Doucet, and Perez, "Maintaining multimodality through mixture tracking," in *Proc. 9th IEEE Int. Conf. Comput. Vis.*, Oct. 2003, p. 1110.
- [12] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Conf.*, Jun. 2011, pp. 1201–1208.
- [13] A. Andriyenko and K. Schindler, "Multi-target tracking by continuous energy minimization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2011, pp. 1265–1272.
- [14] C. Dicle, O. I. Camps, and M. Sznajder, "The way they move: Tracking multiple targets with similar appearance," in *Proc. IEEE Int. Conf. Comput. Vis.*, Dec. 2013, pp. 2304–2311.
- [15] J. H. Yoon, M.-H. Yang, J. Lim, and K.-J. Yoon, "Bayesian multi-object tracking using motion context from multiple objects," in *Proc. IEEE Winter Conf. Appl. Comput. Vis.*, Jan. 2015, pp. 33–40.
- [16] Y. Rubner, C. Tomasi, and L. J. Guibas, "The earth mover's distance as a metric for image retrieval," *Int. J. Comput. Vis.*, vol. 40, no. 2, pp. 99–121, Nov. 2000.
- [17] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1627–1645, Sep. 2010.
- [18] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. (CVPR)*, vol. 1, Jun. 2005, pp. 886–893 vol. 1.
- [19] X. Wang, T. X. Han, and S. Yan, "An HOG-LBP human detector with partial occlusion handling," in *Proc. IEEE 12th Int. Conf. Comput. Vis.*, Sep. 2009, pp. 32–39.
- [20] P. Dollár, Z. Tu, P. Perona, and S. Belongie, "Integral channel features," in *Proc. Brit. Mach. Vis. Conf.*, 2009.
- [21] S. Zhang, C. Bauckhage, and A. B. Cremers, "Informed Haar-like features improve pedestrian detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 947–954.
- [22] P. Viola, M. J. Jones, and D. Snow, "Detecting pedestrians using patterns of motion and appearance," *Int. J. Comput. Vis.*, vol. 63, no. 2, pp. 153–161, Feb. 2005.
- [23] Z. Cai, M. Saberian, and N. Vasconcelos, "Learning complexity-aware cascades for deep pedestrian detection," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3361–3369.
- [24] Z. Sun, G. Bebis, and R. Miller, "On-road vehicle detection using evolutionary Gabor filter optimization," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 2, pp. 125–137, Jun. 2005.
- [25] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, San Diego, CA, USA, May 2015, pp. 1–14.
- [26] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
- [27] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.
- [28] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.
- [29] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Rich feature hierarchies for accurate object detection and semantic segmentation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 580–587.
- [30] K. He, X. Zhang, S. Ren, and J. Sun, "Spatial pyramid pooling in deep convolutional networks for visual recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 37, no. 9, pp. 1904–1916, Sep. 2014.
- [31] R. Girshick, "Fast R-CNN," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Santiago, CA, USA, Dec. 2015, pp. 1440–1448.
- [32] S. Ren, K. He, R. Girshick, and J. Sun, "Faster R-CNN: Towards real-time object detection with region proposal networks," in *Proc. Adv. Neural Inf. Process. Syst.*, 2015, pp. 91–99.
- [33] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 779–788.
- [34] J. Redmon and A. Farhadi, "YOLO9000: Better, faster, stronger," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 7263–7271.
- [35] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C. Y. Fu, and A. C. Berg, "SSD: Single shot multibox detector," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 21–37.
- [36] L. Huang, Y. Yang, Y. Deng, and Y. Yu, "DenseBox: Unifying landmark localization with end to end object detection," 2015, *arXiv:1509.04874*. [Online]. Available: <http://arxiv.org/abs/1509.04874>
- [37] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua, "Multiple object tracking using K-Shortest paths optimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1806–1819, Sep. 2011.

- [38] T. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE J. Ocean. Eng.*, vol. 8, no. 3, pp. 173–184, Jul. 1983.
- [39] D. Reid, "An algorithm for tracking multiple targets," *IEEE Trans. Autom. Control*, vol. AC-24, no. 6, pp. 843–854, Dec. 1979.
- [40] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2016, pp. 3464–3468.
- [41] F. Yu, W. Li, Q. Li, Y. Liu, X. Shi, and J. Yan, "Poi: Multiple object tracking with high performance detection and appearance feature," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2016, pp. 36–42.
- [42] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," in *Proc. IEEE Int. Conf. Image Process. (ICIP)*, Sep. 2017, pp. 3645–3649.
- [43] Q. Zhao, Z. Yang, and H. Tao, "Differential Earth Mover's distance with its applications to visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 2, pp. 274–287, Feb. 2010.
- [44] V. Karavasili, C. Nikou, and A. Likas, "Visual tracking using the Earth Mover's distance between Gaussian mixtures and Kalman filtering," *Image Vis. Comput.*, vol. 29, no. 5, pp. 295–305, Apr. 2011.
- [45] I. Leichter, "Mean shift trackers with cross-bin metrics," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 695–706, Apr. 2011.
- [46] G. Yao and A. Dani, "Visual tracking using sparse coding and Earth Mover's distance," *Frontiers Robot. AI*, vol. 5, p. 95, Aug. 2018.
- [47] M. Cuturi, "Sinkhorn distances: Lightspeed computation of optimal transport," in *Proc. Adv. Neural Inf. Process. Syst.*, 2013, pp. 2292–2300.
- [48] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, "Network flows," Massachusetts Inst. Technol., Cambridge, MA, USA, Sloan White Paper 2059-88, 1988.
- [49] L. Wen, D. Du, Z. Cai, Z. Lei, M.-C. Chang, H. Qi, J. Lim, M.-H. Yang, and S. Lyu, "UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking," 2015, *arXiv:1511.04136*. [Online]. Available: <http://arxiv.org/abs/1511.04136>
- [50] R. Stiefelhagen, K. Bernardin, R. Bowers, J. Garofolo, D. Mostefa, and P. Soundararajan, "The CLEAR 2006 evaluation," in *Proc. Int. Eval. Workshop Classification Events, Activities Relationships*. Berlin, Germany: Springer, 2006, pp. 1–44.
- [51] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2009, pp. 248–255.
- [52] L. Wang, W. Ouyang, X. Wang, and H. Lu, "Visual tracking with fully convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3119–3127.
- [53] I. Dagan, L. Lee, and F. Pereira, "Similarity-based methods for word sense disambiguation," in *Proc. Assoc. Comput. Linguistics*, 1997, vol. 6493, no. 10, pp. 56–63.
- [54] L. Zhao, Y. Song, C. Zhang, Y. Liu, P. Wang, T. Lin, M. Deng, and H. Li, "TGCN: A temporal graph convolutional network for traffic prediction," *IEEE Trans. Intell. Transp. Syst.*, to be published, doi: 10.1109/TITS.2019.2935152.



YANJIIE ZENG received the B.S. degree in civil engineering from Chongqing Jiaotong University, Chongqing, China, in 2013. He is currently pursuing the Ph.D. degree with the School of Civil and Transportation Engineering, South China University of Technology, Guangzhou, China. His research interests include object detection and visual tracking.



XINSHA FU gained the state council special allowance, in 1993, and was exceptionally promoted as a Professor, in 1998. He currently works on highway planning and design, computer aided engineering and design of highways, transportation infrastructure management systems, intelligent transportation systems, 3S technology, and teaching and research of traffic information. He was awarded 2 second prizes and 7 third prizes of provincial- and ministry-level awards. He has published more than 60 articles and five monographs.



LEI GAO received the master's degree in cartography and geographical information systems from Central South University, Changsha, China, in 2018. He is currently pursuing the Ph.D. degree in transportation engineering with Tongji University. His research interests include traffic data mining, deep learning, and agent-based simulation.



JIawei ZHU received the B.S. degree in geographic information system and the M.S. degree in cartography and geographic information science from Central South University, Changsha, China, in 2013 and 2016, respectively, where she is currently pursuing the Ph.D. degree in surveying and mapping. Her research interests include network representation learning, topological data analysis, and spatio-temporal data mining.



HAIFENG LI (Member, IEEE) received the master's degree in transportation engineering from the South China University of Technology, Guangzhou, China, in 2005, and the Ph.D. degree in photogrammetry and remote sensing from Wuhan University, Wuhan, China, in 2009. He was a Research Associate with the Department of Land Surveying and Geo-Informatics, The Hong Kong Polytechnic University, Hong Kong, in 2011, and a Visiting Scholar with the University of Illinois at Urbana-Champaign, Urbana, IL, USA, from 2013 to 2014. He is currently a Professor with the School of Geosciences and Info-Physics, Central South University, Changsha, China. He has authored more than 30 journal articles. His current research interests include geo/remote sensing big data, machine/deep learning, and artificial/brain-inspired intelligence. He is also a Reviewer for many journals.



YUHENG LI received the M.Sc. degree in high-power radio frequency science and engineering from the University of Strathclyde, Glasgow, U.K., in 2013. He is currently the Deputy Director of the Technology Development Department, China Academy of Electronics and Information Technology. His current research interest includes space electronic information system management.