# Robust Nighttime Vehicle Detection by Tracking and Grouping Headlights

Qi Zou, Haibin Ling, Siwei Luo, Yaping Huang, and Mei Tian

***Abstract*—Nighttime traffic surveillance is difficult due to insufficient and unstable appearance information and strong background interference. We present in this paper a robust nighttime vehicle detection system by detecting, tracking, and grouping headlights. First, we train AdaBoost classifiers for headlights detection to reduce false alarms caused by reflections. Second, to take full advantage of the complementary nature of grouping and tracking, we alternately optimize grouping and tracking. For grouping, motion features produced by tracking are used by headlights pairing. We use a maximal independent set framework for effective pairing, which is more robust than traditional pairing-by-rules methods. For tracking, context information provided by pairing is employed by multiple object tracking. The experiments on challenging datasets and quantitative evaluation show promising performance of our method.**

***Index Terms*—Vehicle detection, vehicle headlight pairing, multiple object tracking, intelligent transportation system.**

## I. INTRODUCTION

TRAFFIC surveillance at night is important for road safety. Accident statistics show that 48 percent of fatalities occur during night [1]. Computer vision-based intelligent traffic surveillance provides techniques for vehicle detection and tracking, which can be used in high-level tasks such as traffic flow analysis, vehicle behavior understanding, and long term motion prediction. All these tasks are important for enhancing driving safety and reducing the chance of accidents. It is also challenging since appearance information is insufficient and unstable for vision-based monitoring systems. Besides, environmental lights and reflections interfere with vehicle detection.

Using appearance information like color, shape or typical vehicle patterns to detect vehicles from different views achieves good performance, but most of these works address the problem during the daytime. At night, above appearance features become invalid, and headlights/taillights are almost the only reliable features. However, street lamps and reflections are often as salient as headlights so as to interfere with the headlights extraction. To solve the problem, different methods have been designed, such as rule-based methods [2], [3], [5]–[8], physical-model-based methods [9], [10] and machine-learning-technique-based methods [4], [11]–[16].

Rule-based methods can produce accurate results if the rules are carefully designed. For example, selecting accurate thresholds for intensity, size, shape, etc., can filter out most low-contrast, large sized and irregular reflections. Such kinds of methods are of high computational efficiency, but are sensitive to specific parameters. Machine-learning-based methods possess good discrimination and better adaptability. However, none of these methods can completely solve the problem. Some works [5], [8], [10], [15], [19] further exploit spatial and temporal coherence to get accurate detection. In terms of spatial and temporal coherence, pairing and tracking are performed.

Pairing helps better discern vehicle headlights from reflections. If a bright blob has a pair in its neighborhood, and both the blobs have similar shape, size, contrast, etc., it is very likely that the two blobs are a pair of headlights from a vehicle. However, pairing also faces difficulties. First, a blob may have more than one similar pair in its neighborhood, especially in dense traffic scenes. Sometimes a blob may have no pair due to occlusion. Second, the similarity relation between two blobs may not be stable along time. Therefore, as suggested in [14], pairing has been used as an optional cue. To better use pairing, an algorithm should reliably deal with matching in dense traffic and selecting stably correlated pairs.

Tracking helps identify vehicle headlights since it provides motion information. *Multiple object tracking* (MOT) in complex traffic scenes is a nontrivial task. It has to solve a multiple dimensional data association problem, whose difficulty is further enlarged by noisy detection. To build a robust tracker, interference from noisy detection and ambiguity in data association must be carefully handled.

In solving the problem of detecting vehicles from headlights candidates, paring and tracking are two complementary subproblems. Persistent co-occurrence of counterparts benefits tracking the right objects and excluding non-headlights. On the other hand, consistent motion correlation helps grouping the headlights to form vehicles.

In this paper, we consider the scenario where a camera is mounted at a fixed position above road, as commonly deployed in traffic monitoring systems. We focus on robust nighttime vehicle tracking, which serves as a basis for high-level traffic surveillance tasks. For example, tracking results from our

Manuscript received September 23, 2014; revised December 8, 2014 and March 9, 2015; accepted April 9, 2015. Date of publication May 21, 2015; date of current version September 25, 2015. This work was supported in part by NSFC under Grants 61473031, 61273364, 61272354, 61300176, and 61105119; by Beijing Higher Education YETP under Grant YETP0546; by Beijing NSF under Grant 4152042; and by Beijing Committee of Science and Technology under Grant Z131110002813118. The work of H. Ling was supported in part by the U.S. NSF under Grant IIS-1218156 and CAREER Award IIS-1350521. The Associate Editor for this paper was H. Huang.

Q. Zou, S. Luo, Y. Huang, and M. Tian are with the Department of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China (e-mail: qzou@bjtu.edu.cn; swluo@bjtu.edu.cn; yphuang@bjtu.edu.cn; mtian@bjtu.edu.cn).

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

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2015.2425229

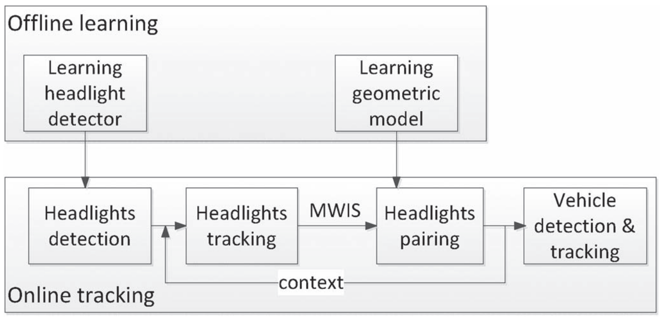



Fig. 1. Framework of vehicle detection at night.

algorithm can be used directly for vehicle counting and velocity estimation, or provide inputs for traffic behavior analysis and prediction.

Addressing the above challenges, we propose a solution that comprises two parts: an offline learning process and an online inference process, as summarized in Fig. 1. During the offline process, a geometrical model is learned and a headlight detector is trained to discriminate reflections from headlights. During the online process, a context-based MOT method is performed following the headlights detection. Then the headlight trackers provide motion information for pairing. Here, a *maximal weighted independent set* (MWIS) method is proposed for effective pairing. Compared with traditional pairing methods that use domain-specific rules and need many thresholds, our solution is more robust. Pairing is based on motion information from tracking and in turn provides spatial context for headlight tracking. After the alternate optimization, vehicles are detected and tracked.

The main contributions of this paper are: first, to take full advantage of the complementarity between pairing and tracking, we solve pairing and tracking in a unified framework by an alternative and iterative optimization. Second, we use context information provided by a paired member to assist headlight tracking. The spatial-temporal contextual information efficiently reduces the disturbance from false detections and ambiguity in data association. Third, we use an MWIS method for effective pairing. Advantages of MWIS are: first, it explicitly represents affinity between headlights and conflicts among pairs in a graphic model. Therefore it can be reduced to a classic vertex packing problem to be solved optimally over the entire candidate set. Compared with traditional pairing-by-rules methods, which are reduced to a local selection problem solved on each pair independently, our method is potentially capable of finding better solution. Second, it is easy to integrate temporal coherence into MWIS to select stably correlated pairs.

This paper is organized as follows: in Section II we briefly review related works. In Section III we explain the offline learning process. In Section IV, following headlight detection, a novel MOT model assisted by spatial context is presented. After that, we introduce headlights pairing under the MWIS framework, and subsequent vehicle detection and tracking. Our iterative optimization algorithm is also given in this section. In Section V we show experiment results and draw conclusions in Section VI.

## II. Related Works

A nighttime traffic surveillance system usually consists of headlight detection, headlight clustering and vehicle detection and tracking.

### A. Headlight Detection Methods

The main difficulty of headlight detection is to discern vehicle headlights from environment lamps and reflections. Reflections include reflective lane markings, traffic signs and reflections of headlights on road or water surfaces. To solve the problem different methods have been designed, which can be classified into rule-based, physical-model-based and machine-learning-technique-based methods.

Rule-based methods are the most commonly used ones. Following a threshold based blob extraction, or some variants such as adaptive threshold and multilevel threshold, they detect headlights according to rules which include prior knowledge and statistical laws on contrast, color, position, size and shape. Huang *et al.* [2] introduced a block-based contrast analysis method to detect vehicles in nighttime scenes. Though it can effectively detect moving objects, it has difficulty in filtering out moving and high-contrast reflections. Chern and Hou [3] analyze data in red–green–blue (RGB) color space to detect the redness of rear lights. They adopt a variable threshold respecting the fact that near spots are brighter than the spots faraway. Other color spaces such as Hue, Saturation, and Value (HSV) [5], $L*a*b$ [6] and YCbCr color space [7] are also used to extract headlights. Chen *et al.* [8] used rules on the width-height-ratios and areas to identify blobs as potential vehicle headlights.

Physical-model-based methods use theories of light diffusing to discern headlights from reflections. Lee [9] used the Retinex model to remove the reflections. Zhang *et al.* [10] proposed to use light attenuation law to get reflection intensity map. However, real nighttime scenes are far more complex than those depicted by ideal models. Retinex model is likely to mistakenly remove dazzling headlights. Reflection intensity map is likely to false detect small sized bright beams as headlights.

Having achieved promising performances, machine learning techniques become popular in intelligent transportation. Robert [4] learned eigen-vehicles and used decision trees to classify non-vehicles, cars and heavy vehicles. Other works employ support vector machines (SVM) [11], [12], AdaBoost [13]–[15] and hidden Markov models [16] which are powerful in discrimination and generalization.

In our system we use AdaBoost for initial headlight detection, i.e., deciding whether candidate blobs are headlights or not. Our usage of AdaBoost is different than that in [13], which learns frontal view of vehicles, and that in [15], which incorporates tracking in the learning of AdaBoost.

### B. Headlight Pairing Methods

Pairing, or more generally clustering headlights, is indispensable for producing vehicle candidates when the output of detectors is headlights. Usually pairing is according to symmetry of a pair of headlights [5], proximity in positions, and similarities in areas and shapes [8], [17], [18]. These spatial correlations can judge headlight-pairs in every frame, but sometimes are sensitive to noises. Chen *et al.* [8] first got headlight tracks, and then paired two tracks if they move coherently for a period of time. This motion based pairing technique improves robustness of pairing, dependent on correct object tracking. In their work, an observation is matched to a trajectory if their overlapping area is above a threshold. It is efficient for traffic scenes with sufficiently high frame rate. However, when displacements between two frames are large and overlapping is few, it may fail to associate data. Zhang *et al.* [10] firstly estimated vanishing points and then used bidirectional reasoning to effectively pairing.

Motion information is also used for pairing in our method. Such information reveals temporal coherence and dynamics, which can be used to improve robustness against false detection. Different from existing works, we build a graphical model to depict between-headlight affinity and the conflicts among pair candidates. Our model solves the pairing problem in a global fashion, and is more suitable for dense traffic scene than previous pairing methods that search in local neighborhood.

### C. Vehicle Tracking Methods

The most common tracking methods used in traffic surveillance are Kalman filters and nearest neighbor matching based on locations and appearance features. Tracking is performed at different levels: the vehicle light level [10], [15], the vehicle level [4], [5], [9], [19] or both [8]. [5] and [19] tracked vehicles localized by rear-light pairs and implemented a tracking-based detection strategy to improve vehicle detection accuracy. Reference [15] used belief propagation on a factor graph to perform MOT. It can efficiently recover occlusion, deal with split and merge. Once speed is estimated, static blobs such as street lamps, blinkers and static reflections can be excluded from vehicle headlights. Some dynamic noises that only appear in a limited number of frames can also be excluded [19]. Long-lasting and moving reflections are still difficult to remove when motion information is independently used.

Using context to improve MOT is of interest because of its good adaptability. Yang *et al.* [24] proposed to search for auxiliary objects to assist track targets. The auxiliary object must move coherently with the target. Shi *et al.* [25] proposed the maximum consistency context which can filter out noisy distractions and applied it in wide area traffic scene.

We also use motion context, named *headlight dynamics context*, to guide multiple object association across frames, but our method differs from [25], named *motion coherence context*, mainly in two aspects. First, we introduce context to reduce disturbance from reflections and other false detections, while [25] uses context mainly to reduce disturbance from vehicles moving in opposite directions. Second, [25] selects context providers directly from targets' neighborhood, while our method determines context providers by a pairing algorithm. Selecting context with maximal motion consistence from neighborhood is effective in their application, but not for our case. There may be multiple local maxima in a target's neighborhood due to reflections. At the same time, different targets may select the same object as context, which causes confliction in the pairing process. Therefore, a graphic-model-based pairing algorithm is proposed in our work to produce the right context for headlight tracking.

## III. Offline Process

During the offline learning process, a geometric model is learned and a headlight detector is trained to discriminate reflections from headlights.

### A. Learning Headlight Detector

We use the AdaBoost+Haar object detection framework [20] to build a headlight detector that discriminates headlights from non-headlights. For each normalized training sample, a 180 dimensional feature vector is extracted which includes 5 types of Haar features at 4 scales and 9 positions on a patch. A major challenge in our task is to discriminate headlight from reflections. The appearance of these two cases is different in contrast patterns between center and periphery. Such difference can be naturally captured by the Haar features, which measure contrast along different directions and at multiple scales.

Positive samples are headlights. Negative samples are reflections on vehicle bodies or road (water) surfaces, reflective lane markings and traffic signs. The Haar features are efficiently computed using the integral image technique. We also conduct a testing process (testing set is different from training set) to quantify the performance achieved by the AdaBoost cascade.

### B. Learning Geometric Model

Given a camera mounted above road, images in a video can be approximated by the perspective projection of the 3D world onto a 2D plane. Due to the perspective effect, the distance between a pair of headlights belonging to the same vehicle, named *between-headlight distance* or *bl-distance*, is large when they are close to the camera, and small when far from the camera. The relation of the bl-distance to vertical image coordinates need be learned for different scenes beforehand.

As shown in Fig. 2, assume the world coordinate system $OXYZ$ is camera-centered, with the $X$ axis toward right, the $Y$ axis facing up, and the $Z$ axis in the vehicle's direction of motion (parallel to the road plane). Assume the image coordinate system is centered at the centroid of the image, with the horizontal and vertical axes parallel to axes in the world coordinate system. Two headlights of the same vehicle are in the scene $(X_1, Y_1, Z_1)$ and $(X_2, Y_2, Z_2)$, with their projected image coordinates $(x_1, y_1)$ and $(x_2, y_2)$. According to the perspective projection equation [21]

$$\frac{X_i}{Z_i} = \frac{x_i}{f}, \quad i = 1, 2 \tag{1}$$

$$\frac{Y_i}{Z_i} = \frac{y_i}{f}, \quad i = 1, 2 \tag{2}$$

where $f$ is the camera focal length. On a flat road, the two headlights of a vehicle are at the same height as well as the same vertical distances $H$ from the camera, i.e. $Y_1 \approx Y_2 \doteq H$, so $y_1 \approx y_2 \doteq y$. The actual distance between a pair of headlights

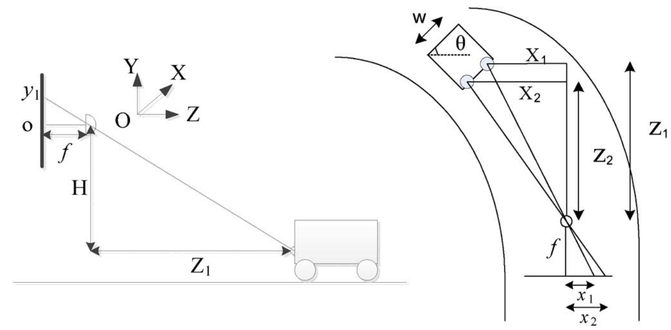



Fig. 2. Geometric model of views from a mounted stationary camera. Left: Coordinate system on a straight road. Right: Top view of a curved road.

in the world coordinates can be denoted as a constant, i.e., $|X_2 - X_1| = W$, so the image distance between a pair of headlights on a straight road is

$$\Delta x = |x_2 - x_1| = y\frac{W}{H}. \tag{3}$$

When the road is curved with an angle $\theta$, as depicted in Fig. 2, the image distance between a pair of headlights

$$\Delta x = |x_2 - x_1| = y\frac{W\cos(\theta)}{H}. \tag{4}$$

If the road is mildly curved, $\cos\theta \approx 1$, formula (4) is the same as formula (3). Therefore, the distance between a pair of headlights is linearly related to their image coordinates y. Considering some variances, we can model the likelihood function as a Gaussian distribution

$$f(\Delta x, y) = c_1 \exp\left\{-\frac{1}{2\sigma_1^2}(\Delta x + ky - \mu_1)^2\right\} \tag{5}$$

where $\Delta x = |x_2 - x_1|$ denotes the image distance between a pair of headlights. $c_1$ is a normalization constant. $k$ represents the linear relation between $\Delta x$ and $y$. $\mu_1$ and $\sigma_1$ are mean and variance of the Gaussian distribution respectively. For each traffic scene, we learn $k$, $\mu_1$, and $\sigma_1$ from a manually labelled training sequence by model fitting.

## IV. Online Process

During the online process, we first detect headlights using the learned headlight detector, and then solve the coupling problem by performing alternately context-based MOT and motion-based pairing. The final output of the process includes vehicle trajectories identified by tracked headlight pairs.

### A. Headlight Detection

Since vehicles in the upper part of a traffic scene are usually far away from the camera and therefore of limited interests, we restrict our task to a region of interest (ROI) in a bottom part of each traffic scene. Specifically, we define an ROI as the near camera portion (i.e., bottom part) of lanes towards the camera. By using such ROIs we not only exclude most faraway vehicles that are not of interest, but also largely reduce the disturbances from street lamps and environments, which often lie in the upper portion of an image. An example of ROI is shown in Fig. 8(a). Automatic ROI detection from a fixed static camera is feasible using offline learning. In this paper, however, we skip this part and focus on robust headlight pairing and tracking. In the rest of the paper, pairing and tracking are performed on the headlights located in such ROIs.

The first step in headlight detection process is to extract bright blobs from ROIs to facilitate subsequent Adaboost classifier-based detector. Blobs with high intensity are extracted by thresholding the luminance and then looking for connected regions. Some postprocessing by morphological operations is performed to remove noises and achieve nontrivial segmentation. The goal of this step is to get all possible headlight candidates and to miss headlights as few as possible. Unfortunately, some false alarms are inevitably involved in.

The second step in headlight detection process is to classify the extracted bright blobs as headlights or not. Here we use a framework similar to but different from Viola and Jones method [20]. Instead of searching in the whole image with sliding windows, we use the bright blobs extracted at the first step as input to the AdaBoost classifier. Since the classifier is applied to vehicle candidates rather than scans the whole image, it is very efficient.

### B. Headlight Tracking by Context-Based Multiple Object Tracking

After detecting headlights in every frame, we track headlights by associating detections in consecutive frames. Some notations are given first. We denote an image sequence as $\{I_t : t = 1, \cdots, N_f\}$, such that $I_t$ is the frame at time $t$. The headlight candidates detected in frame $I_t$ are denoted as $\{l_i^t : i = 1, \cdots, |l^t|\}$ containing $|l^t|$ candidates. For each headlight $l_i^t$ we denote its features as a vector $\{x_i, y_i, a_i, e_i\}$ where $x_i$, $y_i$, $a_i$ and $e_i$ denote the position, area and shape of a headlight respectively. We represent the shape as the ratio of a headlight area to the area of its enclosing bounding boxes. Since most headlights' shapes are convex and nearly circular, their extent ratios $e_i$ are near 1.

Suppose the frame $I_{t-1}$ and $I_t$ are to be associated. To cope with false detections and missing objects, we add dummy elements into the association affinity matrix. For each headlight $l_i^{t-1}$ (or $l_j^t$), we assign a dummy affinity for the case that the headlight has no correspondence in the frame $t$ (or $t-1$). This allows us to handle missing or false detections, as well as a vehicle leaving or entering the scene. These dummy elements make the association matrix square. Without loss of generality, we assume there are $N$ elements in each frame including the dummy ones. The multi-object association can be formulated as to find the assignment $\prod = \{\pi_{ij}\} \in \{0,1\}^{N\times N}$ to maximize certain energy function

$$\tilde{\Pi} = \arg\max_{\Pi} \mathcal{E}(\Pi) = \arg\max_{\Pi} \sum_{i=1}^{N}\sum_{j=1}^{N} s_{ij}\pi_{ij} \tag{6}$$

$$\text{s.t.} \begin{cases} \sum_{i=1}^{N} \pi_{ij} = 1, & j \in \{1, \cdots, N\} \\ \sum_{j=1}^{N} \pi_{ij} = 1, & i \in \{1, \cdots, N\} \\ \pi_{ij} \in \{0,1\}, & i,j \in \{1, \cdots, N\} \end{cases} \tag{7}$$

(7) means cannot assign 2 targets for 1 object or cannot assign 2 objects for 1 target

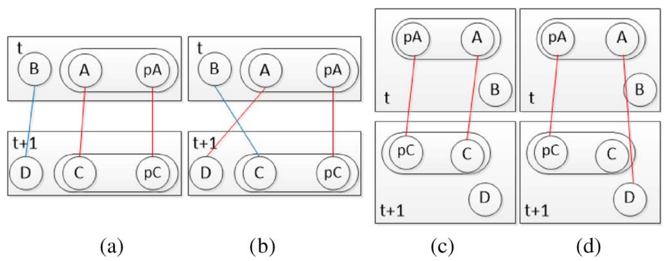



Fig. 3. Two examples of data association between frame $t$ and $t+1$. A headlight and its counterpart are enclosed in an elliptic box. An association between two headlights in frame $t$ and $t+1$ is denoted as a colored line. (a) and (c) show a correct association (A,C) consistent with its context (pA, pC); while (b) and (d) show a wrong association (A,D) inconsistent with its context (pA, pC).

where $\pi_{ij} = 1$ (or 0) indicates an association between $l_i^{t-1}$ and $l_j^t$ exists (or not). $s_{ij}$ is the affinity between $l_i^{t-1}$ and $l_j^t$, which is defined as

$$s_{ij} = \alpha s_{d,ij} + \beta s_{a,ij} + (1-\alpha-\beta) s_{e,ij} \tag{8}$$

where $\alpha$ is the weight of affinity in distance $s_{d,ij}$. $\beta$ and $(1-\alpha-\beta)$ are weights of similarities in area $s_{a,ij}$ and shape $s_{e,ij}$ respectively. $s_{d,ij}$ is defined as

$$s_{d,ij} = c_2 \exp\left\{-\frac{d_{ij}^2}{2\sigma_2^2}\right\}$$
$$d_{ij} = \|\mathcal{P}(x_i, y_i) - (x_j, y_j)\|_2 \tag{9}$$

where $c_2$ is a normalization constant. $d_{ij}$ is the distance between a predicted position $\mathcal{P}(x_i, y_i)$ starting from $(x_i, y_i)$ and an actually observed position $(x_j, y_j)$. The prediction is based on either a constant velocity model or a constant acceleration model, which is estimated from a set of recent consecutive frames. If velocity is unavailable such as in the first two frames or when new objects enter the scene, we predict positions using a velocity prior that is learned from training data. The variance $\sigma_2^2$ is set to $15^2$ that covers the varying range of prediction error and works well for most situations. $s_{a,ij}$ and $s_{e,ij}$ are simply defined as

$$s_{a,ij} = \min\left(\frac{a_i}{a_j}, \frac{a_j}{a_i}\right)$$
$$s_{e,ij} = \min\left(\frac{e_i}{e_j}, \frac{e_j}{e_i}\right). \tag{10}$$

The optimization in (6) can be solved by the Hungarian algorithm [22]. The multi-object association established like this is reliable when traffic flow is low. However, in a congested traffic or high-speed situation, ambiguity in association increases greatly. Such a solution is prone to give false results. Considering a vehicle usually has a pair of headlights moving consistently, we introduce the spatial context provided by the counterpart of a headlight. This idea is illustrated in Fig. 3. A correct association between two consecutive frames [e.g., (A, C) in Fig. 3] should be consistent with its context [i.e., (pA, pC)]. By contrast, an incorrect association [e.g., (A, D)] often violates the consistency with its context. Therefore, spatial context provided by the counterparts can help validate correctness of associations.

To utilize this spatial context, we add a term which describes this context to the energy function (6) without modifying the constraints (7)

$$\max_{\Pi} \mathcal{E}(\Pi) = \max_{\Pi} \sum_{i=1}^{N}\sum_{j=1}^{N} (s_{ij} + r_{ij})\pi_{ij} \tag{11}$$

where $r_{ij}$ describes the spatial relation between the cross-time headlight association $(l_i^t, l_j^{t-1})$ and their association of their partners denoted by $(l_{g(i)}^t, l_{g(j)}^{t-1})$. Specifically, $r_{ij}$ is defined as

$$\begin{aligned} r_{ij} &= r\left(\left(l_i^t, l_j^{t-1}\right), \left(l_{g(i)}^t, l_{g(j)}^{t-1}\right)\right) \\ &= \gamma \mathbf{mag}\left(\left(l_i^t, l_j^{t-1}\right), \left(l_{g(i)}^t, l_{g(j)}^{t-1}\right)\right) \\ &\quad + (1-\gamma)\mathbf{ori}\left(\left(l_i^t, l_j^{t-1}\right), \left(l_{g(i)}^t, l_{g(j)}^{t-1}\right)\right) \end{aligned} \tag{12}$$

where $l_{g(i)}^t$ denotes the partner of $l_i^t$, i.e., $l_i^t$ and $l_{g(i)}^t$ are paired headlights belonging to the same vehicle. How to get paired headlights will be introduced in the next section. $\gamma$ is a weight. **mag** and **ori** are defined as

$$\mathbf{mag}\left((l_i, l_j), (l_p, l_q)\right) = \frac{2 d_{ij} d_{pq}}{d_{ij}^2 + d_{pq}^2} \tag{13}$$

denominator = 0?

$$\mathbf{ori}\left((l_i, l_j), (l_p, l_q)\right) = |\cos(\theta_{ij} - \theta_{pq})|$$
$$\theta_{ij} = \arctan(y_j - y_i / x_j - x_i) \tag{14}$$

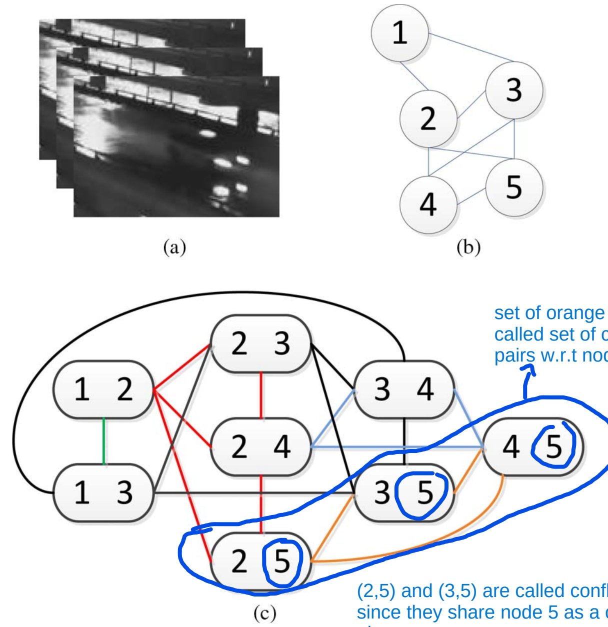
where $d_{ij}$ is the distance between headlight $l_i$ and $l_j$. Formula (12)–(14) capture the fact that if two headlights are from the same vehicle, they will very likely share the same motion pattern, i.e., moving with similar speed and direction. Such a constraint serves as spatial context to reduce association errors. It is robust to small variations and noises. After we get frame-wise data associations along the whole sequence, we can connect them at common vertices and get trajectories.

Fig. 4 shows samples of two-frame association. Here we list the context based association (CA), the association without context (NoCA) and the association by overlapping area based tracking (OAA, the tracking strategy used in [8]). While NoCA is vulnerable to false detection, CA can remove most of the ambiguity caused by false detection. OAA cannot associate data when the overlapping between the same object from two frames is small. In case vehicles move fast, OAA breaks the whole track into different trajectories. In our tracking, a headlight $l_i^{t-1}$ which cannot be matched to any headlight in frame $I_t$ is kept using the constant velocity prediction for a short period of time, but eventually die out if its consecutive invisible times is larger than a threshold. Headlights in $I_t$ which cannot be matched to any current trajectory are used to start new trajectories.

### C. *Pairing by Maximum Weighted Independent Set*

We depict the pairing relations in a frame by an MWIS graphical model. Let $L_t = \{l_i^t : i = 1, \cdots, |l_t|\}$ denote the set

Fig. 4. Between frame association with different methods. For a clear display, images outside the ROIs are cropped. Top: CA. Middle: NoCA. Bottom: OAA. Detected headlights are bounded by yellow boxes. Red (green) lines denote correct (false) links between detections in two consecutive frames. Headlights bounded by green boxes denote the detections that cannot be matched to any detection in previous frame.




set of orange edges are called set of conflicting pairs w.r.t node 5

(2,5) and (3,5) are called conflicting pair since they share node 5 as a common element.

Fig. 5. MWIS graphic model. (a) Part of a frame. (b) A regular graph corresponding to (a) where every node denotes a headlight. An edge between two nodes denotes a potential pairing relation of the two headlights. (c) The MWIS model. A new graph where candidate pairs in the regular graph have become nodes. Colored edges link conflicting pairs, and all edges in the same conflicting set are in the same color. Edges in the MWIS model represent constraints to ensure each headlight is used only once in the final pairing solution.

of all headlights detected in frame $I_t$, and $g_k^t = (l_i^t, l_j^t)$, $l_i^t, l_j^t \in L_t$ be a candidate pair which is a 2-tuple. These candidate pairs are found based on spatial proximity of the observations. The set of all candidate pairs in frame $I_t$ is given as $P_t = \{g_k^t : k = 1, \cdots, |g_t|\}$ where $|g_t|$ is the total number of candidate pairs in that frame.

Based on $P_t$, we generate a graph $G = \{V, E, W\}$. The vertex set $V$ is just the set of all candidate pairs, i.e., $V = P_t$. The edge set $E = \{E_m : m = 1, \cdots, M\}$ consists of $M$ subsets $E_m$ with each subset made up of undirected edges connecting only those conflicting pairs, i.e., the pairs sharing the same headlight. For a headlight $l_m^t$, all the pairs $(l_m^t, *)$ and $(*, l_m^t)$ that include $l_m^t$ are in conflict, because $l_m^t$ can join only one pair. Therefore, they form a conflict subset $E_m$ with respect to $l_m^t$. The weight set $W = \{w_{ij} : (i,j) \in V\}$ is associated with vertices. Each vertex has a positive weight defined by the similarity between a pair of headlights $(l_i^t, l_j^t)$. An example of such a graph model is given in Fig. 5. In the MWIS model, edges in a conflict set, e.g., {(24, 34), (24, 45), (34, 45)} are in the same color.

A subset of $V$ can be represented by an indicator vector $x = (x_{ij}) \in \{0,1\}^{|g_t|}$, where $x_{ij} = 1$ or 0 indicates a vertex exists in the subset or not. Here a vertex corresponds to a pair of headlights. Then the pairing problem can be formulated as finding the following MWIS over graph $G$:

$$\tilde{x} = \arg\max_x \mathcal{E}_G(x) = \arg\max_x \sum_{(i,j)\in V} w_{ij} x_{ij} \tag{15}$$

$$\text{s.t.} \begin{cases} x_{ij} \in \{0,1\}, & \forall (i,j) \in V \\ \sum_{x_{ij}\in E_m} x_{ij} \leq 1, & m = 1, 2, \cdots, M \end{cases} \tag{16}$$

where the first constraint in (16) means it is a binary optimization problem where vertices (candidate pairs) are either selected or not, 1 or 0. The second constraint in (17) ensures that at most one pair is selected among each conflict set, i.e., a headlight can be paired with at most once.

The affinity $w_{ij}$ of a pair of headlights is defined as

$$w_{ij} = f(\Delta x_{ij}, y_{ij}) \left(\lambda s_{a,ij} + (1-\lambda) s_{vel,ij}\right) \tag{17}$$

$$y_{ij} = \frac{(y_i + y_j)}{2} \tag{18}$$

where $\lambda$ is a weight, $f(\Delta x_{ij}, y_{ij})$ and $s_{a,ij}$ are defined in (5) and (10). $s_{vel,ij}$ is the similarity in velocities. $f(\Delta x_{ij}, y_{ij})$ serves as a soft constraint. If the distance between two detected headlights does not follow the pairing distance indicated by the statistical prior, $f(\Delta x_{ij}, y_{ij})$ will be close to 0. Then $w_{ij}$ will be close to 0, which means headlight $l_i$ and $l_j$ are highly unlikely to be paired into the same vehicle. According to (5), when $(\Delta x_{ij} + k y_{ij})$ is $2\sigma_1$ far away from the average, it practically prevents $l_i$ and $l_j$ to form a pair. Only when $(\Delta x_{ij} + k y_{ij})$ follows the statistical prior, we need further compute their area and velocity similarities to decide whether the pair $(l_i, l_j)$ is in the final solution of MWIS. The velocity similarity term $s_{vel,ij}$ is defined as

$$s_{vel,ij} = \frac{1}{4} \sum_{\tau=0}^{3} r\left(\left(l_i^{t-\tau}, l_i^{t-\tau-1}\right), \left(l_j^{t-\tau}, l_j^{t-\tau-1}\right)\right) \tag{19}$$

which measures the motion similarity of $l_i^t$ and $l_j^t$ in four consecutive frames from $I_{t-3}$ to $I_t$. $\{l_i^1, l_i^2, \cdots, l_i^t\}$ denotes the trajectory of a headlight that has been tracked in sequential frames $I_1$ to $I_t$. $r(,)$ has been defined in (12).

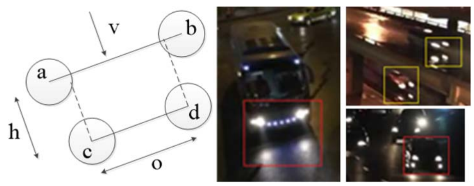



Fig. 6. Graphic description of the condition for postprocessing and 3 real examples. $v$ points to the velocity direction; $h$ is the distance between two headlight pairs (a,b) and (c,d) along the direction $v$; $o$ denotes the length of overlapping between (a,b) and (c,d) along the direction perpendicular to $v$. Two headlight pairs (a,b) and (c,d) are merged into the same group if $o \geq 0.8\min(\mathrm{dis}(a,b), \mathrm{dis}(c,d))$ and $h$ is shorter than a vehicle.

Note that the optimization (15) is different than the optimization (11). In (15) the MWIS framework is a vertex packing problem used to group headlights in a frame to form vehicle candidates, while the optimization (11) is an assignment problem used to associate headlights in consecutive frames to form tracks. We solve the above MWIS problem by the greedy randomized adaptive search procedure (GRASP) [23] with complexity $O(n^3)$.

Since our pairing process is based on the assumption that a pair of headlights form a vehicle, it does not account for vehicles which possess four headlights. In that case, the four headlights may be paired as two vehicles. To solve the problem, we add a postprocessing step that groups 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, as described in Fig. 6. In our implementation, we measure the alignment in two steps: (1) projecting the pairs onto an axis perpendicular to their moving direction, and (2) computing the overlapping ratio of the projected positions. If the ratio is above a threshold, the pairs are considered to be aligned. We approximate the vehicle length by three times the headlight height. A similar postprocessing step also works for grouping a headlight pair with a pair of reflected beams.

The pairing and tracking processes interwind with each other. The pairing algorithm take the tracking results as input, and vice versa. In our algorithm, we iteratively and alternatively optimize them, each time working on one of them and fixing the other, resulting in a constrained quadratic programming in each step. The whole system is summarized in Algorithm 1. If objects have been tracked in less than 4 frames, their velocity similarity (step 12) is computed from the actual frames that have been tracked. If velocity is unavailable when new objects appear, we just use proximity and area similarity to compute the pairing affinity. In our experiments, our algorithm converges to the optimal solution in about 5 iterations.

**Algorithm 1** Headlight tracking and pairing

**Input**: headlight sets $\{l^{t-1}\}$ and $\{l^t\}$ in consecutive frames
**Output**: association matrix $\Pi$ and pairing indicator $x$
1: Compute $\{s_{ij}\}$ according to (8) {Tracking initial}
2: $\mathcal{E}_{\max} = 0, \Pi = \{\pi_{ij}\} = [\,], r_{ij} = 0$
3: $s_{\mathrm{vel},ij} = 0, \mathcal{E}_{G\max} = 0, x = \{x_{ij}\} = [\,]$ {Pairing initial}
4: iter = 1
5: **while** iter $\leq$ MAX$_{\mathrm{it}}$ **do**
6: Update energy function $\mathcal{E}(\Pi)$ as in (11)
7: Solve the assignment problem as in (11) using the Hungarian algorithm, get $\tilde{\Pi}$ and corresponding $\tilde{\mathcal{E}}$
8: **if** $\tilde{\mathcal{E}} > \mathcal{E}_{\max}$ **then**
9: $\mathcal{E}_{\max} = \tilde{\mathcal{E}}$
10: $\Pi = \tilde{\Pi}$
11: **end if** done!
12: Compute $s_{\mathrm{vel},ij}$ as in (19) by the associations $\Pi$
13: Update $w_{ij}$ according to (17)
14: Solve the MWIS problem as in (15) using the GRASP algorithm, get $\tilde{x}$ and corresponding $\tilde{\mathcal{E}}_G$
15: **if** $\tilde{\mathcal{E}}_G > \mathcal{E}_{G\max}$ **then**
16: $\mathcal{E}_{G\max} = \tilde{\mathcal{E}}_G$
17: $x = \tilde{x}$
18: **end if**
19: Compute $r_{ij}$ as in (12) based on $x$
20: iter = iter + 1
21: **end while**

The following parameters are set using empirical knowledge: $\alpha$ and $\beta$ in (8) are set as 0.4 and 0.3 respectively in all sequences. $\gamma$ in (12) is set as 0.5 in sequence 2 and 4, and as 0.7 in the other 3 sequences. $\lambda$ in (18) is set as 0.4 in sequence 1 and 2, and as 0.3 in the other 3 sequences. Roughly speaking, large $\gamma$ is used for vehicles with large velocity change, and small $\lambda$ is used when headlight size is less reliable. This happens to images captured by a camera with undesirable exposure. The parameters related to the geometric model of a specific scene are estimated from statistic data of the training frames, such as $\mu_1, \sigma_1$ in (5) and $\sigma_2$ in (9).

## V. Experiment

This section describes the implementation of the proposed vehicle detection and tracking system. The proposed system was tested on five videos, namely a *congested highway* (seq.1: 1330 frames, $360 \times 640$ pixels per frame), an *urban traffic* scene (seq.2: 200 frames, $360 \times 280$), a *busy street* (seq.3a: 1170 frames, $240 \times 428$, seq.3b: 750 frames, $240 \times 428$, and seq.4: 12375 frames, $360 \times 640$), a *bridge traffic* in a rainy night (seq.5: 300 frames, $240 \times 428$) and another rain night traffic (seq.6: 500 frames, $596 \times 336$). Seq.3a and seq.3b describe the same scene (i.e. the same road and camera configurations) but in different weather (seq.3a without rain and seq.3b in rain). The longest sequence is 8 minutes, covering various traffic conditions and a wide range of velocities.

The proposed method is compared with 3 baselines in headlight detection and 2 baselines in vehicle tracking: a contrast-based [2], a rule-based [8] and a physical-model based method [10]. [2] is designed for night-time visual surveillance. First it searches over a frame with a sliding window to find blocks containing potential objects. The block is analyzed by thresholding the local contrast to decide whether it contains objects. Second, contrast changes in two frames are used to filter the results of the first step and obtain the moving objects. Third, the detected objects are tracked using nearest-neighbor data association strategy. [8] uses rules on the width-height-ratios and areas to identify blobs as potential vehicle headlights.

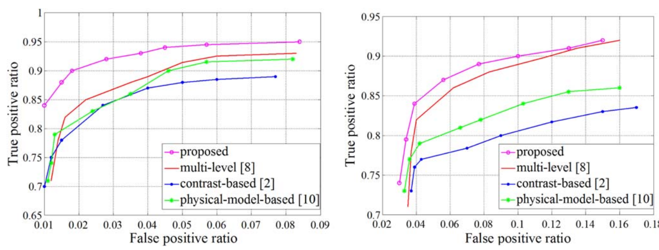



Fig. 7. ROCs of headlight detection on two sequences. (a) ROC of sequence 1; (b) ROC of sequence 5.

Then headlights are tracked to produce headlight trajectories. They pair two headlight trajectories if they keep similar in areas and shapes and move coherently for a period of time. At last, the paired headlights are tracked to verify vehicles. Reference [10] uses light attenuation law to discriminate headlights from reflections. Then they track and pair headlights using the estimated vanishing point and a bidirectional reasoning.

### A. Headlight Detection

Some samples of our headlight detection results are shown in Fig. 8. By applying the Adaboost classifier, our method can detect headlights under different environment illumination conditions. For comparison, we also list headlight detection results of a rule-based method [8] and a physical-model-based method [10].

For the rule-based method, a bright blob $b_i$ is identified as a potential headlight if it satisfies the following two conditions [8]: first, the width-to-height ratio of $b_i$ is within the range of [0.8,1.2], since most headlights have a nearly circular shape; second, the area of $b_i$ is within the range of [minA,maxA] where minA and maxA are estimated from the statistical data for different traffic scenes.

For the physical-model-based method, we compute *reflection intensity maps* (RIM) using the method in [10]. It is according to the light attenuation law. In the headlight regions, RIM takes a low value, whereas in the reflection regions, RIM tends to have a much higher value, as shown in Fig. 8(d).

The comparative results show that the rule-based method cannot remove middle and small regular-shaped reflections. It is suitable for a preliminary classification. The physical-model-based method cannot deal with small and strong reflections, and sometimes mistakes headlights as reflections. This may be attributed to the reason that the physical law of light attenuation depicts the ideal situations. Real nighttime traffic scenes are far more complex than the ideal situations.

The quantitative evaluation of headlight detection is shown in Fig. 7. Two measures are adopted to evaluate the performance: true positive ratio (TP) and the false positive ratio (FP). TP is the ratio of the number of correctly detected headlights to the number of ground truth. FP is the ratio of reflections that are detected as headlights to the number of ground truth. Ground truth for headlight detection is the real headlights labeled in every frame independently. To get TP and FP at different levels for plotting the ROC curves, we tune the threshold which is used to get blobs with high intensity in our method. For [2], we tune its thresholds for local contrast and contrast change. For [10], we tune the threshold for image intensity. And multi-level thresholds are tuned in [8] to segment the foreground.

It shows that the proposed method outperforms the three baselines on all the sequences. [8] performs good due to the multi-level thresholds and carefully designed multiple rules according to specific domain knowledge. The contrast-based method has difficulty in identifying moving reflections with strong contrast, though some falsely detected objects can be removed using feedback from the tracking algorithm. The physical-model-based method tends to false detect small sized bright beams and reflections as headlights. And some of the false detections can also be corrected in the following pairing and tracking.

### B. Headlight Tracking and Pairing

To evaluate the MOT performance on headlights, we compare the tracking with pairing and without pairing on part of the data, as shown in Table I. Tracking without pairing just implements two-frame matching by the Hungarian algorithm. Tracking with pairing also uses the Hungarian algorithm for matching, but its cost function incorporates the context provided by pairing. Ground truth for headlight tracking is headlight trajectories labeled on a sequence.

It can be seen that tracking with pairing improves the tracking accuracy consistently. While the improvement is small for sparse traffic scenes, the improvement is large for dense traffic scenes. This may indicate MOT can benefit from pairing which may serve as a mid-level context.

### C. Vehicle Tracking

For quantitatively evaluating the vehicle tracking performance, we adopt two types of metrics: the Jaccard coefficient $J$ [26] for vehicle detection performance, and CLEAR metrics [27] for vehicle tracking performance. Ground truth for vehicle tracking is the vehicle trajectories (trajectories of headlight pairs) labeled on a sequence. The **Jaccard coefficient** $J$ is defined as

$$J = \frac{\Sigma_{\mathrm{t}} \mathrm{TP}_{\mathrm{t}}}{\Sigma_{\mathrm{t}}(\mathrm{TP}_{\mathrm{t}} + \mathrm{FP}_{\mathrm{t}} + \mathrm{FN}_{\mathrm{t}})} \tag{20}$$

where $\mathrm{TP}_{\mathrm{t}}, \mathrm{FP}_{\mathrm{t}}, \mathrm{FN}_{\mathrm{t}}$ respectively denotes the total number of correctly detected, falsely detected and missed vehicles at frame $t$. The groundtruth data are manually labeled.

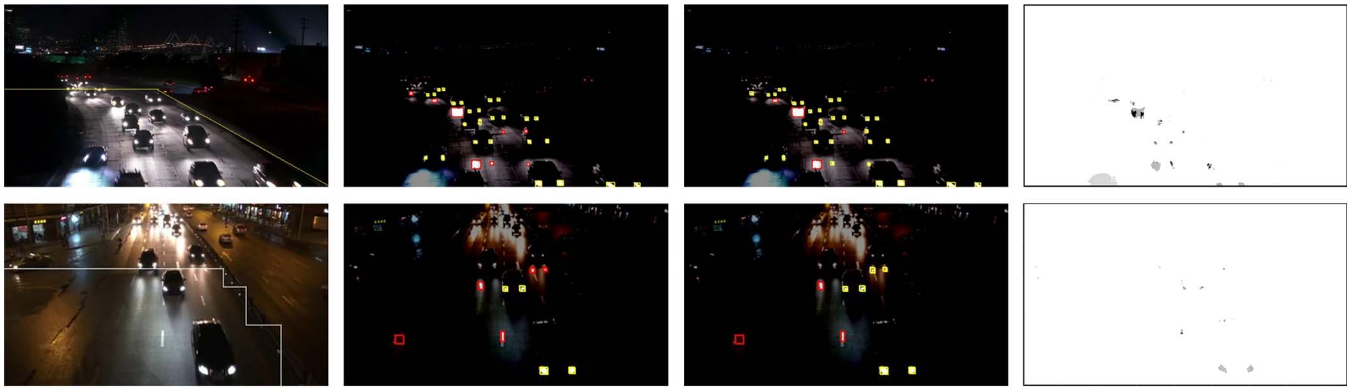
In CLEAR metrics, we use 3 metrics:

**Miss Rate (MR)**: the ratio of misses in the sequence over the number of ground-truth trajectories,

**False Positive Rate (FPR)**: the ratio of false-positive detections that cannot be matched to any ground-truth trajectories over the number of groundtruth,

**Mismatch Rate (MMR)**: the ratio of the number of identity switches over the number of groundtruth.

Fig. 9 shows some representative examples of vehicle detection in a congested highway traffic sequence. Though the traffic flow is high with frequent reflections, beams and dazzling lights, the proposed system can correctly detect and track most



(a) (b) (c) (d)

Fig. 8. Headlight detection results of different methods. (a) Image with ROI. (b) Result of our method. Reflections are bounded in red boxes and headlights are in yellow boxes. (c) Result of rule-based method. Note blobs with areas above a threshold have been excluded in blob extraction for both (b) and (c). (d) RIM of physical-model-based method. Darker (lighter) gray means more (less) likely to be reflections.

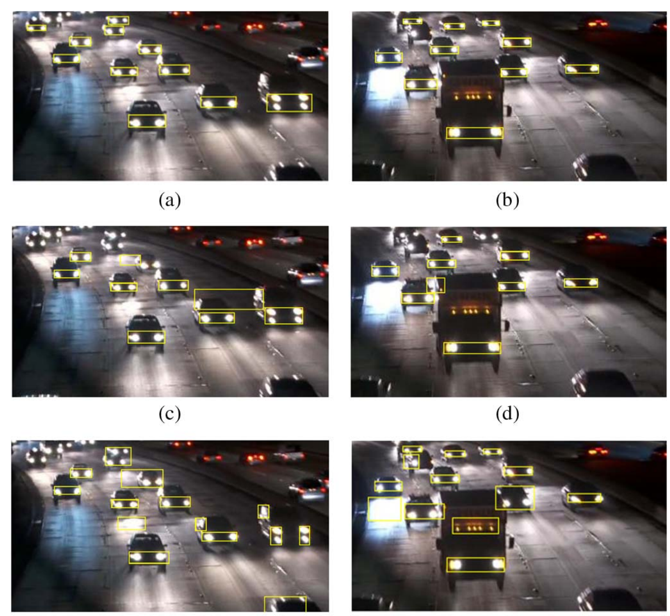
TABLE I
QUANTITATIVE EVALUATION OF HEADLIGHT TRACKING WITH (W) AND WITHOUT (WO) PAIRING

| Data | WO | | | W | | |
|---|---|---|---|---|---|---|
| | J. | MR | FPR | J. | MR | FPR |
| sparse | 90.5% | 3.0% | 7.2% | 94.8% | 2.2% | 3.1% |
| dense | 77.6% | 18.0% | 5.7% | 86.4% | 10.9% | 3.1% |

vehicles. This is owing to three factors: (1) the classifiers learned from training data remove most non-headlights and provide accurate headlights detection; (2) context based tracking produces reliable headlight trajectories, which provide reliable motion information for pairing; and (3) the MWIS based pairing groups headlights in a global fashion and outputs accurate vehicle detection.

The quantitative evaluation is listed in Table II. The MR, FPR and MMR measure the tracker's performance at keeping trajectories of objects. For the contrast-based method, we only measure its MR and FPR, since its tracking error is mainly caused by false detections. Contrast-based method is effective when the local contrast of reflections is low. Its performance will degrade when local contrast of reflections is as high as that of headlights. Such strong reflections are frequent for reflective lane markings and reflections in rainy night. What's more, it has difficulty in accurately detecting vehicles by contrast-based technique alone without any grouping steps. The method in [8] is designed for detecting vehicles and motorbikes, so it allows unpaired single headlights. This makes some long-lasting moving reflections mistakenly treated as single-light motorbikes. Our method assumes paired headlights form a vehicle, so it will remove a single bright blob from vehicle candidates. Some reflections, even though being misclassified as headlight at beginning, will not be tracked as vehicles if they do not have consistent pairs. Our method is less efficient in computation than [8] mainly due to the pairing process (detailed in section D).

Samples of the experiment results in a busy street sequence are shown in Fig. 10. Main difficulty of this sequence includes frequently changing velocities, intersecting streets, and pairs of moving reflections. The method in [8] is designed for vertical streets without considering crossings. The pairing condition and clustering criterion in their method do not work for vehicles which do not run vertically. A failure example of their pairing is shown in Fig. 10(d) (left up car). A failure example of their clustering is given in Fig. 10(c) (left car), where a pair of headlights and a pair of reflections are mistakenly detected as two cars. This is because the headlight pair and the reflection pair cannot be combined for violating the clustering criterion. Our method considers arbitrary directions of velocity, so it can correctly handle such cases.



(a) (b) (c) (d) (e) (f)

Fig. 9. Comparative results on sequence 1 for a congested highway traffic scene. (a), (b) Proposed method. (c), (d) Method of Chen *et al.* (e), (f) Contrast based method.

Samples of the experiment results on two rain sequences are shown in Fig. 11. These sequences are characterized by many glares due to reflections on the water surface and pairs of beams accompanying pairs of headlights. Our method gets

TABLE II
QUANTITATIVE EVALUATION OF THE PROPOSED SYSTEM ON VEHICLE TRACKING

| | Proposed | | | | [8] | | | | Contrast-based | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | Jaccard coef. | MR | FPR | MMR | Jaccard coef. | MR | FPR | MMR | Jaccard coef. | MR | FPR |
| sequence 1 | 91.7% | 6.4% | 1.9% | 0.2% | 82.6% | 11.2% | 7.5% | 0.7% | 41.9% | 24.6% | 33.5% |
| sequence 2 | 95.6% | 2.3% | 2.1% | 0% | 90.9% | 3.3% | 6.4% | 0.2% | 73.9% | 7.5% | 18.6% |
| sequence 3a | 86.5% | 9.6% | 4.5% | 2.1% | 72.6% | 21.1% | 8.7% | 4.0% | 52.3% | 27.1% | 20.6% |
| sequence 3b | 85.8% | 5.9% | 9.7% | 2.8% | 79.3% | 10.0% | 13.5% | 2.7% | 60.5% | 23.2% | 26.9% |
| sequence 4 | 85.4% | 10.1% | 5.0% | 2.5% | 78.0% | 9.2% | 16.4% | 2.2% | 64.7% | 20.3% | 23.2% |
| sequence 5 | 82.7% | 6.9% | 12.6% | 0.5% | 80.8% | 7.1% | 14.9% | 0.5% | 58.4% | 16.3% | 25.3% |
| sequence 6 | 83.4% | 6.0% | 11.9% | 0.4% | 78.5% | 9.3% | 17.6% | 0.9% | 51.4% | 26.9% | 29.2% |

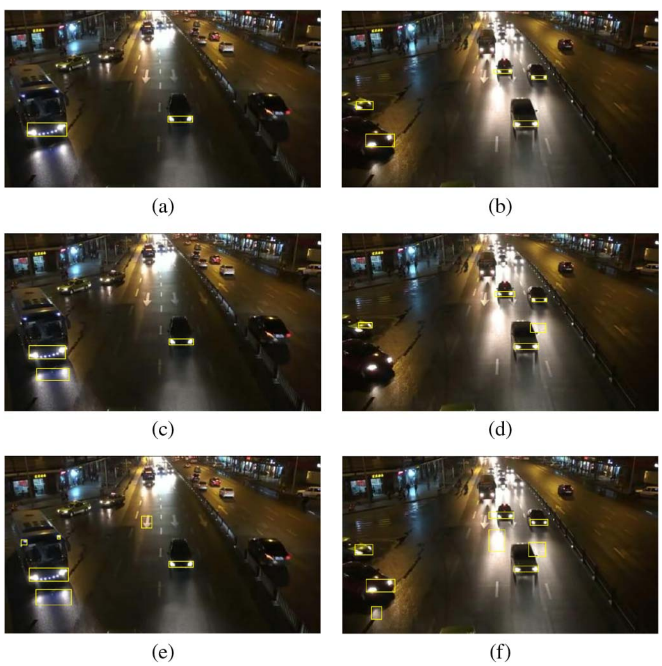


Fig. 10. Comparative results on sequence 3a for a nighttime busy street scene. (a), (b) Proposed method. (c), (d) Method of Chen *et al.* (e), (f) Contrast based method.

ride of most glares by the classification, tracking and pairing process. Some beam pairs which cannot be discriminated from headlight pairs are tackled by our postprocessing step: they are grouped as a single vehicle if they satisfy the grouping conditions.

In the last sequence (bottom row in Fig. 11), latitude of the camera is very low above the ground. This makes the headlight size and between-headlight distance increase dramatically as vehicles approach. Such a viewpoint is less common in traffic surveillance, however it makes sense in multi-target tracking in that it can check tracking performance against large variation in target size and velocity. For the method of Chen *et al.*, it is difficult to estimate lane width. Therefore, some false pairing may happen between nearby headlights. Short-time occlusion in bridge traffic sequence can also be solved by the proposed method. We keep a vehicle's trajectory and predict its positions if it is unassigned with any observation and at the same time it is not at the end of a road. Fig. 12 shows an example.

We also show the tracking and pairing results on consecutive frames in Fig. 13. Headlight trajectories are displayed as colored lines where different colors represent different identities. It is shown that vehicles are stably tracked as headlight pairs. Noisy tracks consisting of false detections exist, however, these tracks are less likely to be paired, or in other words, these tracks are mostly filtered out of the optimal solution by our MWIS-based pairing.

### D. Discussion

The contrast-based method gains low tracking accuracy when applied in nighttime vehicle tracking, but it performs well in tracking other nighttime objects like persons. This may be caused by the particularity of nighttime vehicle tracking. At night, headlights are almost the only salient features, therefore vehicles tracking is reduced to tracking groups of headlights. The contrast-based method detects blocks that contain headlights and tracks these blocks. However, a block may contain a single headlight or two headlights, depending on the size of blocks. Contrast-based methods have difficulty in correctly locating independent headlights. Applying a multi-target tracking algorithm directly on such blocks without segmentation and grouping will lead to unreliable vehicle tracking, just as the contrast-based method does.

Running time of the proposed system is related to the number of headlights $(n)$ and the iteration times $(k)$. Currently our unoptimized system runs at about 1–2 fps (frame-per-second) on dense traffic scenes (approximate 30 headlights per frame), and 5–10 fps on sparse scenes (not more than 12 headlights per frame). The most time-consuming part is the pairing process whose computational complexity is $O(kn^3)$. In future, we can optimize the algorithm by GPU parallel programming.

The proposed method has several limitations left for our future work. 1) The Adaboost based headlight detector needs considerable training samples. In our current version, it uses 2000 positive samples and 6000 negative samples for training. To address this issue, we plan to learn a headlight detector by online collected training samples. 2) Vehicles out of ROIs are difficult to detect and track, although they do not influence driving safety. Distant vehicles out of ROIs can be detected and tracked later when they run into ROIs. 3) Constant velocity based prediction can solve short-period occlusions, but still cannot solve long-term occlusion. 4) Pairing will miss single-light motorcycles or false pair a motorcycle's headlight with other detections. Vehicles with four symmetrical headlights can

(a) (b) (c)

(d) (e) (f)

Fig. 11. Comparative results on sequence 5 for a bridge traffic scene in a rainy night and sequence 6 for a rainy scene. (a), (d) Proposed method. (b), (e) Method of Chen *et al.* (c), (f) Contrast based method.

(a) (b) (c)

Fig. 12. Occlusion solved by the proposed method. (a) Before occlusion. (b), (c) After occlusion.

Fig. 13. Examples of trajectories and pairing results. Top: Source frame. Bottom: Headlight trajectories in colored lines and paired headlights connected by white lines. For a clear display, images in the bottom row are cropped, zoomed, and enhanced.

be correctly detected by a grouping in neighborhood (details in Section IV).

It is also worth noting that, though the proposed method is applied to traffic monitoring with a fixed camera, our key technologies such as context-based tracking, MWIS-based pairing and their collaboration, can be adapted to handle inputs from moving cameras such as vehicle mounted ones. These technologies will help to track vehicles robustly in presence of noisy detection.

## VI. Conclusion

In this paper, we proposed a nighttime traffic tracking system. It first detects headlights through robust learning-based solution, and then tracks vehicles by pairing their headlights. The main contribution of the proposed system lies in its learning-based detection, context-sensitive tracking and graphic model-based headlight pairing. The proposed system is carefully evaluated on several nighttime traffic sequences involving different environmental factors. The advantage of the system is quantitatively and qualitatively demonstrated in these experiments in comparison with state-of-the-arts.

## Acknowledgment

The authors would like to thank Tao Yiqin, Feng Yichao and Ma Li for labeling data, and thank anonymous reviewers for valuable suggestions to improve the paper.

## REFERENCES


[1] "Traffic safety facts, 2012 Motor vehicle crashes: Overview," U.S. National Highway Traffic Safety Administration, Washington, DC, USA, 2013.

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

[3] M.-Y. Chern and P.-C. Hou, "The lane recognition and vehicle detection at night for a camera-assisted car on highway," in *Proc. IEEE Int. Conf. Robot. Autom.*, Sep. 2003, vol. 2, pp. 2110–2115.

[4] K. Robert, "Night-time traffic surveillance a robust framework for multi-vehicle detection, classification and tracking," in *Proc. IEEE Conf. AVSS*, 2009, pp. 1–6.

[5] 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, Jun. 2010.

[6] I. Cabani, G. Toulminet, and A. Bensrhair, "Color-based detection of vehicle lights," in *Proc. Intell. Veh. Symp.*, Jun. 2005, pp. 278–283.

[7] S. Nagumo, H. Hasegawa, and N. Okamoto, "Extraction of forward vehicles by front-mounted camera using brightness information," in *Proc. IEEE Can. Conf. Elect. Comput. Eng.*, May 2003, vol. 2, pp. 1243–1246.

[8] 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, May 2011.

[9] I. Lee, H. Ko, and D. Han, "Multiple vehicle tracking based on regional estimation in nighttime CCD images," in *Proc. IEEE Can. Conf. Acoust., Speech, Signal Process.*, May 2002, vol. 4, pp. 3712–3715.

[10] 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, Mar. 2012.

[11] P. Alcantarilla *et al.*, "Night time vehicle detection for driving assistance lightbeam controller," in *Proc. IEEE Intell. Veh. Symp.*, 2008, pp. 291–296.

[12] S. Görmer, D. Müller, S. Hold, M. Meuter, and A. Kummert, "Vehicle recognition and TTC estimation at night based on spotlight pairing," in *Proc. IEEE ITSC*, 2009, pp. 1–6.

[13] W. Wang, C. Shen, J. Zhang, and S. Paisitkriangkrai, "A two-layer nighttime vehicle detector," in *Proc. Int. Conf. DICTA*, 2009, pp. 619–626.

[14] A. López *et al.*, "Nighttime vehicle detection for intelligent headlight control," in *Proc. 10th Int. Conf. ACIVS*, 2008, pp. 113–124.

[15] 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, Jun. 2012

[16] A. Jazayeri, H. Cai, J. Y. Zheng, and M. Tuceryan, "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.

[17] R. Cucchiara, M. Piccardi, and P. Mello, "Image analysis and rule-based reasoning for a traffic monitoring system," *IEEE Trans. Intell. Transp. Syst.*, vol. 1, no. 2, pp. 119–130, Jun. 2000.

[18] J. Firl, M. Hoerter, M. Lauer, and C. Stiller, "Vehicle detection, classification and position estimation based on monocular video data during night-time," in *Proc. 8th ISAL*, Müchen, Germany, 2009.

[19] B. Tian, Y. Li, B. Li, and D. Wen, "Rear-view vehicle detection and tracking by combining multiple parts for complex urban surveillance," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 2, pp. 592–601, Jun. 2014

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

[21] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry, *An Invitation to 3-D Vision: From Images to Geometric Models*. New York, NY, USA: Springer-Verlag, 2003.

[22] H. W. Kuhn, "The Hungarian method for the assignment problem," *Naval Res. Logistics Quart.*, vol. 2, no. 12, pp. 83–97, 1955.

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

[24] M. Yang, Y. Wu, and G. Hua, "Context-aware visual tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 7, pp. 1195–1209, Jul. 2009.

[25] X. Shi, H. Ling, P. Li, W. Hu, and E. Blasch, "Using maximum consistency context for multiple target association in wide area traffic scenes," in *Proc. IEEE ICASSP*, 2013, pp. 2188–2192.

[26] P. Sneath and R. Sokal, "Numerical taxonomy," in *The Principle and Practice of Numerical Classification*. New York, NY, USA: Freeman, 1973.

[27] K. Bernardin and R. Stiefelhagen, "Evaluating multiple object tracking performance: The CLEAR MOT metrics," *EURASIP J. Image Video Process.*, vol. 2008, pp. 1–10, 2008.


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

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

**Haibin Ling** received the B.S. and M.S. degrees from Peking University, Beijing, China, in 1997 and 2000, respectively, and the Ph.D. degree in computer science from the University of Maryland, College Park, MD, USA, in 2006. From 2000 to 2001, he was an Assistant Researcher at Microsoft Research Asia. From 2006 to 2007, he was a Postdoctoral Scientist at UCLA. He then worked for Siemens Corporate Research as a Research Scientist. In 2008, he joined Temple University, Philadelphia, PA, USA, where he is now an Associate Professor.

His research interests include computer vision, medical image analysis, visual privacy protection, and human–computer interaction. Dr. Ling has served as an Area Chair of CVPR 2014 and as a Guest Co-Editor for Pattern Recognition. He was a recipient of the Best Student Paper Award at ACM UIST in 2003 and the NSF CAREER Award in 2014.

**Siwei Luo** received the Ph.D. degree in computer science from Shinshu University, Nagano, Japan, in 1984. He is currently a Professor and Doctoral Supervisor at the School of Computer and Information Technology, Beijing Jiaotong University, Beijing, China. His research interests include computer vision, pattern recognition, and parallel computing.

**Yaping Huang** received the Ph.D. degree in signal processing from Beijing Jiaotong University, Beijing, China, in 2004.

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

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