Joint Headlight Pairing and Vehicle Tracking by weighted Set Packing in Nighttime Traffic Videos

Qi Zou, Haibin Ling, Yu Pang, Yaping Huang, Mei Tian

***Abstract*—We propose a Set Packing (SP) framework for joint headlight pairing and vehicle tracking. Given headlight detections, traditional nighttime vehicle tracking methods usually first pair headlights and then track these pairs. However, the poor photometric condition often introduces tremendous noises in headlight detection and pairing, which leads to unrecoverable errors for vehicle tracking. To overcome the challenge, we propose to jointly model these two tasks in a weighted SP framework. Specifically, a graph is built which takes candidate pair track hypotheses as nodes and encodes in edges both the disjoint constraints for tracking and the no-sharing-headlight constraints for pairing. Solving a weighted SP problem on such a graph produces vehicle trajectories, and facilitates pairing with temporal context and in turn produces high quality vehicle trajectories. The solution, however, raises the issue of unmanageable graph scale since the number of track hypotheses grows exponentially over time. To address this issue, pruning strategies are developed to solve the joint model efficiently. The proposed system is evaluated on two traffic datasets including videos under various challenging conditions. Both quantitative and qualitative results show that our method outperforms other tested methods, both in nighttime vehicle tracking and in multi-target tracking, confirming the benefits of jointly modeling the two tasks.**

***Index Terms*—Vehicle tracking, set packing, joint pairing-tracking model, nighttime traffic surveillance**

I. Introduction

Nighttime vehicle tracking plays an important role in traffic surveillance. It is a core module in many applications such as intelligent headlight control [1], illegal parking detection and traffic flow estimation [2]. It also provides techniques for general traffic behaviour analysis and prediction. In the dark condition, headlights or taillights are almost the only salient features for identifying vehicles. Therefore, nighttime vehicle tracking is often converted into headlight group tracking (or headlight pair tracking in most situation). The special case that targets have identical or very similar appearance requires developing methods relying less on appearance. This Pairing, together with pair tracking, becomes one of the main problems of nighttime vehicle tracking. This also makes nighttime vehicle tracking different from general multi-target tracking (MTT).

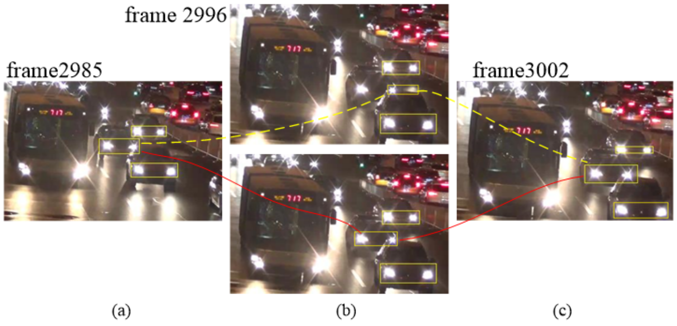


Fig. 1. Benefit of joint headlight pairing and pair tracking. Pairs are in yellow boxes, and trajectories indicated by links. (a)(c)temporal context. (b)pairing ambiguity. Top: traditional methods that perform headlight pairing and pair tracking separately may produce false pairs (dashed yellow links). Bottom: the proposed joint model resolves pairing ambiguities by considering temporal context in trajectories (solid red links).

Tracking multiple vehicles in nighttime traffic is challenging. First, there is a large ambiguity among vehicle headlights since they share similar appearances and shapes. Second, headlight detections are noisy. Reflections cause false positives since they are often as salient as headlights and sometimes move consistently with headlights. Such noises in turn bring troubles to tracking-by-detection methods that rely on accurate detections. Third, in dense or fast traffic scenarios, headlights often spatially interact with each other, causing ambiguities in pairing and data association. Sometimes an illusion of interaction is caused by reflections. The problem is more complicated in the cases of congested traffic and rainy night. Previous works either pair headlights in each frame first and then associate the pairs to form trajectories [1]–[5], or alternatively, track individual headlights first and then discover groups based on tracklet analysis, similar to pedestrian group detection and event detection [6]–[8]. All these methods treat headlight pairing and pair tracking as two separate modules.

In this paper we incorporate headlight pairing and pair tracking in a unified framework. We believe the two tasks are tightly related: vehicle tracking performance is dependent on pairing quality, meanwhile pairing performance can be improved by considering temporal information and constraints in vehicle trajectories. Fig. 1 exemplifies the benefit of joint headlight pairing and pair tracking. Separate pairing and tracking lead to false pairs, as shown in the middle frame at the top row. A joint model can avoid such errors by considering temporal context to resolve pairing ambiguities. In this case, the pairing decision in the bottom row is preferred with the knowledge that the track hypothesis in the bottom row has a higher probability than the track hypothesis in the top row.

The core of our joint pairing and tracking system is the

Q. Zou, Y. Huang and M. Tian is with Beijing Key Lab of Transportation Data Analysis and Mining, Beijing Jiaotong University, Beijing, China, 100044 (e-mail: qzou@bjtu.edu.cn).

H. Ling and Y. Pang is with the Department of Computer and Information Science, Temple University, PA 19121 (e-mail: hbling@temple.edu).

This work is supported by NSFC (61473031, 61472029,61370127),Fundamental Research Funds for the Central Universities (2016JBM016, 2015JBM036) and in part by US NSF Grants CNS-1618398, IIS-1449860 and IIS-1350521.

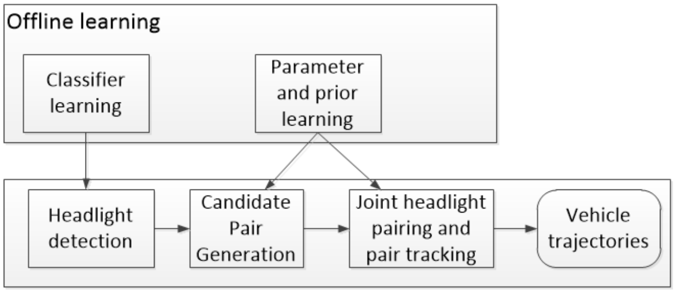

Offline learning
Classifier learning
Parameter and prior learning
Headlight detection
Candidate Pair Generation
Joint headlight pairing and pair tracking
Vehicle trajectories

Fig. 2. Framework of our weighted SP-based nighttime vehicle tracking system.

weighted *Set Packing* (SP) model, which brings two main benefits. First, it simultaneously handles the inter-frame disjoint constraints for data association and the intra-frame no-sharing-headlight constraints for pairing. Second, SP exploits high-order motion information by defining track hypotheses as nodes of the SP model. However, this leads to a problem: hypothesis space will grow exponentially over time. To solve the joint model efficiently, we adopt pruning strategies to keep the number of track hypotheses manageable. The framework of our weighted SP-based nighttime vehicle tracking system is in Fig. 2.

Our contribution can be summarized as: (1) We propose an integrated nighttime vehicle tracking system that jointly optimizes headlight pairing and pair tracking, rather than treats them separately. The standard pipeline which gets robust pairs first and then makes data associations needs more intermediate processes and thus is less efficient. Whats more, errors in any intermediate process will accumulate and have no chance to be recovered. Our joint optimization has less intermediate processes and can get more robust pairs due to the delay decision. (2) We model the joint problem by a weighted SP framework, and adopt effective pruning strategies to control the scalability issue. Besides, perspective projection is considered to get precise geometry features and motion models, and hence to improve the reliability and compactness of candidate pairs and track hypotheses. To evaluate the effectiveness of the proposed system, we test it on three nighttime traffic datasets involving various challenges. Our algorithm demonstrates excellent performance in comparison with state-of-the-art solutions for both nighttime vehicle tracking and general MTT.

In the rest of the paper, after summarizing related works in Sec. II, we first introduce the proposed framework in Sec. III. Then, we describe the joint problem formulation and its solver in Sec. IV. Sec. V describes the implementation and Sec. VI the experiments. Finally, we conclude the paper in Sec. VII.

II. Related Works

A. Nighttime Vehicle Tracking

Previous studies of nighttime vehicle tracking can be divided into two classes according to whether vehicles or headlights are taken as primitives. The former does not require headlight grouping, while the latter does.

Taking vehicles as primitives, [9] directly detects nighttime vehicles based on contrast analysis and tracks vehicles based on nearest neighbor matching. [10] extracts vehicle proposals using an objectness measure and trains a convolutional neural network to recognize vehicle types. In their work, nighttime scenes accommodate the same vehicle detection model as that for daytime conditions. Its robustness to illuminations and noises in nighttime scenes depends on a large-scale training dataset covering different situations.

Taking headlights as primitives, a nighttime traffic surveillance system usually consists of headlight/taillight detection, headlight pairing and vehicle tracking. The main difficulty of headlight detection is to discern headlights from reflections from roads, water, vehicle surfaces and/or lane markings. Methods addressing this issue can be roughly classified into three groups. (1) Rule-based methods: prior knowledge and statistical laws are used on color [3], position, size and shape [2], [11]. (2) Physical-model-based methods: in [12], the Retinex model is used to remove the reflections; in [4], [13] light attenuation law is used to discriminate reflections and headlights. (3) Learning-based methods: decision tree [11], support vector machines (SVM) [14] and AdaBoost [1], [5] are learned from training samples and are powerful in discrimination and generalization.

Usually pairing is based on the symmetry of a pair of headlights [3], [15], for example, the proximity, similarities in areas and shapes [2], [11]. These spatial correlations can match headlight pairs but are sometimes sensitive to noises. To get stable pairs, motion cues are often integrated. [2] first obtains headlight trajectories, and then pairs two trajectories if they move coherently over a period of time. [4] estimates vanishing points and uses bidirectional reasoning to pair effectively.

The most commonly used tracking methods in nighttime vehicle tracking include the Kalman filter [3], [11] and nearest neighbor matching [2], [4], [13]. A tracking-based detection strategy is employed in [3] to improve vehicle detection accuracy. Some recent advances in MTT, which is not specially designed for nighttime vehicle tracking but for general object tracking, is out of the scope of this paper.

The proposed system shares the same headlight detection with [5], but is otherwise completely different. [5] follows a sequential pipeline: "headlight detection–headlight tracking–headlight pairing–pair tracking". It alternatively optimizes context-based headlight tracking and temporal-information-based pairing. Then pair tracking serves as a post-processing of headlight pairing. By contrast, we pair headlights and track the pairs simultaneously by a weighted SP.

B. Set Packing and MTT

Using set packing (SP) in MTT can be traced back to the work by Morefield on using 0-1 programming for data association [16]. The Maximum-Weight Independent Set (MWIS) problem, as a special case of SP, is used for tracking in [17]. The data association is formulated as a union of two-frames MWIS on each independent subgraph and then a linking operation. [18] proposes to associate detections by a relaxed network flow algorithm, which is equivalent to an SP problem. Unlike traditional network flow methods, it can encode motion smoothness on three frames into the cost function.

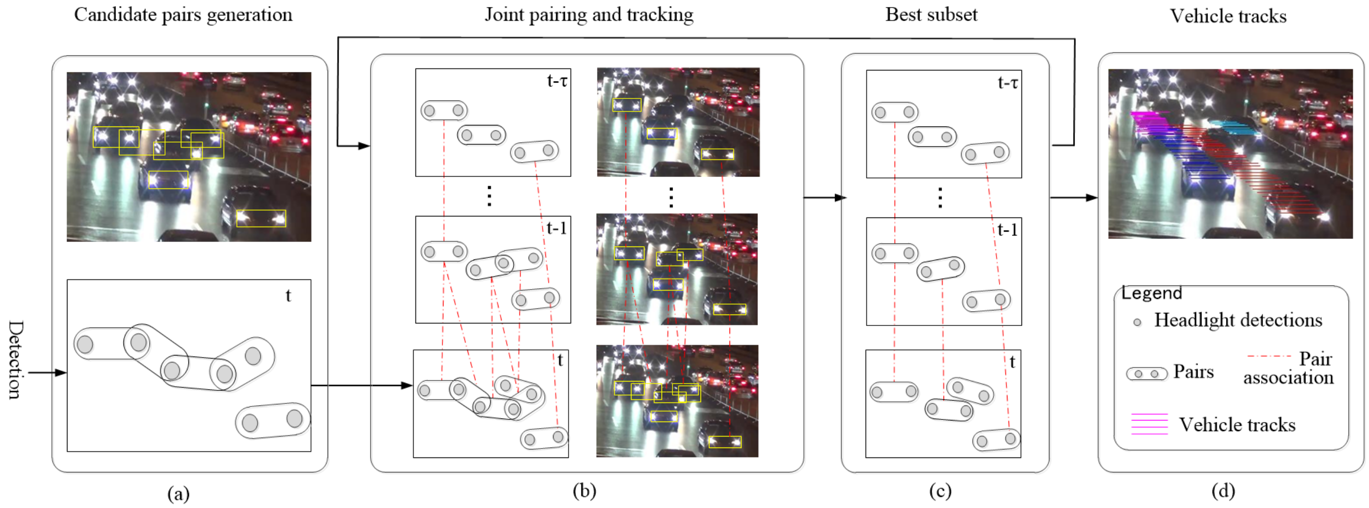


Fig. 3. System core part overview (best viewed in color). (a) Candidate pairs of detected headlights in a new frame t. (b) Joint pairing and tracking. Our online system works in a sequentially forward way: candidate pairs in the frame t update existing pair track hypotheses, seeking for (c) the best subset of non-conflicting tracks up to t. The selected subset is used to prune the track hypotheses and progressively build into (d) final vehicle tracks.

Our work differs from these works in two aspects: firstly, MTT does not consider grouping/pairing, so it alone cannot solve our problem. Secondly, our joint framework uses SP in a different way. Previous works use SP to solve data association over determined detections, but we use it for data association over undetermined pairs where the pairing decision is delayed until ambiguities are resolved by temporal context. Additional constraints on pairing and the dependence between pairing and tracking have to be handled in our joint formulation.

In delaying decision, Multiple Hypothesis Tracking (MHT) [19] can be an alternative to SP, but MHT does not consider pairing. It has to be modified to solve our problem. Recently, [20] presents a MHT method which uses features from deep convolutional neural networks for appearance modeling, and achieves good performance.

Group discovery and group behaviour analysis are of great interest recently as an extending topic of tracking. Our work is different from the method in [6] which aims to discover groups by clustering individual trajectories, and those in [7], [8], [21], which improve individual tracking performance using grouping behaviour as context. Our work focuses on pair tracking, rather than pair discovery and individual tracking.

III. Background and Framework Overview

A. Weighted Set Packing

Since the weighted SP is our basic model, we first provide a formal definition. Suppose we are given a universe set consisting of n elements $U=\{e_i : i=1,\cdots,n\}$, a family of m subsets $S=\{s_i : s_i \subseteq U, i=1,\cdots,m\}$, and each subset s_i has a weight a_i. The weighted SP problem [22] is to find a maximum weighted subfamily from S such that any two subsets in the subfamily are mutually disjoint. Using a binary vector $\Pi=(\pi_i)\in\{0,1\}^m$ to represent a solution, in which $\pi_i=1$ if s_i is in the solution, the weighted SP can be formulated as

$$\tilde{\Pi} = \arg\max_{\Pi} \sum_{i=1}^{m} a_i \pi_i \tag{1}$$

$$\text{s.t.} \quad \begin{cases} \pi_i+\pi_j \leq 1, \forall s_i \cap s_j \neq \emptyset \\ \pi_i \in \{0,1\}, \forall i \in \{1,\cdots,m\} \end{cases} \tag{2}$$

There are two ways to use SP in our task. One is to perform pair selection by taking the headlight set as a universal set and each candidate pair as a subset. The other is to perform data association by modeling an observation sequence from consecutive frames as a subset. In this paper, we use SP in the second way for the ultimate aim is vehicle tracking and the second way is suitable for the joint pairing and tracking framework.

B. Framework Overview

In this paper we study the online nighttime vehicle tracking problem, with focus on robust headlight pairing and pair tracking. Specifically, after a new frame is observed, headlights are first extracted (not the focus in this paper); the detected headlights are then paired and tracked by our system.

The core idea of our framework is to jointly model headlight pairing and pair tracking, so as to make them benefit each other and produce high quality vehicle trajectories. As summarized in Fig. 3, from the detected headlights in a new frame I^t, we first produce a candidate pair set $\mathbb{P}^t$, which is much larger than the set of true headlight pairs. This process will be explained in Sec. IV-A. Let $\mathbf{p}_{k_t}^t$ denote a candidate pair from $\mathbb{P}^t$. Then a candidate pair sequence $\mathbf{p}_{k_{t-\tau}}^{t-\tau}\mathbf{p}_{k_{t-\tau+1}}^{t-\tau+1}\cdots\mathbf{p}_{k_t}^t$ defines a *pair track hypothesis* over $\tau+1$ frames. From a set of such pair track hypotheses, we use a weighted SP to select the best subset, which means the subset of pair track hypotheses that has no conflict (i.e. does not share any observations at any time) and has the highest total reliability. Therefore,

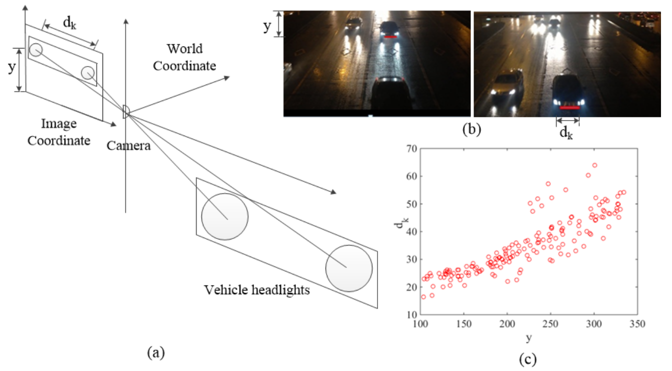


Fig. 4. Illustration of the perspective projection. (a) The geometric model of (b) average vehicle width (approximated by d_k) in the image coordinate system is proportional to the vertical image coordinate y. (c) Statistical data of d_k vs. y from real videos.

the solution of the weighted SP determines true pairs from candidate pairs and simultaneously produces pair tracks.

Specifically, our online system works in a sequentially forward way: after reading in a new frame, pair track hypotheses are updated, from which the best subset is selected to prolong current trajectories and prune current track hypotheses. Then the system proceeds to the next frame.

IV. Problem Formulation

In this section, we first introduce candidate pair generation and then give the joint pairing and tracking model. The associated joint problem solver is derived in the last subsection.

A. Candidate Pair Generation

We use the AdaBoost+Haar detector [23] for headlight extraction. The training set contains patches extracted from early fractions of all sequences. Positive samples are headlights and negative ones are reflections on vehicle bodies or road (water) surfaces, reflective lane markings and traffic signs. The AdaBoost+Haar detector is chosen due to its excellent balance between accuracy and efficiency. Note that though the detector removes most reflections, some false or missing detections survive or escape inevitably. Since most headlights around the margin of traffic scenes are of little use in real applications, we follow a popular strategy in the traffic surveillance to define a region of interest (ROI) for each traffic scene. Such ROI excludes the margin part of traffic scenes and covers only the lanes where vehicles are coming towards the camera.

For a frame at time t, denoted as I^t, let $\mathbb{H}^t = \{\mathbf{h}_i^t = (x_i^t, y_i^t, a_i^t, e_i^t) : i = 1, \cdots, N_h^t\}$ be the detected headlights (may contain false positives), where (x_i^t, y_i^t) is the position, a_i^t the area, e_i^t the aspect ratio, and N_h^t the number of detected headlights.

We apply a perspective-projection-aware proximity criterion to get candidate pairs. This makes the set of candidate pairs compact, discarding unreliable candidates. Let the candidate pair set at time t be $\mathbb{P}^t = \{\mathbf{p}_k^t = \big(\mathbf{p}_k^t(1), \mathbf{p}_k^t(2)\big) : k = 1, \cdots, N_p^t,\ 1 \le \mathbf{p}_k^t(1) < \mathbf{p}_k^t(2) \le N_h^t\}$, where $\mathbf{p}_k^t(1)$ and $\mathbf{p}_k^t(2)$ denote the two headlights in the pair $\mathbf{p}_k^t$ such that

$$(d_k - \beta y_{\mathbf{p}_k^t(1)} - \mu_0)^2/\sigma_0^2 \quad \le \quad \rho_1 \qquad (3)$$

$$|y_{\mathbf{p}_k^t(1)} - y_{\mathbf{p}_k^t(2)}| \quad \le \quad \epsilon \qquad (4)$$

where $d_k = \|(x_{\mathbf{p}_k^t(1)}, y_{\mathbf{p}_k^t(1)}) - (x_{\mathbf{p}_k^t(2)}, y_{\mathbf{p}_k^t(2)})\|_2$ denotes the between-headlight distance. β and μ_0 account for the linear relation between d_k and y due to perspective projection transformation. The difference between d_k and the linear function of y is assumed to be normally distributed with variance σ_0. ρ_1 is the distance threshold. ϵ is assigned to be the headlight height. We assume the view angle of the camera is oriented to the driving lanes. Under such a setting, the two headlights of a vehicle have similar vertical coordinates, with a distance not more than the height of that headlight. The geometric model for perspective projection is illustrated in Fig. 4. The intuition is that the mean vehicle width (approximated by d_k) in the image coordinate system is proportional to its distance to the camera, and also proportional to the vertical image coordinate y. This law is verified statistically by data from real videos, as shown in Fig. 4(c). β, μ_0 and σ_0 are learned from a small training set for each video using model fitting.

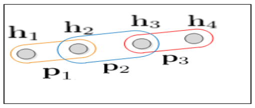


Fig. 5. A toy example of headlight pairs and conflict sets where $\mathbb{H} = \{\mathbf{h}_1, \mathbf{h}_2, \mathbf{h}_3, \mathbf{h}_4\}$, $\mathbb{P} = \{\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3\} = \{(1,2),(2,3),(3,4)\}$ and $\mathbb{C} = \{\mathbb{C}_1, \mathbb{C}_2, \mathbb{C}_3, \mathbb{C}_4\} = \{\{1\},\{1,2\},\{2,3\},\{3\}\}$

Within frame I^t, the pairing problem is to select from $\mathbb{P}^t$ a subset of pairs that maximizes the total pairing affinity, subject to the *no-sharing-member constraint* (i.e., one headlight can be paired at most once). For this purpose, we define the conflict set $\mathbb{C}_i^t$ for each headlight $\mathbf{h}_i^t$ as

$$\mathbb{C}_i^t = \{k : \mathbf{p}_k^t(1) = i \ \text{ or } \ \mathbf{p}_k^t(2) = i,\ k = 1, \cdots, N_p^t\}.$$

Note that if a headlight is claimed by only one candidate pair, it has no conflict, and its conflict set contains only one pair. Such trivial conflict sets can be ignored in our algorithm. For notation conciseness, however, we keep these trivial sets so the total number of conflict sets is N_h^t. An illustrative example is given in Fig. 5.

A traditional solution is to first pair headlights in each frame and then track the pairs. However, pairing in a single frame may not be reliable. Therefore, we propose to leave headlight pairing undetermined until the pairing ambiguities can be resolved in the following joint formulation.

B. Joint Formulation of Pairing and Tracking

The key idea is to make pairing and tracking decisions respecting temporal context. We illustrate the joint formulation with an undirected graph $G = (V, E, A)$ constructed from a simple example of a three-frame problem as in Fig.6(c). Each node in V is a pair track hypothesis $\mathcal{T}_l = \mathbf{p}_{k_{t-\tau}}^{t-\tau}\ \mathbf{p}_{k_{t-\tau+1}}^{t-\tau+1} \cdots \mathbf{p}_{k_t}^t$ and associated with a binary variable $z_l, l \in V$. Each edge (l, i) in E connects two conflicting nodes $\mathcal{T}_l$ and $\mathcal{T}_i$. Conflicts come

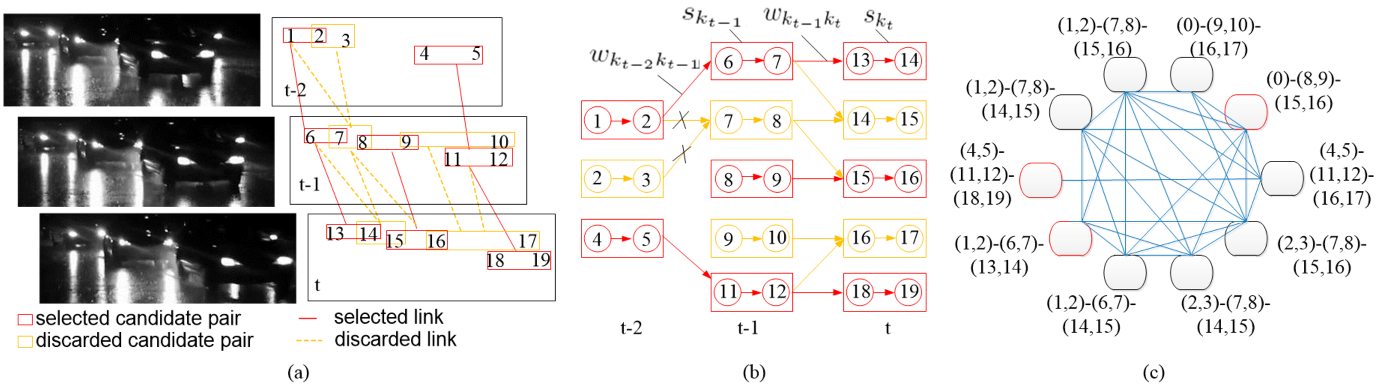


Fig. 6. Illustration of the proposed joint formulation by weighted SP. (a) Pairs and associations in three frames; (b) a simplified flow model for the example of (a). A cross marks an example of track hypothesis pruning. The flows in red denote optimal tracks selected in the final solution. Pruning at $t-2$ cuts flows that sharing any headlight with the optimal tracks at $t-2$; (c) a graph $G=(V,E,A)$ representing the weighted SP model for the example of (a) where a node is a pair track hypothesis and an edge connects two nodes which are conflicting. This figure is best viewed in color.

from two aspects: (1) two tracks share the same pair, or (2) two pairs from two tracks share the same headlight. Each node has a weight $a_l \in A$ describing the pair track affinity. The solution of the joint problem is a subset of V where any two nodes are non-conflicting and the total weight reaches maximum.

$$\arg\max_{\mathbf{z}} \sum_l a_l z_l \tag{5}$$

$$\text{s.t.} \begin{cases} z_l + z_i \le 1, \ \forall (l,i) \in E \\ z_l \in \{0,1\} \end{cases} \tag{6}$$

This classic weighted SP problem can be further formulated as a multidimensional assignment problem. We use an indicator vector $X = (x_{k_{t-\tau:t}})$ [1] to represent a subset of V, where a binary variable $x_{k_{t-\tau:t}}$ indicates whether a pair track hypothesis $\mathbf{p}^{t-\tau}_{k_{t-\tau}} \mathbf{p}^{t-\tau+1}_{k_{t-\tau+1}} \cdots \mathbf{p}^t_{k_t}$ (a node in V) is selected in the subset or not. Note that a track hypothesis may not claim any pair in some frames due to occlusion or missing detection, so we allow dummy pairs $\mathbf{p}^t_0$ as $k_t = 0$. Now the joint pairing and tracking problem can be formulated as

$$\tilde{X} = \arg\max_X \sum_{k_{t-\tau}=0}^{N_p^{t-\tau}} \sum_{k_{t-\tau+1}=0}^{N_p^{t-\tau+1}} \cdots \sum_{k_t=0}^{N_p^t} a_{k_{t-\tau:t}} x_{k_{t-\tau:t}}$$

where

$$a_{k_{t-\tau:t}} = \sum_{i=t-\tau+1}^{t} (w_{k_{i-1}k_i} + \alpha s_{k_i}) \tag{7}$$

such that

$$\begin{cases} x_{k_{t-\tau:t}} \prod_{i=t-\tau}^{t} (\pi^i_{k_i} - 1) = 0 & (8) \\ \sum_{k_{t-\tau}=1}^{N_p^{t-\tau}} \cdots \sum_{k_{i-1}=1}^{N_p^{i-1}} \sum_{k_{i+1}=1}^{N_p^{i+1}} \cdots \sum_{k_t=1}^{N_p^t} x_{k_{t-\tau}\cdots k_i \cdots k_t} \le 1, & \\ \qquad k_i = 1, \cdots, N_p^i; \ i = t-\tau, \cdots, t & (9) \\ \sum_{q \in \mathbb{C}^i_u} \pi^i_q \le 1, \qquad u = 1, \cdots, N_h^i; i = t-\tau, \cdots, t & (10) \end{cases}$$

where $a_{k_{t-\tau:t}}$ is the weight of a track hypothesis, which is a combination of all between-frame association affinities and pair affinities in that track. $w_{k_i k_{i+1}}$ is the association affinity (Eq.16) between pair $\mathbf{p}^i_{k_i}$ and pair $\mathbf{p}^{i+1}_{k_{i+1}}$, and s_{k_i} is the pair affinity (Eq.11) of $\mathbf{p}^i_{k_i}$. $\pi^i_{k_i}$ is a binary variable indicating whether a pair $\mathbf{p}^i_{k_i}$ exists in the final solution ($\pi^i_{k_i} = 1$) or not ($\pi^i_{k_i} = 0$). N^i_h and N^i_p denote the numbers of headlight and candidate pairs at time i respectively.

The first constraint Eq.8 connects headlight pairing and pair tracking together. More specifically, a track hypothesis ($x_{k_{t-\tau:t}} = 1$) implies two things: (1) a candidate pair $\mathbf{p}^{t-\tau}_{k_{t-\tau}}$ from the frame $I^{t-\tau}$, $\mathbf{p}^{t-\tau+1}_{k_{t-\tau+1}}$ from $I^{t-\tau+1}$,$\cdots$, and $\mathbf{p}^t_{k_t}$ from I^t are selected ($\pi^{t-\tau}_{k_{t-\tau}} = 1$, $\pi^{t-\tau+1}_{k_{t-\tau+1}} = 1$, $\ldots$, and $\pi^t_{k_t} = 1$), and (2) these pairs form a track hypothesis. By contrast, $x_{k_{t-\tau:t}} = 0$ indicates either some candidate pairs are not selected or corresponding track hypothesis does not exist. The second constraint Eq.9 is for data association. There is one constraint for each pair in each frame, ensuring one pair is assigned to at most one track. For example, $\big((1,2)-(6,7)-(13,14)\big)$ and $\big((1,2)-(7,8)-(14,15)\big)$ in Fig.6 can not be in the solution simultaneously. The third one Eq.10 is the no-sharing-headlight constraint for pairing. We have one constraint for each conflict set in each frame such as $\{(1,2),(2,3)\}$ in I^{t-2} and $\{(7,8),(8,9)\}$ in I^{t-1} in Fig.6. It enforces that in any frame one headlight can join at most one pair. The last two constraints illustrate the cases when two nodes are connected by an edge.

Note that, the first and third constraints render the problem in Eq.7 no longer a classic linear assignment problem. The data to be associated is now the undetermined pairs which may have conflict. Besides, the pairing and tracking processes interwind with each other.

C. Affinity Definitions

Between-headlight affinity: for a candidate pair $\mathbf{p}^t_{k_t}$, we define the affinity between the two headlights in $\mathbf{p}^t_{k_t}$ as a combination of proximity, area similarity, shape similarity and

[1] For conciseness, we use the notation $k_{t-\tau:t}$ for $k_{t-\tau}k_{t-\tau+1}\cdots k_t$.

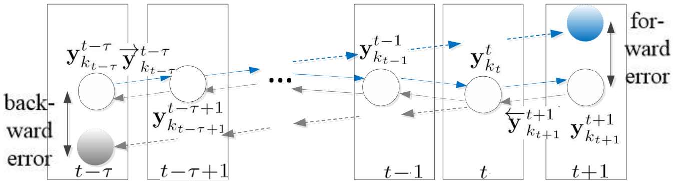

$\mathbf{y}^{t-\tau}_{k_{t-\tau}}$
$\overrightarrow{\mathbf{y}}^{t-\tau}_{k_{t-\tau}}$
$\mathbf{y}^{t-\tau+1}_{k_{t-\tau+1}}$
$\mathbf{y}^{t-1}_{k_{t-1}}$
$\mathbf{y}^{t}_{k_t}$
$\overleftarrow{\mathbf{y}}^{t+1}_{k_{t+1}}$
$\mathbf{y}^{t+1}_{k_{t+1}}$
back-ward error
for-ward error
$t-\tau$
$t-\tau+1$
$t-1$
t
$t+1$

Fig. 7. Illustration of the bidirectional error. Solid arrow lines denote actual associations, and dashed lines denote predicted positions. Forward error measures difference between actual position $y^{t+1}_{k_{t+1}}$ and predicted position based on forward velocity $\overrightarrow{y}^{t-\tau}_{k_{t-\tau}}$; similarly backward error measures difference between $y^{t-\tau}_{k_{t-\tau}}$ and predicted position based on backward velocity $\overleftarrow{y}^{t+1}_{k_{t+1}}$.

motion similarity.

$$s_{k_t} = \begin{cases} \rho_s, & k_t = 0 \\ s^m_{k_t} + \sum_{i=1}^{3} c_i \exp\{-\frac{1}{2\sigma_i^2}(d^i_{k_t} - \mu_i)^2\}, & \text{otherwise} \end{cases} \quad (11)$$

We assign a constant ρ_s for a dummy pair's affinity. When $\mathbf{p}^t_{k_t}$ is not a dummy pair, its affinity is a combination of headlight motion similarity $s^m_{k_t}$, headlight proximity $d^1_{k_t}$, headlight area similarity $d^2_{k_t}$ and shape similarity $d^3_{k_t}$. c_is are the weights; μ_i and σ_i are learned for different traffic scenes. More specifically, let $(u, v) = \big(\mathbf{p}^t_{k_t}(1), \mathbf{p}^t_{k_t}(2)\big)$, these items are defined as:

$$d^1_{k_t} = \|(x_u, y_u) - (x_v, y_v)\|_2 \quad (12)$$

$$d^2_{k_t} = \min(a_u/a_v, a_v/a_u) \quad (13)$$

$$d^3_{k_t} = \min(e_u/e_v, e_v/e_u) \quad (14)$$

Headlight motion similarity is computed as similarity in velocity derived from headlight tracklets. Let $\vec{d_u}$ and $\vec{d_v}$ be velocities of two headlights, then $s^m_{k_t}$ is computed as:

$$s^m_{k_t} = \gamma \frac{2\|\vec{d_u}\| \; \|\vec{d_v}\|}{\|\vec{d_u}\|^2 + \|\vec{d_v}\|^2} + (1-\gamma)\frac{\vec{d_u}\vec{d_v}}{\|\vec{d_u}\| \; \|\vec{d_v}\|} \quad (15)$$

where γ is the weight to balance between velocity magnitude and orientation.

Pair association affinity: $w_{k_t k_{t+1}}$ linking pairs $\mathbf{p}^t_{k_t}$ and $\mathbf{p}^{t+1}_{k_{t+1}}$ is measured jointly by the motion similarity ($w^m_{k_t k_{t+1}}$) and the shape similarity ($w^s_{k_t k_{t+1}}$) for non-dummy pairs.

$$w_{k_t k_{t+1}} = \begin{cases} \ln(1-\rho_D), & k_t = 0 \text{ or } k_{t+1} = 0 \\ \lambda w^s_{k_t k_{t+1}} + (1-\lambda) w^m_{k_t k_{t+1}}, & \text{otherwise} \end{cases} \quad (16)$$

where ρ_D is the detection confidence. Following [24], we define the association affinity for a dummy pair as the log likelihood. For non dummy pairs $\mathbf{p}^t_{k_t}$ and $\mathbf{p}^{t+1}_{k_{t+1}}$, $w^s_{k_t k_{t+1}}$ is measured by a combination of pair width similarity and pair height similarity. λ is the weight that is set small for unreliable headlight sizes, typically for images captured under an undesirable viewpoint or bad exposure.

The motion model is another key component. In applications of tracking objects with fixed cameras, constant velocity assumption is the most practical model. Let a pair track hypothesis $\mathcal{T}_l = \mathbf{p}^{t-\tau}_{k_{t-\tau}} \mathbf{p}^{t-\tau+1}_{k_{t-\tau+1}} \cdots \mathbf{p}^t_{k_t}$ has the positions $[\mathbf{y}^{t-\tau}_{k_{t-\tau}}, \mathbf{y}^{t-\tau+1}_{k_{t-\tau+1}}, \cdots, \mathbf{y}^t_{k_t}]$ with $\mathbf{y}^t_{k_t} = (c^x_{k_t}, c^y_{k_t})$ denoting x and y coordinates of a pair's centroid. To reduce noisy interference, we use bidirectional deviation error, as described in Fig. 7, to measure motion affinity between $\mathcal{T}_l$ and $\mathbf{p}^{t+1}_{k_{t+1}}$.

$$b_{k_t k_{t+1}} = \frac{1}{2(\delta-1)} \sum_{\tau=1}^{\delta-1} \|\mathbf{y}^{t-\tau}_{k_{t-\tau}} + \overrightarrow{\mathbf{y}}^{t-\tau}_{k_{t-\tau}}(\tau+1) - \mathbf{y}^{t+1}_{k_{t+1}}\|_2 + \frac{1}{2(\delta-1)} \sum_{\tau=1}^{\delta-1} \|\mathbf{y}^{t+1}_{k_{t+1}} + \overleftarrow{\mathbf{y}}^{t+1}_{k_{t+1}}(\tau+1) - \mathbf{y}^{t-\tau}_{k_{t-\tau}}\|_2 \quad (17)$$

where $\overrightarrow{\mathbf{y}}^{t-\tau}_{k_{t-\tau}}$ denotes the forward velocity of $\mathbf{p}^{t-\tau}_{k_{t-\tau}}$ and $\overleftarrow{\mathbf{y}}^{t+1}_{k_{t+1}}$ denotes the backward velocity of $\mathbf{p}^{t+1}_{k_{t+1}}$. The first part of Eq.17 computes the deviation error of forward prediction and the second part computes the error of backward prediction. The parameter $\delta = \min(4, |\mathcal{T}_l|)$ where $|\mathcal{T}_l|$ is the number of frames in the tracklet $\mathcal{T}_l$. Since strong correlations usually exist mainly between nearby frames, we use at most four recent frames for position prediction. We only calculate the prediction error for track hypothesis of at least two frames. Based on the bidirectional error $b_{k_t k_{t+1}}$, our motion affinity is computed as

$$w^m_{k_t k_{t+1}} = \exp\Big\{-\frac{1}{2(\sigma^{t+1}_l)^2}(b_{k_t k_{t+1}})^2\Big\} \quad (18)$$

where $(\sigma^{t+1}_l)^2$ is the variance of bidirectional prediction error for $\mathcal{T}_l$ at time $t+1$. Its estimation is similar to the velocity estimation by Kalman filtering.

D. Joint Problem Solver

The joint pairing and tracking problem Eq.7 can be solved by either a greedy randomized adaptive searching algorithm (GRASP) [25] or a relaxed continuous algorithm [17]. A key problem here the salability since the number of nodes in the graph, i.e., the number of track hypotheses, grows exponentially with the length of the track hypothesis. It has to resort to aggressive pruning strategies.

We adopt effective pruning strategies. First, an adaptive gating technique is used when generating track hypotheses. A necessary condition for extending a track hypothesis $\mathcal{T}_i$ with a candidate pair $\mathbf{p}^{t+1}_j$ from a new frame I^{t+1} is

$$(d_{ij} - \eta c^y_i - \mu_4)/\varphi^t_i \le \rho_2 \quad \text{for} \quad d_{ij} = \|\mathbf{y}^t_i - \mathbf{y}^{t+1}_j\|_2 \quad (19)$$

where $\mathbf{y}^t_i$ is the position of $\mathcal{T}_i$ at time t, $\mathbf{y}^{t+1}_j$ is the position of $\mathbf{p}^{t+1}_j$, and d_{ij} measures the distance a vehicle is expected to move between frames I^t and I^{t+1}. η and μ_4 account for the linear relation between d_{ij} and c^y_i (vertical component of $\mathbf{y}^t_i$) due to perspective projection, similar to Eq. 3. This linear relation accounts for the fact that vehicles close to the camera seem to move faster than distant vehicles in the image coordinate system. φ^t_i is the velocity variance of track hypothesis $\mathcal{T}_i$ at time t estimated by Kalman filtering. ρ_2 is the distance threshold, whose sensitivity analysis is given in Fig.11(a). η and μ_4 are learned from a training set using model fitting. All training sets are different from the validation sets.

A gating technique is widely used in the MTT methods and is crucial to balance efficiency and accuracy. Usually the gating criterion is fixed for all objects in all situations. In classical MTT, this criterion is set to not miss candidate tracks but at the price of increasing false positives. Our method uses an adaptive gating technique, which considers velocity changes of vehicles and perspective projection. This enables us to effective reduce missing candidates without significantly increasing the number of hypotheses.

Second, we adopt a standard pruning strategy that is used in multiple hypothesis tracking. The non-conflicting track hypotheses of the highest reliability (HRT) are kept and other hypotheses which share any headlight in the root node of the HRT are pruned. Specifically, when reading in a frame I^t, track hypotheses are maintained from $I^{t-\tau}$ to I^t and each track hypothesis is scored. The best set (having maximal total weight) of non-conflicting tracks can then be found by solving the joint optimization problem in Eq.7. Vehicle trajectories up to frame $I^{t-\tau}$ are determined. Then the track hypotheses which share any headlight with the pairs $\mathbf{p}_i^{t-\tau}$ in the solution are pruned. Afterwards, the algorithm proceeds to the next frame. Take the case in Fig.6 for example ($\tau = 2$), after $\big((1,2)-(6,7)-(13,14)\big)$ is identified as one track in the solution, $\big((1,2)-(7,8)-(14,15)\big)$, $\big((1,2)-(7,8)-(15,16)\big)$, $\big((2,3)-(7,8)-(14,15)\big)$ and $\big((2,3)-(7,8)-(15,16)\big)$ are pruned. This means, we delay pairing and tracking decisions for frame I^t until $I^{t+\tau}$ arrives. After pruning, the number of hypotheses will not grow rapidly when the track hypotheses are extended to new frames.

V. Implementation

A. Track initialization and termination

We use an empirical rule for track initialization. In particular, a new pair track is initialized once a headlight pair is identified as a newly appearing pair. However, it is verified as a vehicle trajectory only if if has been tracked in at least four consecutive frames.

A track can terminate due to occlusion or moving out of the scene. In our system, a pair track hypothesis that cannot be matched to any pair in a new frame is kept using the constant velocity prediction for a short period of time. However, this track hypothesis will be deleted if it keeps unmatched in ten consecutive frames.

B. Handling special vehicles

For vehicles possessing four headlights instead of two, they are handled by a special rule to avoid being identified as two vehicles. We group two pairs into one pair if these two pairs are well aligned and the distance between them is smaller than the length of a vehicle. This also works for grouping a headlight pair with a pair of reflected beams as examples shown in Fig. 8.

Occlusions of one headlight can be handled as following. Single headlights that cannot be paired with any other headlights form a single-headlight set in each frame. Then track hypotheses are generated for single-headlight sets in consecutive frames. These hypotheses together with the track hypotheses formed by headlight pairs are sent to the joint optimization. And the most reliable non-conflicting tracks are selected using SP. If a single headlight has been tracked for at least 20 frames, it is considered to be a vehicle. In this way, single-headlight vehicles such as motorbikes can also be solved.

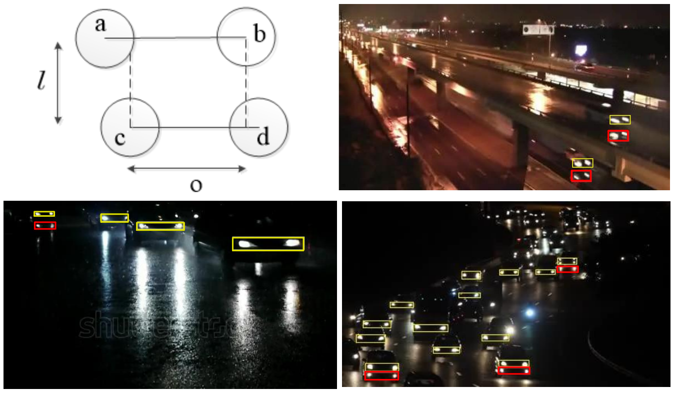


Fig. 8. Examples of handling false positives. (a) Graphic description of the special rule. l is the distance between the headlight pair (a,b) and (c,d), o denotes the overlapping length of (a,b) and (c,d). Two pairs (a,b) and (c,d) are combined into the same group if $\frac{o}{\min(\mathrm{dis}(a,b),\mathrm{dis}(c,d))} \geq 0.8$ and l is less than a vehicle length. (b)(c) A pair of reflections (marked by red boxes) accompanying a pair of headlights (yellow boxes). (d) Four-headlight-vehicles correctly handled by the special rule.

TABLE I
Specifications of the NiTra dataset

Scene type	Seq. name	Quty. of frames	Quty. of vehicles	Density	Resolution	Frame rate(fps)
Urban	s1	1500	42	sparse	640*360	30
	s2	581	31	dense	428*240	15
	s3	268	15	sparse	428*240	15
	s4	451	27	sparse	428*240	15
High-way	s5	1300	94	dense	640*360	30
	s6	350	7	sparse	280*360	25
Rainy-night	s7	198	3	sparse	428*240	30
	s8	895	70	dense	596*336	30
	s9	1500	44	sparse	960*540	28

VI. Experiments

A. Experimental setup

Datasets. The first dataset in our experiment is the the *NiTra Dataset*[2] (short for Nighttime Traffic). It is collected by ourselves with manually labeled groundtruth. It consists of three types of nighttime traffic scenes: Urban, Highway and Rainynight. The Urban subset includes four sequences. They are characterized by vehicles moving in frequently changing velocities, intersecting streets and pairs of moving reflections. The Highway subset includes two sequences, where light-colored vehicles and glares caused by street lamps are the main challenges. The Rainynight subset consists of three sequences. They are characterized by many reflections on the water surface. One of them has a high density and a curvy road, which leads to frequent occlusions. The camera was set up on an elevated platform with a sufficient height to get reliable and clear features of vehicle headlights. And the view angles of the camera were adjusted to be oriented to the lanes. Specifications of the NiTra dataset are listed in Table I.

[2] http://tdam-bjkl.bjtu.edu.cn/qzou/index.html

In addition to the NiTra dataset, we also conduct experiments on the ChangeDetection dataset [26] and the UA-DETRAC dataset [27]. ChangeDetection includes various scenarios for motion detection, among which there is a subset of nighttime traffic videos. UA-DETRAC is a new MOT dataset, among which two sequences satisfy our assumption (dark night, salient headlights, front view) and are used for validation. In experiments, we set $\rho_1 = 4$, $\rho_s = 0.3$, $\rho_D = 0.5$, $c_1 = 0.6$, $c_2 = 0.2$, $c_3 = 0.2$, $\rho_2 = 5$ and $\tau = 4$.

TABLE II
QUANTITATIVE EVALUATION OF VEHICLE TRACKING PERFORMANCE ON THE NITRA DATASET.

Dataset	Method	FPR↓ [a]	MR↓	MOTA↑	MOTP↓
Urban	ours	**3.5%**[b]	10.8%	**85.0%**	**27.0%**
	rul det+SP	5.7%	11.3%	80.3%	27.4%
	con det+SP	20.4%	15.8%	61.6%	30.3%
	rul[c][2]	8.3%	13.5%	75.8%	28.1%
	con[d][9]	26.0%	19.2%	50.3%	31.3%
	MWIS[e][5]	5.0%	**9.5%**	82.5%	29.2%
	MHT [19]	7.6%	16.4%	73.6%	**27.0%**
High-way	ours	**1.7%**	7.2%	**90.1%**	18.1%
	rul det+SP	6.2%	8.3%	83.5%	17.3%
	con det+SP	15.0%	11.7%	69.2%	22.3%
	rul [2]	6.8%	9.7%	81.9%	**17.2%**
	con [9]	20.2%	14.6%	60.7%	25.7%
	MWIS [5]	6.5%	5.2%	87.6%	19.6%
	MHT [19]	8.0%	**4.2%**	85.2%	20.5%
Rainy-night	ours	**10.2%**	**9.5%**	**78.6%**	19.6%
	rul det+SP	12.9%	11.3%	72.7%	**19.6%**
	con det+SP	24.5%	18.4%	50.2%	27.4%
	rul [2]	14.4%	11.6%	70.8%	**19.2%**
	con [9]	27.3%	19.1%	44.0%	30.4%
	MWIS [5]	12.0%	11.2%	75.1%	21.5%
	MHT [19]	11.1%	12.3%	71.3%	21.0%

[a] down arrow means the smaller the better
[b] bold text means the best in a column
[c] the rule based method
[d] the contrast based method
[e] the MWIS based method

Evaluation Metrics. In order to evaluate tracking performance of the joint model and investigate the impact of pairing on individual tracking, four CLEAR metrics [28] are used in our study: *multiple object tracking accuracy* (MOTA), *multiple object tracking precision* (MOTP), *miss rate* (MR) and *false positive rate* (FPR). They are standard metrics in evaluating MTT systems. Specifically, FPR is the ratio of false-positives over the number of groundtruth. MR is the ratio of misses over the number of groundtruth. The MOTA accounts for all errors (false positives, misses and mismatches) made by the tracker over all frames. Higher MOTA means better tracking performance. The MOTP is the tracking precision defined as either average overlapping ratio or average distance between all matched result-groundtruth. It shows the ability of a tracker to estimate precise object positions. For complex multi-target scenarios, the MOTA is shown to be more interesting for overall performance evaluation [28].

Two types of comparisons. We conduct two types of comparisons separately for the state-of-the-arts in nighttime vehicle tracking and for general MTT methods. In particular, the first one evaluates our joint model in comparison with recently proposed nighttime vehicle tracking systems. The systems either follow the standard pipeline as "headlight tracking–per frame headlight pairing–pair tracking", or use ordinary multi-target trackers to detect vehicles as primitives. The second type of comparison is to investigate the impact of pairing on individual tracking, by comparing a variant version of our method with general MTT algorithms. Given headlight detections, an MTT algorithm outputs headlight trajectories while the proposed method outputs pair trajectories. So it is unfair to compare them directly. To deal with this issue, we compare a variant version of our method (separating headlight trajectories from the paired ones) with other MTT methods.

B. Comparison with nighttime vehicle tracking systems

Some examples of vehicle tracking results of the proposed system are shown in Fig. 9. Under various challenging conditions, such as the dense traffic, strong reflections and occlusions, the proposed system tracks most vehicles reliably.

The proposed method is compared with six methods: a contrast-based method [9], a rule-based method [2], our SP-based tracker with the headlight detector used in [2], our SP-based tracker with the headlight detector used in [9], an MWIS-based method [5] and MHT [19]. All methods are compared over same sequences. The algorithm in [9] deals with vehicles rather than headlights directly and therefore avoid the grouping procedure. Both [2] and [5] follow the standard pipeline. [2] applies scene-specific rules for headlight verification and pairing, and a nearest-neighbor approach for tracking; [5] uses MWIS for pairing and a context-based Hungarian algorithm for tracking. [19] uses MHT to solve data association. In the comparison, we use a modified version (incorporating pairing and adding no-sharing-headlight constraints) of the classical MHT to solve our problem.

The quantitative evaluation of vehicle tracking performance on NiTra is listed in Table II (performance averaged over all sequences of each type). We also show examples of pairing results of the five methods in Fig. 10. From the results we observe: (1) generic MTT routine which detects vehicles as primitives is not suitable for nighttime vehicle tracking (as in Fig. 10(a)(g)). Vehicles in nighttime videos are not as informative and reliable as in daytime. They cannot provide accurate input to subsequent tracking, leading to the lowest MOTA of of 50.3% and 44%. (2) Performance of the standard pipeline is affected by the performance of each module, since the standard pipeline is a combination of a headlight tracker, a pairing approach and a pair tracker. [2] and [5] can get correct results when each of their modules produces reliable intermediate results. (3) Our system improves 3% in MOTA compared with the MWIS-based method, improves 9% in MOTA compared with the standard pipeline [2], and improves 5%-11% compared with MHT, owing to the joint optimization framework. (4) The tracking performance will degrade if the headlight detector quality descends, which is shown by the

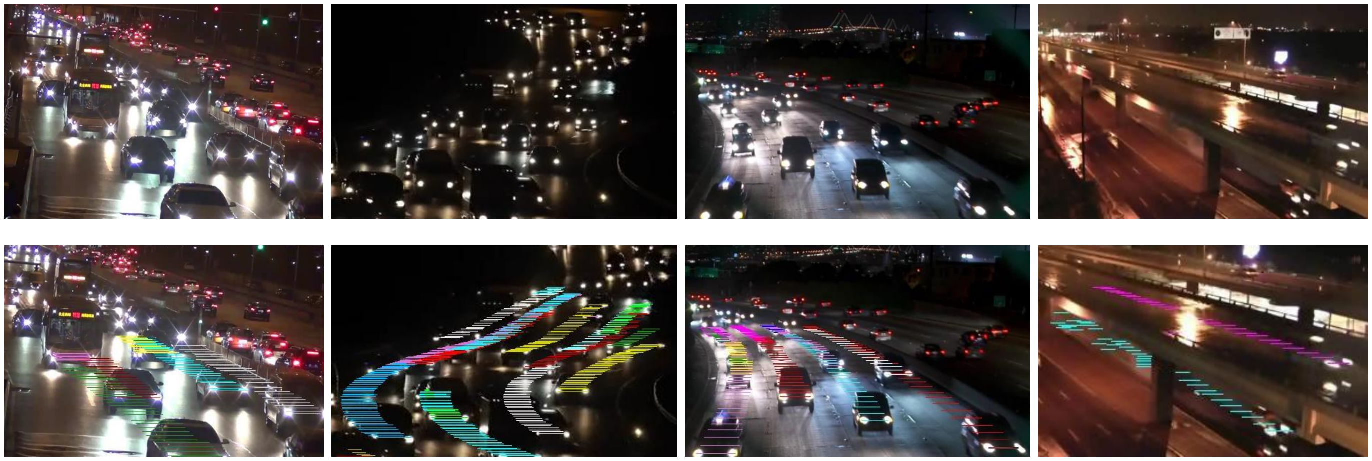


Fig. 9. Examples of vehicle tracking results. Top: input frames. Bottom: output pair trajectories in colored lines (different colors represent different identities).

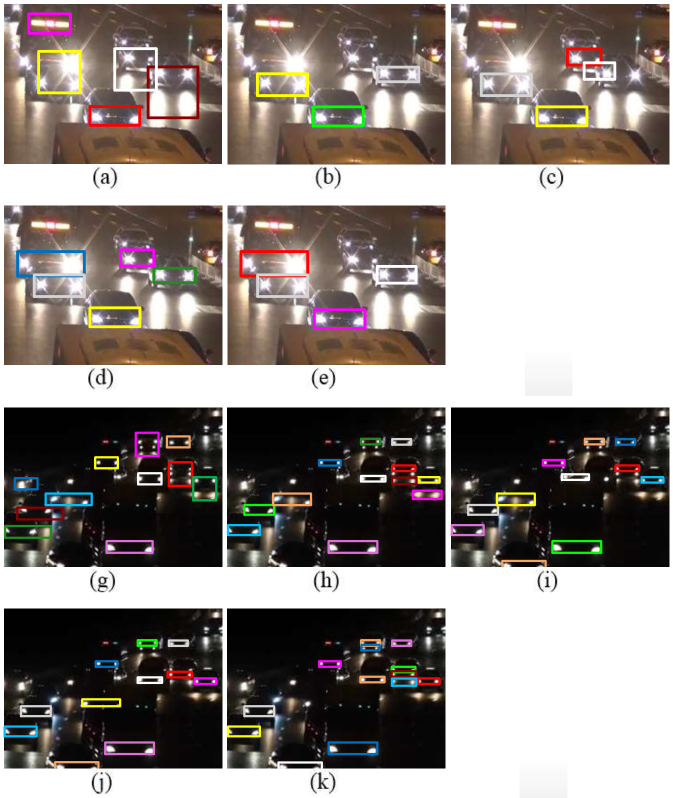


Fig. 10. Two examples of pairing results where different colors represent different identities. (a)(g) contrast based method [9]; (b)(h) rule-based method [2]; (c)(i) MWIS-based method [5]; (d)(j) proposed method; (e)(k) MHT [19].

comparison of ours with ’rul det+SP’ and ’con det+SP’. MOTP is measured in terms of overlapping ratio with the threshold 0.8.

For the ChangeDetection dataset [26], two sequences are used in our evaluation: busyBoulvard and fluidHighway. They present different challenges for nighttime vehicle tracking, characterized by a special view angle of side view and low frame rates (about 3-5 fps) which lead to large displacements between consecutive frames. The ChangeDetection dataset is designed for motion detection, and may not be ideal for tracking. However, public datasets that contain nighttime traffic scenes are rare. Performances on ChangeDetection dataset are summarized in Table III. The result of [2] on this dataset is not as good as that on the NiTra dataset, because its data association is based on overlapping between detections in two frames. However such overlapping in this dataset is rare. Our method performs better despite the challenges. Here MOTP is measured in terms of Euclidean distance. For the UA-DETRAC dataset [27], our tracking performance is 81.0% in MOTA, which improves 2.1% compared with the second best MWIS-based method.

TABLE III
QUANTITATIVE EVALUATION OF VEHICLE TRACKING PERFORMANCE ON CHANGEDETECTION DATASET.

Sequence	Method	FPR	MR	MOTA	MOTP
busy-Boulvard	ours	**12.3%**	**18.1%**	**61.6%**	8.7
	rul [2]	21.9%	29.3%	38.4%	9.2
	con [9]	18.9%	25.5%	46.6%	11.4
	MWIS [5]	16.3%	20.2%	58.5%	**8.2**
	MHT [19]	14.7%	20.0%	59.2%	8.6
fluid-Highway	ours	9.7%	15.9%	**72.6%**	6.5
	rul [2]	10.8%	17.1%	64.9%	6.5
	con [9]	21.4%	**15.3%**	52.5%	8.9
	MWIS [5]	11.3%	16.2%	69.3%	7.0
	MHT [19]	**9.2%**	16.8%	70.5%	**6.4**

We also show the effect of our hypothesis pruning strategy. Without any pruning, the growth in the number of detections will lead to a rapid growth in the number of track hypotheses, as shown in Fig. 11(b). Compared with MHT, which adopts a gating technology and an ”N-scan-back” algorithm [19] to prune hypotheses, our strategy keeps almost half of the hypotheses. MHT has to maintain a considerable amount of hypotheses (with a fixed gating criterion) to achieve competitive performance. Our pruning strategy considers velocity changes and perspective projection to screen hypotheses. This enables us to effectively cut hypotheses while keeping competitive tracking accuracy, as shown in Fig. 11(c). We only compare tracking performance on the sequences where vehicles are

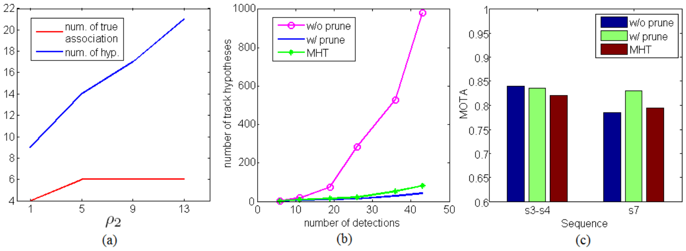


Fig. 11. Comparisons of our methods with and without hypotheses pruning strategies, and with MHT. (a) Sensitivity analysis for the pruning parameter ρ_2. (b) The number of track hypotheses versus the number of detections in three consecutive frames. (c) MOTA for sequence s7 and averaged by s3 and s4, compared among three methods.

TABLE IV
COMPARISON OF MTT PERFORMANCE ON HEADLIGHTS

Dataset	Methods	FPR	MR	MOTA	MOTP
Urban	Tensor [30]	10.5%	**7.2%**	75.3%	28.4%
	SMOT [29]	7.5%	9.1%	80.8%	**26.1%**
	Ours	**5.9%**	8.6%	**83.1%**	35.9%
High-way	Tensor [30]	6.6%	**6.1%**	85.2%	24.3%
	SMOT [29]	6.5%	8.3%	84.7%	**21.7%**
	Ours	**4.1%**	6.8%	**87.6%**	37.5%
Rainy-night	Tensor [30]	15.0%	**10.1%**	69.6%	**25.3%**
	SMOT [29]	**12.3%**	12.0%	71.3%	28.6%
	Ours	12.4%	10.2%	**74.2%**	41.0%

sparse, since without pruning the problem will be untractable in dense traffic scenarios.

Our method is still challenged in cases of long-term occlusions, vehicles with arbitrary number of headlights and crossroads.

C. *Comparison with State-of-the-arts in MTT*

As stated in Sec. I, the proposed method differs from state-of-the-arts in general MTT, which makes the comparative tests hard. Thus a variant version of our method in terms of headlight tracking is reported. Specifically, first vehicles are tracked using the proposed joint method via SP and then single headlight tracks are restored. In our task, headlights have similar appearance, so it is unfair to compare our method with MTT methods relying on discriminative appearance. To this end, we select SMOT [29] for comparison which is specially designed for tracking multiple objects with similar appearance. Besides, tracking is treated as a multidimensional assignment problem in our task. This makes it unfair to compare our method with MTT methods adopting two-dimensional assignment model. So we select a tensor-based method [30], which also uses multidimensional assignment, for comparison.

All the experiments are conducted using the same headlight detections, i.e., detections obtained by our AdaBoost-based detector. For SMOT, the time window parameter is set to 30, 40 and 50 respectively and the best result is selected. For tensor-based method, the 5th-order tensor is used.

The tracking results are shown in Table IV. Fig. 12 gives an example of headlight trajectories produced by three MTT methods. We notice that our method achieves the highest accuracy in terms of FPR and MOTA, while our method may lose in MR and MOTP. It is due to the fact that assuming vehicles to be pairs of headlights may miss short-time single-headlight vehicles. Separating headlight trajectories from the pair trajectories may decrease location precision (MOTP).

TABLE V
COMPARISON OF AVERAGE RUNNING TIME

	SMOT	Tensor	Ours
Dense traffic	4-6fps	1-2fps	1-2fps
Sparse traffic	10-16fps	6-10fps	5-10fps

Running time of the proposed method is related to the number of track hypotheses (n) and the iteration times (k). In our primitive system implemented in Matlab on a standard PC (2.1GHz, 8G memory, no multi-thread utilized), it runs at 1-2 frame-per-second on dense traffic scenes (approximate 30 headlights per frame), and 5-10 fps on sparse scenes (less than 12 headlights per frame). The most time-consuming part is the optimization problem solver whose computational complexity is $O(kn^3)$. In the future, we can optimize the algorithm by GPU parallel programming. Averagely, our algorithm runs similarly as Tensor, but a bit slower than SMOT, as listed in Table V.

VII. CONCLUSION

In this paper we propose a nighttime traffic tracking system that jointly models headlight pairing and pair tracking in a unified weighted SP framework. The system is evaluated carefully on two traffic datasets and its effectiveness is clearly demonstrated in comparison with state-of-the-arts. The promising performance of our system can be mainly attributed to three ingredients: (1) it inhibits pairing ambiguity by exploiting temporal context in the vehicle tracking; (2) it solves the optimization problem efficiently by using pruning strategies; and (3) it learns geometric models to encode rich discriminative information.

Having demonstrated promising results, we will seek improvements along several directions. First, while it works well for tracking and pairing headlights, how to deal with long term occlusion (e.g., one headlight is occluded) is a challenging issue. Second, it currently does not consider vehicle type, while it would be more accurate to build specific models for different types of vehicles (e.g., big trucks). In addition to these improvements, we are also interested in extending the proposed SP model to handle small group discovery and tracking in crowds, which request scaling the group size to more than two, or to a dynamically varying size.

ACKNOWLEDGMENT

The authors would like to thank anonymous reviewers for valuable suggestions and comments. Ling and Pang gratefully acknowledge the support of NVIDIA Corporation for this research.

REFERENCES

[1] J. C. Rubio, J. Serrat, A. M. López, and D. Ponsa, "Multiple-target tracking for intelligent headlight control," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 453–462, 2012.

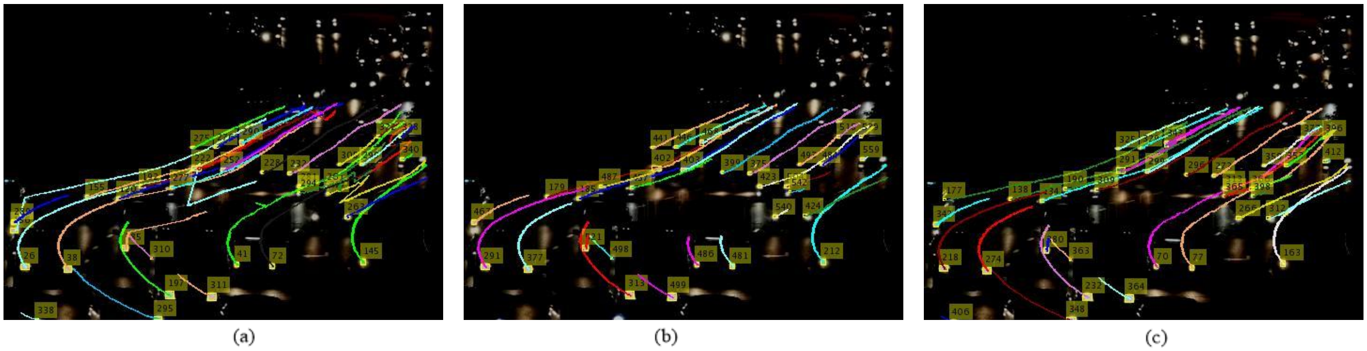


Fig. 12. An example of headlight trajectories in a frame of a rainy night sequence. Results of (a) Tensor; (b) SMOT; (c) Ours.

[2] Y.-L. Chen, B.-F. Wu, H.-Y. Huang, and C.-J. Fan, "A real-time vision system for nighttime vehicle detection and traffic surveillance," *IEEE Trans. Ind. Electron.*, vol. 58, no. 5, pp. 2030–2044, 2011.

[3] R. O´Malley, E. Jones, and M. Glavin, "Rear-lamp vehicle detection and tracking in low-exposure color video for night conditions," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 2, pp. 453–462, 2010.

[4] W. Zhang, Q. M. J. Wu, G. Wang, and X. You, "Tracking and pairing vehicle headlight in night scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 140–153, 2012.

[5] Q. Zou, H. Ling, S. Luo, Y. Huang, and M. Tian, "Robust nighttime vehicle detection by tracking and grouping headlights," *IEEE Trans. Intell. Transp. Syst.*, 2015.

[6] W. Ge, R. T. Collins, and B. Ruback, "Vision-based analysis of small groups in pedestrian crowds," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 5, pp. 1003–1016, 2012.

[7] Z. Qin and C. Shelton, "Improving multi-target tracking via social grouping," *CVPR*, 2012.

[8] S. Pellegrini, A. Ess, and L. V. Gool, "Improving data association by joint modeling of pedestrian trajectories and groupings," *ECCV*, 2010.

[9] K. Huang, L. Wang, T. Tan, and S. Maybank, "A real-time object detecting and tracking system for outdoor night surveillance," *Pattern Recognition*, vol. 41, no. 1, pp. 432–444, 2008.

[10] Y. Yao, B. Tian, and F.-Y. Wang, "Coupled multi-vehicle detection and classification with prior objectness measure," *IEEE Trans. Vehicular Technology*, vol. DOI 10.1109/TVT.2016.2582926, 2016.

[11] K. Robert, "Night-time traffic surveillance a robust framework for multi-vehicle detection, classification and tracking," *Proc. IEEE Conf. AVSS*, 2009.

[12] I. Lee, H. Ko, and D. Han, "Multiple vehicle tracking based on regional estimation in nighttime ccd images," *Proc. IEEE Can. Conf. Acoustics, Speech, and Signal Processing*, pp. 3712–3715, 2002.

[13] C. Tang and A. Hussain, "Robust vehicle surveillance in night traffic videos using an azimuthally blur technique," *IEEE Trans. Vehicular Technology*, vol. 654, no. 10, pp. 4432–4440, 2015.

[14] S. Görmer, D. Müller, M. M. S. Hold, and A. Kummert, "Vehicle recognition and ttc estimation at night based on spotlight pairing," *Proc. IEEE ITSC*, 2009.

[15] J.-M. Guo, C.-H. Hsia, K. Wong, J.-Y. Wu, Y.-T. Wu, and N.-J. Wang, "Nighttime vehicle lamp detection and tracking with adaptive mask training," *IEEE Trans. Vehicular Technology*, vol. 65, no. 6, pp. 4023–4032, 2016.

[16] C. Morefield, "Application of 0-1 integer programming to multitarget tracking problems," *IEEE Transactions on Automatic Control*, vol. 22, no. 3, pp. 302–312, 1977.

[17] W. Brendel, M. Amer, and S. Todorovic, "Multiobject tracking as maximum weight independent set," in *CVPR*, 2011, pp. 1273–1280.

[18] A. A. Butt and R. T. Collins, "Multi-target tracking by lagrangian relaxation to min-cost network flow," in *CVPR*, 2013, pp. 1846–1853.

[19] I. J. Cox and S. L. Hingorani, "An efficient implementation of reids multiple hypothesis tracking algorithm and its evaluation for the purpose of visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 2, pp. 138–150, 1996.

[20] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg, "Multiple hypothesis tracking revisited," *ICCV*, 2015.

[21] X. Chen, Z. Qin, L. An, and B. Bhanu, "An online learned elementary grouping model for multi-target traking," *CVPR*, pp. 4321–4328, 2014.

[22] M. W. Padberg, "On the complexity of set packing polyhedra," *Annals of Discrete Mathematics*, vol. 1, pp. 421–434, 1977.

[23] P. Viola and M. J. Jones, "Robust real-time face detection," *Int. J. Comp. Vis.*, vol. 57, no. 2, pp. 137–154, 2004.

[24] S. Blackman and R. Popoli, *Design and Analysis of Modern Tracking Systems*. Artech House, 1999.

[25] T. A. Feo, M. G. C. Resende, and S. H. Smith, "A greedy randomized adaptive search procedure for maximum independent set," *Operations Research*, vol. 42, pp. 860–878, 1994.

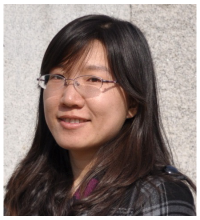
[26] *ChangeDetection*. [Online]. Available: http://www.changedetection.net/

[27] *UA-DETRAC*. [Online]. Available: http://detrac-db.rit.albany.edu/

[28] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The clear mot metrics," *EURASIP J. Image and Video Proc.*, 2008.

[29] C. Dicle, M. Sznaier, and O. Camps, "The way they move: Tracking multiple targets with similar appearance," *ICCV*, 2013.

[30] X. Shi, H. Ling, J. Xing, and W. Hu, "Multi-target tracking by rank-1 tensor approximation," *ICCV*, 2013.



Qi Zou received the Ph.D. degree in computer science from Beijing Jiao Tong University, Beijing, China, in 2006.

In 2014, she was a Visiting Researcher with the Department of Computer and Information Science, Temple university, Philadelphia, USA. She is currently an Associate Professor in School of Computer and Information Technology, Beijing Jiao Tong University. Her research interests include computer vision, pattern recognition and intelligent transportation systems.

Haibin Ling received B.S. and M.S. from Peking University in 1997 and 2000, respectively, and Ph.D. from the University of Maryland in Computer Science in 2006. From 2000 to 2001, he was an assistant researcher at Microsoft Research Asia. From 2006 to 2007, he worked as a postdoctoral scientist at UCLA. He then worked for Siemens Corporate Research as a research scientist. In 2008, he joined Temple University where he is now an Associate Professor.

Ling's research interests include computer vision, augmented reality, human computer interaction, and medical image analysis. Dr. Ling received the Best Student Paper Award of ACM UIST in 2003 and the NSF CAREER Award in 2014. He has served as Area Chairs of CVPR 2014 and CVPR 2016, and currently serves on the editorial board of IEEE Trans. on Pattern Analysis and Machine Intelligence and the Pattern Recognition journal.

Yu Pang received B.S. from Peking University. He is currently working toward the Ph.D. degree advised with Professor Ling at Temple University. His research interests include intelligent transportation systems, image processing, and pattern recognition. He has published papers in ICCV.

Yaping Huang obtained her Ph.D. degree in signal processing from Beijing Jiao Tong University, Beijing, China, in 2004.

She is currently a Professor and Doctoral Supervisor of the School of Computer and Information Technology, Beijing Jiao Tong University. Her research interests include computer vision, pattern recognition and machine learning. She has published over 20 papers in peer-reviewed journals and conferences, including the Information Sciences, the NeuroComputing, the Neural Processing Letters and The Visual Computer.

Mei Tian received the Ph.D. degree in computer science from Beijing Jiao Tong University, Beijing, China, in 2007. She is currently an Assistant Professor in School of Computer and Information Technology, Beijing Jiao Tong University. Her research interests include computer vision, pattern recognition and neural computation.