Real-Time Multipedestrian Tracking in Traffic Scenes via an RGB-D-Based Layered Graph Model

Shan Gao, Zhenjun Han, Ce Li, Qixiang Ye, *Member, IEEE*, and Jianbin Jiao, *Member, IEEE*

***Abstract*—Multipedestrian tracking in traffic scenes is challenging due to cluttered backgrounds and serious occlusions. In this paper, we propose a layered graph model in image (RGB) and depth (D) domains for real-time robust multipedestrian tracking. The motivation is to investigate high-level constraints in RGB-D data association and to improve the optimization from the trajectory level to the layer level. To construct a layered graph, we define constraints in the depth domain so that pedestrian objects in the image domain are assigned to proper layers. We use pedestrian detection responses in the RGB domain as graph nodes, and we integrate 3-D motion, appearance, and depth features as graph edges. An online updating depth factor is defined to describe the depth relationships among the observations in and out of the layers, and the occlusion issue is processed with an analytical layer-level strategy. With a heuristic label switching algorithm, multiple pedestrian objects are optimally associated and tracked. Experiments and comparison on five public data sets show that our proposed approach significantly reduces pedestrian's ID switch and improves tracking accuracy in the cases of serious occlusions.**

***Index Terms*—Multi-pedestrian tracking, layered graph model, RGB-D data, occlusion.**

I. Introduction

REAL-TIME and accurate pedestrian localization is crucial to Advanced Driving Assistance Systems (ADASs) [1]–[3]. Pedestrian tracking establishes the association of the pedestrians over time, which might be used to obtain pedestrian dynamic information, thus improving the accuracy and efficiency of pedestrian localization.

Pedestrian tracking in suburban districts has made great strides, given scenes where backgrounds are simple and occlusions seldom occur [4]–[6]. In complex scenes such as crowded urban districts, the problem of multi-pedestrian tracking remains far from being solved for frequent occlusions among objects with high dynamic backgrounds. The problem is often aggravated when using a heads-up-view RGB camera without any depth information.

In addition to conventional RGB cameras, depth sensors are increasingly utilized in robotics and ITS. RGB data is obtained with CCD cameras, while D data is typically obtained with stereo cameras [5]–[11] or with a laser range sensor [12]–[19]. In [5], the visual odometry from stereo vision was leveraged to localize a pedestrian's Region-Of-Interests (ROIs), on which a filtering procedure based on empirically defined velocity, size, and color constraints was explored for performing in pedestrian tracking. In [9], scene geometry with stereo depth information was initially built with a calibrated camera on a moving platform, and then a tracking-by-detection framework was utilized to perform pedestrian localization. In [20], a Bayesian fusion system adopted a two-stage strategy, in which, a laser-based detector and feet trajectory tracker were combined in a tracking-by-detection framework. In [21], the laser-based tracker and online trained vision-based classifier were employed when the targets were in close proximity. In the above approaches, RGB and D data are often serially integrated, i.e., D data is leveraged to extract ROIs, reducing false detection and/or tracking and improving tracking efficiency. Without using an optimal data integration strategy, however, the implementation between RGB and D data would not be fully explored.

In this paper, we propose a real-time multi-pedestrian tracking approach by integrating vision and depth (RGB-D) data. Based on RGB-D data, we propose a novel data association model, the layered graph model, and develop an occlusion handling strategy. With optimal graph modeling, we improve the conventional discrete-continuous relation from trajectory-level to layer-level, which enables tracklets (short trajectories) to be accurately associated in a much smaller search space. The layered graph model, occlusion handling strategy and well-designed low level tracking features jointly yield a robust multi-pedestrian tracking system in complex traffic scenes.

The proposed approach is specified for the ADAS in traffic scenes, which are quite different from those specified for surveillance scenarios. In traffic scenes, the continuity of a trajectory is often broken by serious occlusion among objects, and the velocity, orientation features of objects are not as reliable as those in surveillance video. We take advantage of the depth data to establish the pedestrian relation in and out of layers to address the occlusion problem. The proposed layered graph model integrates the vision and depth features to track multi-pedestrian in a unified framework, improving reliability in dynamic backgrounds. In addition, this approach emphasizes an online real-time performance while most trajectory-level

Manuscript received June 13, 2014; revised October 10, 2014 and February 11, 2015; accepted April 9, 2015. Date of publication May 13, 2015; date of current version September 25, 2015. This work was supported in part by the National Basic Research Program of China (973 Program) under Grant 2011CB706900 and in part by the National Science Foundation of China under Grants 61039003, 61271433, and 61202323. The Associate Editor for this paper was H. Huang. (*Corresponding author: Qixiang Ye.)*

The authors are with the School of Electronic, Electrical, and Communication Engineering, University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: qxye@ucas.ac.cn).

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.2015.2423709

association methods [22]–[25] use an off-line strategy to connect the pedestrian observations in all frames. The contributions described in this paper are as follows: 1) A layered graph model using RGB-D data for multi-pedestrian tracking in traffic scenes, 2) An improved layer-level data association approach, and 3) An effective occlusion handling strategy specified for RGB-D based tracking.

The remainder of this paper is organized as follows: the related work is described in Section II. The proposed layered graph formulation is presented in Section III. The occlusion handling is introduced in Section IV. Section V describes the LGM optimization method. Experimental results and conclusions are presented in Sections VI and VII.

II. Related Work

This work is related to visual tracking, i.e., graph-based data association. It is also related to trajectory-level analysis and occlusion handling.

Graph-Based Data Association: Data association may be formulated as a graph model, in which object observations in frames are graph nodes and connections among the observations in adjacent video frames are graph edges. The outputs are represented by several subgraphs of the inputs in which observations of the same objects are connected. When solving the data association problem, bi-partite graph matching and K-partite graph matching [26]–[28] which consider global optimal matching in fixed-size temporal windows, i.e., the network flow, are typically used. Zhang *et al.* [29] mapped the Maximum-A-Posteriori (MAP) data association problem into a cost-flow network with a non-overlap constraint on trajectories for multi-object tracking. Berclaz *et al.* [30] used the K-shortest path algorithm to reach global optimal matching. Pirsiavash *et al.* [31] proposed a globally-optimal greedy algorithm to search for the successive shortest paths by defining a residual graph in the network. Brendel *et al.* [32] presented the tracking progress as finding the maximum weight-independent set of the graph in every two consecutive frames. The above approaches are usually effective in cases of partial occlusions, i.e., surveillance video with a specific top-view angle. In cases of serious occlusions in traffic scenes, however, they fail to perform long-term tracking. That is because the heads-up view is the most common in traffic scenes, where full occlusion issues in dynamic backgrounds are frequent.

Trajectory-Level Analysis: A large number of tracking approaches are related to trajectory-level analysis. Yang *et al.* [23], [33] used a trajectory-based Conditional Random Field (CRF) energy function to learn the affinity and dependency among the object observations online. Andriyenko *et al.* [24] used a continuous energy minimization method with gradient descent and greedy discontinuous jumps to explore scattered areas of the solution space. Milan *et al.* [34] proposed a discrete-continuous CRF model, which used detection- and trajectory-level constraints to distinguish objects. Wen *et al.* [35] adopted a tracklets-dense neighborhoods searching strategy in relation graph to guarantee the trajectory smoothness affinity. When applying these approaches in dynamic traffic scenes, however, the results may be unsatisfactory. Dynamic backgrounds break the continuity of trajectories, making the objective function converge to a local minimum, which results in tracking failure.

Occlusion Handling: Occlusion handling in multi-pedestrian tracking approaches is primarily divided into two classes, according to whether the occlusion relationship or the depth ordering of objects is inferred. The first class is designated as the "implicit" model [36]–[38]. For example, Enzweiler *et al.* [36] utilized local head, torso, and leg detectors combined in a mixture-of-experts framework and leveraging stereo and flow cues. Kwak *et al.* [37] inferred occluded regions with a patch classifier and improve tracking performance. These methods are sometimes available because partial features of occluded objects are extracted in partial occlusion issues, which may fail in cases of serious occlusion. Few of these approaches model the interaction among different targets. The second class is an "explicit" model, in which the occlusion relation among objects or depth ordering of objects is explicitly considered, e.g., [22], [39]. Such models competently handle occlusions, but the pedestrian trajectories must then be obtained when all the frames from a video sequence have been analyzed, which fails to meet the real-time requirement of the ADAS application.

Most relevant works are from [40] and [22], where Ablavsky *et al.* [40] and Zamir *et al.* [22] defined graph models to formulate the multi-pedestrian tracking process. The former defined pedestrian position and background as different graphical model layers in a static parking surveillance video. However, this background modeling method is not available for dynamic backgrounds in real driving scenarios. The latter defined a full-connected graph to connect all of the object detection within a temporal window. With the generalized minimum clique optimal algorithm in a fully-connected graph, tracklets are calculated globally. Nevertheless, this off-line strategy is not real-time.

III. Modeling Multi-Pedestrian Tracking

Given a video sequence with depth data, we first calculate pedestrian observation regions in RGB and D domains with an off-the-shell detection model [41]. From the observation regions, the combined features are extracted to describe pedestrians in terms of their 3-D position, appearance, and motion characteristics. We re-formulate the multi-pedestrian tracking problem as a novel layered graph model (LGM). As shown in Fig. 1, we introduce the layering and occlusion handling strategies based on RGB-D data to deal with the serious occlusion issues. In the LGM, the node, edge, and layer elements respectively represent the observation, feature similarity, and depth partition. We minimize the cost function using a heuristic searching algorithm in the LGM to approximate optimality, and achieve multiple pedestrians tracking.

A. Layered Graph Model

Multi-pedestrian tracking is formulated as a data association problem, which finds accurate tracklets in a layered graph $G = (N, L, E)$, where N, L, and E respectively denote the set of nodes, layers, and edges.

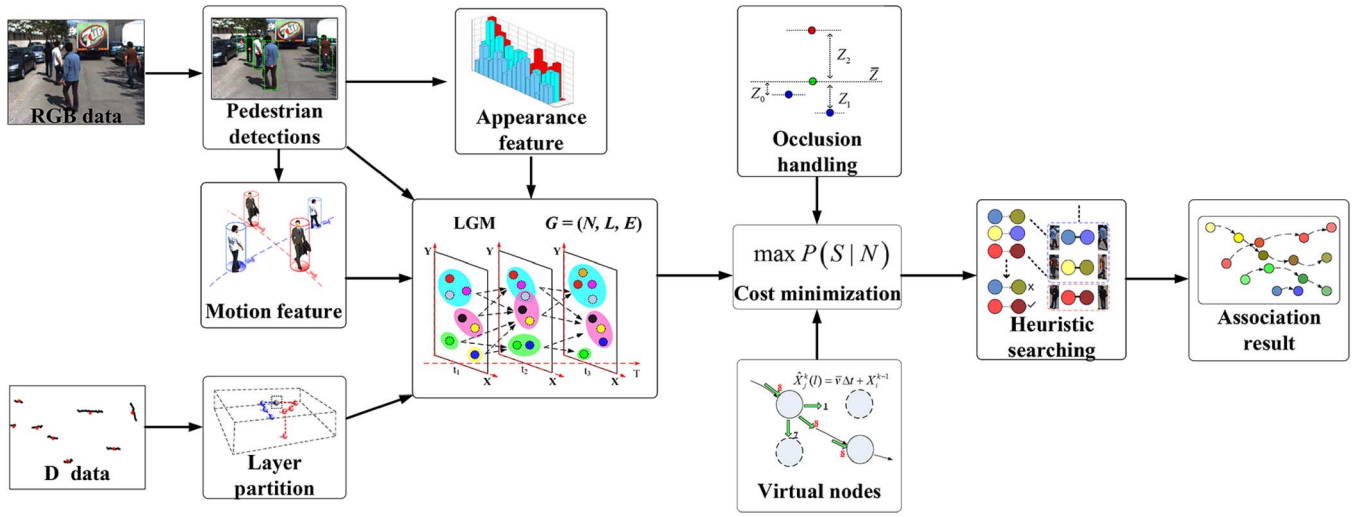
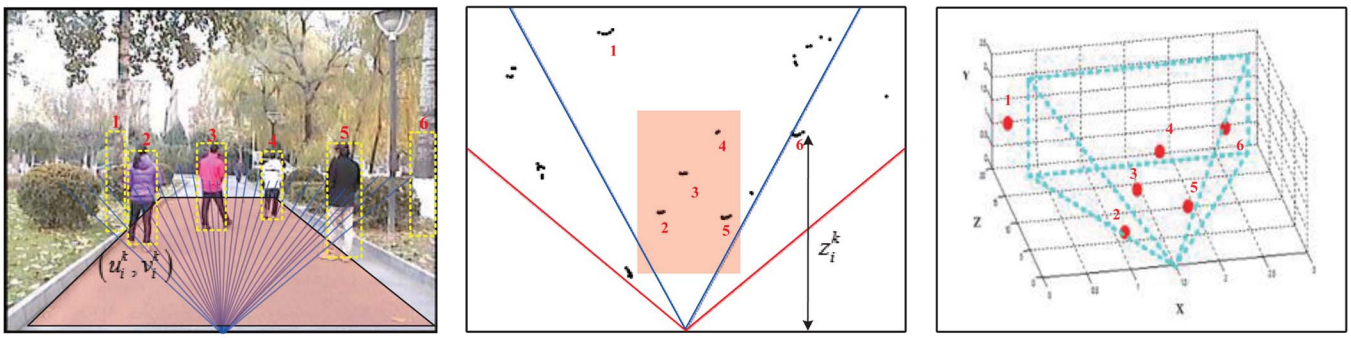


Fig. 1. Framework of our approach.



(a) (b) (c)

Fig. 2. (a) Pedestrian observations in the RGB domain and (b) observations in the D domain. The red and blue lines denote the scanning range from the D and RGB sensors, respectively, (c) observations are projected to 3-D space which combines the RGB and D data.

Node: N consists of K disjoint parts. Each part represents one frame. The nodes in the LGM represent the object observations. n_i^k denotes the ith node in the kth frame, where $i \in \mathbf{Z}^+$: $1 \leq k \leq K$. A node (observation), n_i^k, is associated with the 3-D spatial feature $X_i^k = t(u_i^k, v_i^k, z_i^k)$, the appearance feature φ_i^k, and the motion feature θ_i^k. X_i^k denotes a three-dimensional spatial coordinate of the center of an observation, therein (u_i^k, v_i^k) denotes the observation's center in the RGB domain as Fig. 2(a), and z_i^k denotes the depth value as Fig. 2(b). φ_i^k represents which is a unified vector combining the Histogram of Oriented Gradient (HOG) and color features (HOGC) [42]. θ_i^k denotes the motion feature including velocity and orientation.

Layer: L indicates that the nodes in the same frame are divided into l layers, and l_i^k denotes the layer's ID which the ith node in the kth frame belongs to, where $1 \leq l_i^k \leq l$. A red dashed rectangle in Fig. 3 denotes a layer, which contains at least one node. The nodes in l layers are represented as $N^{(1)}$, $N^{(2)}, \cdots, N^{(m)}$.

Edge: Edge E in the layered graph is defined as $E = \{(n_i^{k-1}, n_j^k) | \ |l_i^{k-1} - l_j^k| \leq 1\}$, denoting that the nodes in adjacent frames and neighboring layers are connected, as shown in Fig. 3. Note that not all of the nodes in adjacent frames are connected by edges. The weight of an edge between two nodes, $\omega(n_i^{k-1}, n_j^k)$, means the affinity between two nodes, which contains the information from above X_i^k, φ_i^k, and θ_i^k features.

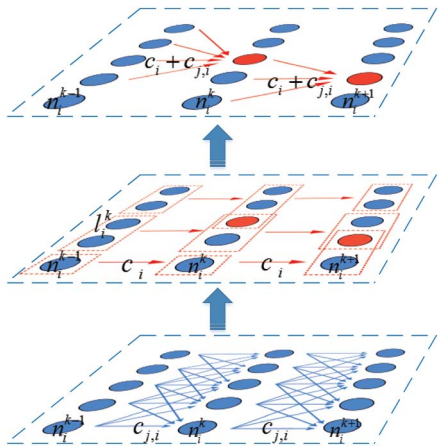


Fig. 3. The layered graph model. A fully-connected graph is in the bottom part, where all the nodes in the adjacent frames are connected, and $C_{j,i}$ is the edge cost. In the middle part, the nodes are divided into different layer marked with red dashed rectangles. The edges connecting the nodes exist inside of the layer. The red nodes in the overlap regions belong to adjacent layers. In the upper part, the edges connecting the nodes of overlap regions exist in the adjacent frames.

The output of the LGM is to find the trajectory of a particular pedestrian and identify them in each frame. Thus a feasible solution to this problem is represented by a subgraph from the entire graph G. Each subgraph therein denotes a feasible

solution, $G_s = (N_s, L_s, E_s)$, which contains one trajectory of a pedestrian (node) in a video sequence. A subgraph contains a set of nodes $N_s = \{n_a^1, n_b^2, n_c^3, \cdots\}$, denoting that the ath node from the 1st frame, the bth node from the 2nd frame, and the subsequent nodes are selected to be in N_s. $L_s = \{l_a^1, l_b^2, l_c^3, \cdots\}$ records each node's layer label. By this definition, $\mathrm{E}_s = \{\mathrm{E}(n_a, n_b) | n_a, n_b \in N_s\}$. Each subgraph G_s represents the tracklet of a single pedestrian. The entire graph G contains a set of tracklets, corresponding to the set of subgraphs.

B. Maximum a Posteriori

Following the maximum-a-posteriori (MAP) formulation [29]–[31], [43], a single trajectory hypothesis is written as a set of object observations, $s^k = (n_a^0, n_b^1, \ldots, n_i^k)$, where $n_i^k \in N$. An association hypothesis $S = \{s^k\}$ is composed of single trajectory hypotheses. The objective function of data association is to maximize the posteriori probability of S, given a set of object observations set N,

$$S^* = \arg\max_S P(S|N) \propto \arg\max_S P(N|S)P(S). \quad (1)$$

Assuming that the motions of all pedestrians are independent of one another, the likelihood probabilities are conditionally independent given the hypothesis. Equation (1) is decomposed as

$$S^* \propto \arg\max_S \prod_i P(n_i|S) \prod_{s^k \in S} P(s^k). \quad (2)$$

Supposing that $S = -\log S^*$, Eq. (2) is equivalent to

$$S = \arg\min_S \sum_i -\log P(n_i|S) + \sum_{s^k \in s} -\log P(s^k) \quad (3)$$

where $P(n_i|S)$ is the likelihood of the layer partition, which describes the depth variation of the observations in the same layer. This likelihood term is formulated as

$$P(n_i|S) = \begin{cases} \beta_i, & \text{if } s^k \in S, n_i \in s^k \\ 1 - \beta_i, & \text{otherwise} \end{cases} \quad (4)$$

where β_i is a depth factor, which describes the depth relation of an observation with the others in the neighboring layer. Note that this factor is updated in each frame, which makes occlusion handling simple but efficient when coping with the serious occlusion issues. This will be detailed in Section IV-A.

In Eq. (3), $P(s^k)$ is modeled as the link probability between two observations in successive frames. It's defined as the edge weight, $P(s^k) = \omega(n_i^{k-1}, n_j^k)$, $k = 1, 2, \ldots, K$, which measures the affinity of appearance, orientation, and 3-D motion.

$$\begin{aligned} &\omega\left(n_i^{k-1}, n_j^k\right) \\ &= P\left(n_i^{k-1} | n_j^k\right) \\ &= \begin{cases} A_{app}\left(\varphi_i^{k-1}, \varphi_j^k\right) A_{ori}\left(\theta_i^{k-1}, \theta_j^k\right) \\ \quad \times A_{mot}\left(X_i^{k-1}, X_j^k\right), & \text{if } \left|l_i^{k-1} - l_j^k\right| \le 1 \\ 0, & \text{otherwise.} \end{cases} \end{aligned} \quad (5)$$

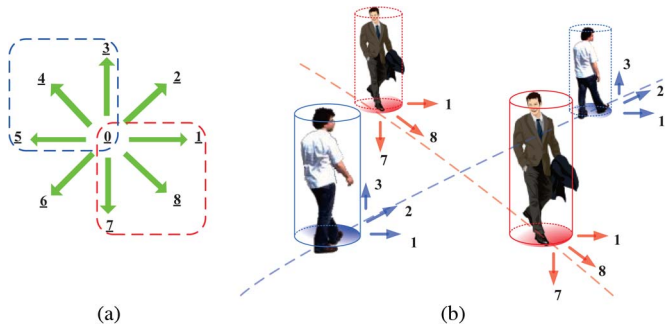


Fig. 4. (a) Orientation feature and (b) motion feature.

The appearance affinity among two observations is defined as a Gaussian distribution

$$A_{app}\left(\varphi_i^{k-1} | \varphi_j^k\right) = G\left(sim\left(\varphi_i^{k-1}, \varphi_j^k\right); 0, \delta_c\right) \quad (6)$$

where φ_i^{k-1} and φ_i^k are HOGC features [42], $sim(\varphi_i^{k-1}, \varphi_j^k)$ calculates the feature correlation between φ_i^{k-1} and φ_j^k. We divide the pedestrian's moving orientation into 9 bins as Fig. 4(a). Therein, 8 bins express different orientations with the resolution of 45°. 0 indicates that a pedestrian remains still in two consecutive frames. We define the orientation affinity as

$$A_{ori}\left(\theta_i^{k-1} | \theta_j^k\right) = \left|\theta_i^{k-1} - \theta_j^k\right| \quad (7)$$

where $|\theta_i^{k-1} - \theta_j^k|$ is the distance between two orientation bins θ_i^{k-1} and θ_j^k. The motion affinity is defined as

$$A_{mot}\left(X_i^{k-1} | X_j^k\right) = G\left(X_j^{k-1} + \bar{v}; X_i^k\right) G\left(X_i^k - \bar{v}; X_j^{k-1}\right) \quad (8)$$

where $\bar{v}$ is the average velocity of the last K frames of the pedestrian, whose moving orientation goes along the bin with the highest probability in Fig. 4. The difference between the predicted position and the observed position is assumed to obey Gaussian distribution.

Because that each observation belongs to one trajectory, two 0-1 indicator variables $e_{j,i}$ and e_i are defined to couple the non-overlap constraints, as

$$e_{j,i} = \begin{cases} 1, & \text{if } n_j^k \text{ is right after } n_i^{k-1} \\ 0, & \text{otherwise} \end{cases} \quad (9)$$

$$e_i = \begin{cases} 1, & \text{if } n_i^k \in s^k \\ 0, & \text{otherwise} \end{cases} \quad (10)$$

where $e_{j,i}$ means that if node n_i at the kth frame is associated with node n_i at the $(k-1)$th frame, there is an edge connecting them ($e_{j,i} = 1$). Otherwise, there is no edge between them. In the same way, the indicator e_i expresses whether the node n_i belongs to the particular trajectory hypothesis s^k. S is non-overlap if and only if $e_i = \sum_j e_{j,i} \le 1$. Based on the indicator variables, an objective cost function is obtained by adding these variables into Eq. (3), as

$$S = \arg\min_s \sum_{i,j} c_{j,i} e_{j,i} + \sum_i c_i e_i. \quad (11)$$

The directed edges are assigned with the cost values

$$\begin{cases} c_{j,i} = -\log P(s^k) \\ c_i = -\log P(n_i|S) = \log \frac{1-\beta_i}{\beta_i} \end{cases} \tag{12}$$

where $c_{j,i}$ corresponds to $e_{j,i}$, and denotes the cost of similarity between pedestrian observations n_j^k and n_i^{k-1}. c_i corresponding with e_i, is the cost reflecting 3-D spatial relation in a layer in the proposed LGM.

C. Layer-Level Constraint

To avoid any "unsafe" association and to reduce the computational complexity, we introduce a layering strategy, which incorporates the depth data as cue. For dynamic backgrounds, the conventional trajectory-level constraints [23], [24], [33], [34] is not as reliable as that in static backgrounds. Therefore, full advantage of D data z_i^k is taken, with the 3-D spatial feature X_i formulated as the layer constraint. We define a root-mean-square deviation, $\varepsilon_i^{(m)}$, which denotes the offset from an observation's 3-D position to the layer's average 3-D position, as

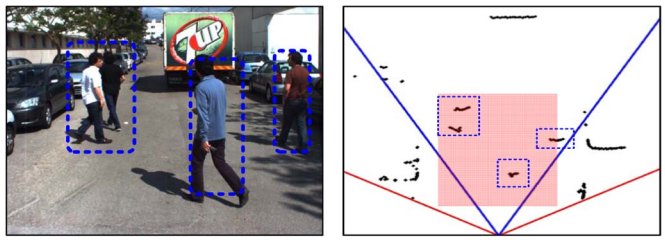
$$\varepsilon_i^{(m)} = \left(\frac{1}{|N^{(m)}|} \sum_i \left\| X_i^{(m)} - \bar{X}^{(m)} \right\|^2 \right)^{1/2} \tag{13}$$

where, $m = 1, 2, \ldots, M$ is the layer's index, $\bar{\mathbf{X}}^{(m)}$ is the average 3-D position of the mth layer in world coordinates, and $|N^{(m)}|$ denotes the number of nodes in the mth layer. By this definition, the observations in a dense region with high probability of occlusion are divided into the same layer. When starting the tracklets association, the observations in the same layer are the first searched. The layer regions are allowed to be overlapped, thus several observations may be owned by more than one layer. Hence, the observations in the overlap region are associated in neighboring layers. Therefore, the nodes could be divided into m layers as $N^{(1)}, N^{(2)}, \ldots, N^{(m)}$. The relationship among different layers is concluded as $N^{(1)} \cup N^{(2)} \cup \cdots N^{(m)} = N$, and $N^{(1)} \cap N^{(2)} \cap \cdots N^{(m)} \geqslant \emptyset$. The objective function Eq. (11) is written as

$$\mathbb{S} = \arg\min \sum_{m=1}^{M} \sum_{i,j} c_{j,i}^{(m)} e_{j,i}^{(m)} + \sum_{m,l}^{M} \sum_{i,j} c_{j,i}^{(m,l)} e_{j,i}^{(m,l)} + \sum_i c_i e_i. \tag{14}$$

Edges in the entire graph split into two: inter- and intra-layer edges, which respectively correspond to the first and the second terms in Eq. (14). The inter-layer edge connection $e_{j,i}^{(m)}$ indicates edges just exist in the mth layer, and the intra-layer edge connection $e_{j,i}^{(m,l)}$ indicates that edges exist between nodes in the mth and the lth layers, where $|m - l| \leq 1$. Notice that the number of edges in Eq. (14) decreases greatly compared with that in Eq. (11). That is because that the number of edges in each layer is much small than that of the entire graph, $\sum_m \sum_{i,j} e_{j,i}^{(m)} \ll \sum_{i,j} e_{j,i}$ and $\sum_{m,l} \sum_{i,j} e_{j,i}^{(m,l)} \ll \sum_{i,j} e_{j,i}$. In this inference, we can find that the layer-level constraint compresses the search space in LGM.

To resolve the objective function in Eq. (14) is equivalent to finding a min-cost path in the LGM. In the formulation, the objective function is mapped into a complete layered K-partite graph in Section III-A. The observations are represented by the nodes in the layered graph over successive frames, and the indicator variables e_i and $e_{j,i}$ denote the connective relation of the nodes, which determines whether the edge between two nodes exists. The search space is divided into M layers, and the nodes in each layer have a closer position relationship.



(a) (b)

Fig. 5. (a) Observations in the RGB domain and (b) observations in the D domain.

A specific case is shown in Fig. 5. Four pedestrians walking in front of a car have been partitioned into three layers according to Eq. (13), and the left two pedestrians in the red dashed rectangle are grouped into one layer, because they are close to each other, particularly in the depth domain. The occlusion issue frequently occurs in the layer containing multiple pedestrians. By contrast, the pedestrians in the blue dashed rectangle are divided into two separate layers according to Eq. (13). The edge between the nodes in the layer was not established without an overlap region when mapping to the LGM. In addition, there is no edge between the nodes in the red and blue layers in the case of Fig. 5, which greatly decreases the graph complexity in the searching progress.

IV. Occlusion Handling

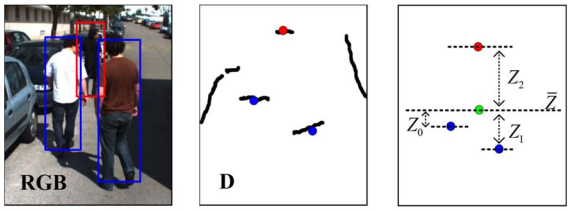
The occlusion handlings in much previous work [22], [39], i.e., the overlap field rate calculation method [39] and the adding hypothetical node method [22], focus on strategies in the image domain. However, they are challenged with serious occlusion in traffic scenes. Fig. 4 illustrates a serious occlusion case. The pedestrian in the red rectangle has been occluded partially, and is more seriously occluded in the following frames. For the occluded pedestrian in this case, when the appearance feature has little significance and the motion estimation provides an inaccurate velocity value, the conventional RGB-based tracker not only loses the object, but also has ID switch error. The essential reason is that the model lacks accurate spatial information from the depth domain.

A. Depth Factor

To deal with this issue, we use an updating depth factor β_i to reflect the depth variation of each node in the LGM, as

$$\beta_i = \frac{1}{1 + \exp\left(\frac{1}{\mathrm{M}} \sum_{j=1}^{M} z_j^{k-1} - \hat{z}_i^k \right)} \tag{15}$$

where, $\hat{z}_i^k = z_i^k - \bar{z}^k$, denotes the relative depth value between an observation's depth z_i^k and its layer's average depth $\bar{z}^k$. z_j^{k-1} denotes the depth value of the node in the neighboring layers in



(a) (b) (c)

Fig. 6. (a) Observations in the RGB domain, (b) observations in the depth domain, and (c) occlusion handling using the depth information.

the $(k-1)$th frame. Again the occlusion case in Fig. 4 serves as an example. The green dashed line (in Fig. 6(c)) is the layer's average depth value $\bar{z}^k$. According to Eq. (15), the depth factor $\beta_i, i = 1, 2, 3$ is obtained. The observation with a small depth value (marked with blue points in Fig. 6(c)) satisfies: $0 < \beta_i < 0.5$; and the observation with a large depth value (marked with red points Fig. 6(c)) satisfies: $0.5 < \beta_i < 1$. When substituting the β_i in Eq. (12), a small depth value exists with a low cost. Consequently, a node with a small depth value is associated before that with a large depth value. Based on Eqs. (12) and (15), the LGM updates the depth variation in each frame, and assigns the edges with the discriminative costs in the layered graph.

The case in which one layer has a single observation is shown in blue dashed rectangles in Fig. 5. In such a case, the layer's average depth, $\bar{z}^k = 0$, thus the depth factor, $\beta_i \approx 0.5$. Invoking Eq. (12), the cost $c_i \approx 0$, suggesting that if a layer has a single observation, its depth cost has little influence on the edge cost, whose cost primarily depends on the affinity measurement.

B. Virtual Nodes

In some cases, a given layer may lack enough observations due to full occlusion or missed detection. In order to address this issue, we add a virtual node to that layer. If one frame does not include any appropriate detection, the virtual node is selected.

The spatial coordinates of the virtual nodes are computed using the estimation motion model $\hat{X}_j^k(l) = \bar{v}\Delta t + X_i^{k-1}$, where the $\bar{v}$ means the average velocity in the last K frames, $\hat{X}^k(l)$ means the spatial location of the virtual nodes in layer l. The moving orientation of the virtual node is along the orientation bin in the last frame as in Fig. 4(a). Because the virtual nodes typically have larger depth values than normal nodes in a layer, we design a constant penalty in the weights of edges connected to the virtual nodes in G. This avoids selecting the virtual nodes if frames contain an appropriated observation.

The virtual nodes are updated at the end of each iteration, when solving the optimization problem of Eq. (14). In the first few iterations, virtual nodes are not likely to be selected as the algorithm continues selecting the existing detections. However, as the optimization process progresses, the clusters which include correct detections are exhausted and the virtual nodes will contribute until the algorithm converges to the final solution G_s.

V. Model Optimization

To solve the proposed LGM, we adopt a heuristic strategy to approximate the optimal solution of Eq. (14), and thereby find the near-best edge combination $\mathrm{E}_s = \{\mathrm{e}_j, e_{j,i} | n_j, n_{j,i} \in N_s\}$. The cost function Eq. (14) with polynomial structure provides an acceptable initial solution after layer partition, when the edges are searched along the depth-cost augmented way, c_i, in the graph. This ensures that any distant pedestrian at an occluded position would not be falsely associated with the occluder, consequently reducing the ID-switching error.

It is known that a good initial solution can yield convergence to a better optimum. Our heuristic searching strategy further improves the solution to a near optimum solution with the iteration constrained by the layer partition. We limit the trajectory search space in neighboring layers, which corresponds to the edges satisfying the Eq. (13). Such a layer-level constraint excludes the invalid edges connecting two layers with a large depth-span, ensuring the edge searching in a much smaller space. The issue that one layer has a single observation further decreases the complexity. Then we use a switching labels algorithm to match the tracklets in the layered graph. For the selected node in each iteration, it is attempted to switch labels with the node in the neighboring layers. If the new overall cost is lower, the change is retained. Given an initial solution, the cost minimization algorithm is shown in Algorithm 1.

TABLE I
Evaluation Metrics

Items	Definition
Recall(↑)	Correctly matched detections / total detections in ground truth
Prec.(↑)	Correctly matched detections / total detections in the tracking result
GT	Number of positions in ground truth
MT(↑)	The ratio of mostly tracked trajectories, which are tracked for more than 80%
ML(↓)	The ratio of mostly lost trajectories, which are tracked for less than 20%
PL(↓)	The ratio of partially lost trajectories, which are tracked in 20-80%
Frag.(↓)	Fragments, the number of times that a ground truth trajectory is interrupted
IDS(↓)	ID switch, the number of times that a tracked trajectory changes its matched ID

Note: For the items with ↑, higher scores indicate better results, for those with ↓, lower scores indicate better results.

Algorithm 1 Switching labels for multi-pedestrian tracking

Input: Layered graph G; cost $\{c_i\}$, $\{c_{j,i}\}$;
Output: Edge combination E_s of G;
Initialization: Finding the edge combination E_s with the lowest cost by a greedy algorithm and calculating its overall cost c_{over} by Eq. (12);
1: **for** $i < N$ **do**
2: Set minimum cost $c_{\min} = +\infty$;
3: **for** $j = i, \cdots, k$ **do**
4: -Switch label of n_i^{k-1} and n_j^k under constrains Eqs. (9) and (10), then evaluate new cost c_{temp};
5: -If $c_{temp} < c_{\min}$, $c_{over} = c_{temp}$;
6: **end for**
7: $c_{over} < c_{\min}$, $c_{\min} = c_{over}$, update E_s with this switch;
8: **end for**

TABLE II
COMPARISONS OF FIVE DATA SETS

Dataset	Method	Recall	Prec.	GT	MT	PL	ML	Frag.	IDS
SYNC	Berclaz et al. [30]	69.6%	74.8%	66	64.5%	22.7%	12.8%	45	23
	Andriyenko et al.[24]	73.4%	78.3%	66	69.7%	19.7%	10.6%	39	18
	Milan et al. [34]	75.6%	80.2%	66	71.2%	18.2%	10.6%	37	16
	NN	54.5%	64.3%	66	45.5%	30.3%	24.2%	52	31
	k-partite	73.1%	78.6%	66	68.2%	15.2%	16.6%	39	17
	LGM	**85.0%**	**89.7%**	66	**80.3%**	**10.6%**	**9.1%**	**21**	**7**
SDL-1	Zhang et al. [29]	67.8%	72.5%	10	60.0%	20.0%	20.0%	7	5
	Andriyenko et al. [24]	76.6%	81.0%	10	70.0%	20.0%	10.0%	5	4
	NN	56.4%	64.9%	10	40.0%	30.0%	30.0%	8	6
	k-partite	68.7%	73.5%	10	60.0%	20.0%	20.0%	5	4
	LGM	**94.5%**	**98.3%**	10	**90.0%**	**10.0%**	**0.0%**	**2**	**0**
SDL-2	Berclaz et al. [30]	68.9%	74.5%	92	60.9%	17.4%	21.7%	58	31
	Andriyenko et al. [24]	70.4%	76.4%	92	63.0%	20.7%	16.0%	51	29
	Yang et al. [33]	72.3%	77.8%	92	64.1%	21.7%	14.2%	47	26
	NN	59.4%	65.6%	92	43.5%	23.9%	32.6%	69	38
	k-partite	67.0%	73.5%	92	59.8%	22.8%	17.4%	49	25
	LGM	**82.4%**	**87.3%**	92	**76.1%**	**15.2%**	**8.7%**	**28**	**14**
SDL-Campus	Zhang et al. [29]	76.4%	79.8%	74	71.6%	18.9%	9.5%	30	16
	Milan et al. [34]	80.0%	84.5%	74	75.7%	16.2%	8.1%	26	14
	NN	67.7%	74.6%	74	60.8%	21.6%	17.6%	37	19
	k-partite	78.3%	82.9%	74	73.0%	17.6%	9.4%	26	15
	LGM	**85.6%**	**89.3%**	74	**81.1%**	**12.2%**	**6.7%**	**14**	**8**
LIPD	Berclaz et al. [30]	72.8%	76.4%	77	68.9%	18.1%	13.0%	24	19
	Yang et al. [33]	77.6%	80.2%	77	72.7%	13.0%	14.3%	19	14
	NN	64.4%	70.1%	77	64.9%	15.6%	19.5%	27	19
	k-partite	71.9%	75.4%	77	70.1%	18.2%	11.7%	25	16
	LGM	**84.9%**	**88.3%**	77	**77.9%**	**11.7%**	**10.4%**	**16**	**10**

Time Complexity Analysis: In each iteration, it is necessary to find the min-cost path. We would like to adopt Dijstra's algorithm to compute the shortest path in $O(N \log N)$, making the overall algorithm $O(MN \log N)$, where M is the number of pedestrians, and N is the number of the nodes in the LGM. However, there are negative edges in our LGM (the layer matching cost c_i is likely to appear as a negative value). The heuristic search algorithm has a complexity of $O(MN^2)$. Therefore, the overall complexity is also polynomial. The nodes in the LGM are divided into different layers, the number of edges sharply decreased with increasing layer number. In experiments, the run time is nearly linear with regard to the number of pedestrians.

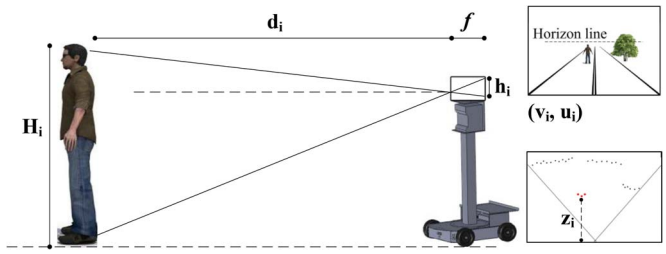


Fig. 7. The platform combining the camera and depth sensors. The two images on the right show the pedestrian observation in the image and depth domain respectively.

VI. EXPERIMENTS

Data Sets and Metrics: To demonstrate the effectiveness of the proposed approach, we performed experiments on five public data sets recorded in traffic scenes: the ISR-UC-sync data set [44], the SDL-1 and SDL-2 data sets [45], the SDL-Campus data set [45] and the LIPD data set [46]. All of the data sets are combined by the video and depth sequences from the camera and depth sensors, respectively. The dynamic backgrounds and heads-up vision have brought much challenge. The common occlusion cases in traffic scenes belong to full occlusion.

To evaluate the tracking performance, we adopt evaluation metrics [23], [25], [33], [47] defined in Table I, which are commonly used metrics when evaluating tracking methods. The items in the table measure not only the pedestrian's ID number, but also the long-term performance of a tracker.

Baseline Methods: In order to verify the accuracy and efficiency of our proposed approach, we systematically compare it with three kinds of state-of-the-art methods. The first kind of baseline represented vision-based methods, including the network flow approaches [29], [30], Andriyenko *et al.*'s continuous energy optimization approach [24], the online learned CRF model [33], the approach based on detection- and trajectory-level exclusion [34]. The second kind is the depth-based method, i.e., the Nearest Neighbor (NN) method, which incorporates the motion and position features to complete the trajectory association. The third kind contains both vision and depth data. It adopts the LGM based on the appearance, motion, and position features, however, without using the layering theory. In other words, all of the pedestrian observations are associated in a full-connected graph.

In the experiment, it is found that the depth cost is sufficient to find the appropriate tracklets when the layer has no overlap region in most data sets. When a layer contains several nodes with similar depth values, the similarity probability cost in Eq. (5) becomes a vital factor for distinguishing them. It is also found that the orientation feature is more informative than the velocity and motion features, especially when the pedestrians swap their positions in tracking. Table II shows quantitative comparisons. Figs. 8–10 represent the results of our approach.

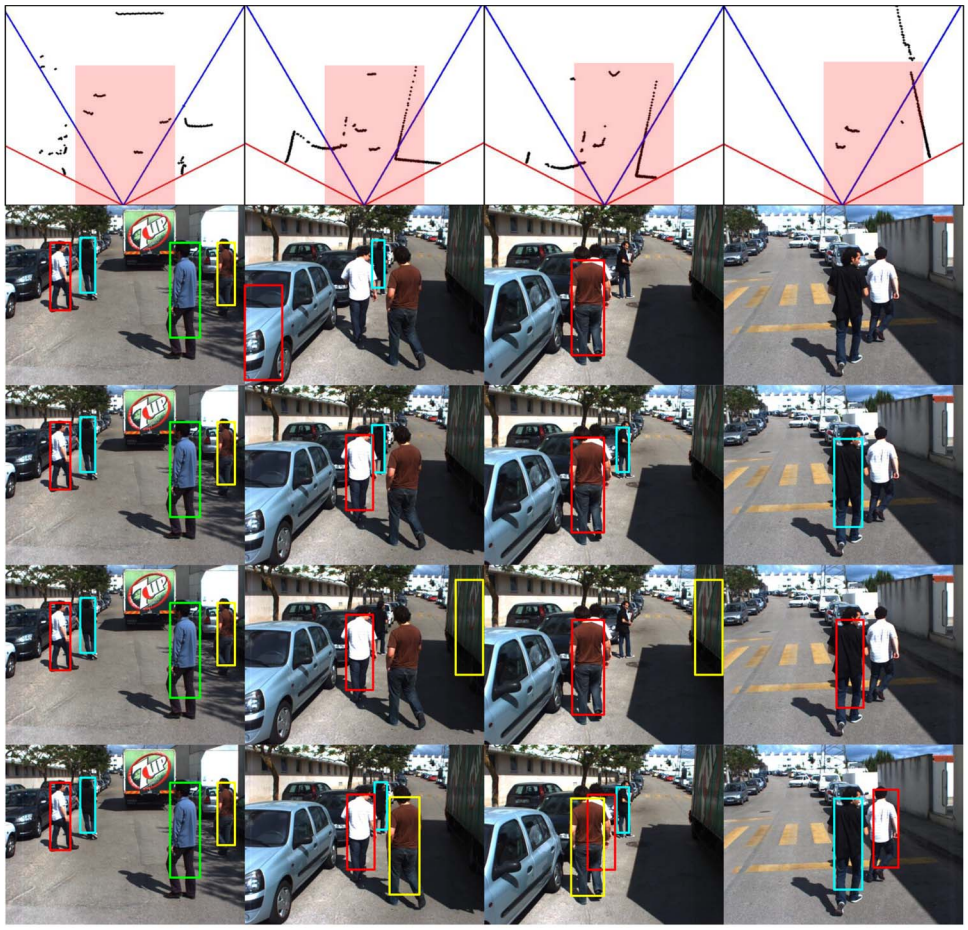


Fig. 8. Tracking examples in the SYNC data set. The first row shows observations in the depth domain. The red and blue lines denote the scanning range from the D and RGB sensors, respectively. The second row shows the tracking result from the "NN" method. It is observed that losing object error is great in the rightmost image. The third row shows the tracking result of the vision-based method [30], in which IDS and losing-object errors occurred. The fourth row shows the tracking result of the K-partite graph method. False detection has been added in the trajectory as the fourth object. The fifth row shows the tracking result of the proposed LGM method.

Parameter Setting: We empirically set $K = 4$ in the LGM, which means that the LGM is a 4-partite graph, corresponding to the successive four frames in the video sequence. We normalize appearance, position, and motion affinity function in Eq. (5) in order to make them comparable. The parameter offset, $\varepsilon_i^{(m)}$, is used to control the area of each layer and the overlap area between the layers. Although frame rates, resolutions and densities are different in the aforementioned data sets, we use the same parameter setting and ground truth data, such that the performance improves relative to previous methods for all of them. This indicates that our approach has a low sensitivity over parameters. The DPM detector [41] we used in the experiments is implemented in its generic, publicly available, pre-trained versions, and is not specifically trained for any data set sequence. Table II shows the comparison results of the above methods on the five data sets.

SYNC Data Set: The SYNC data set is a video sequence with 2147 frames. Long-term and serious occlusion issues are frequent. Fig. 8 illustrates the tracking examples. Cars parking along both sides of the road coincidently have similar colors with the pedestrians and are very close to the pedestrians, which poses great challenges to the track. In this sequence, frames ranging from 150 to 600 have multiple occlusion scenarios. It is found that the pedestrian detector outputs numerous false detections. Many of the false detections have been added to the trajectory by the vision-based methods in the first baseline, and the MT item in Table I decreases. That is because the recurring false and missing detections in dynamic traffic backgrounds make the affinity probability of appearance and motion unreliable, the false detections are not be excluded in the image domain. In this case, although the depth-based method "NN" excludes the false detection in the depth domain, it is unable to distinguish the pedestrians walking closely, due to lacking the appearance feature, therefore the IDS error increases. The depth factor, β_i, provides the LGM with another cue by incorporating the depth data. The large cost is assigned to the pedestrian at the distant place in the LGM, so it tends to be associated with the occluders in such a serious occlusion issue. The proposed LGM approach divides the observations into different layers by the depth factor. The appearance and motion features work within the neighboring layers. The LGM has the lowest ML, Frag., and IDS errors, as shown in Table II. In addition, the time utilized by the LGM is much lower than the K-partite method, because the layered graph limits the search space in neighboring layers, rather than in the entire graph.

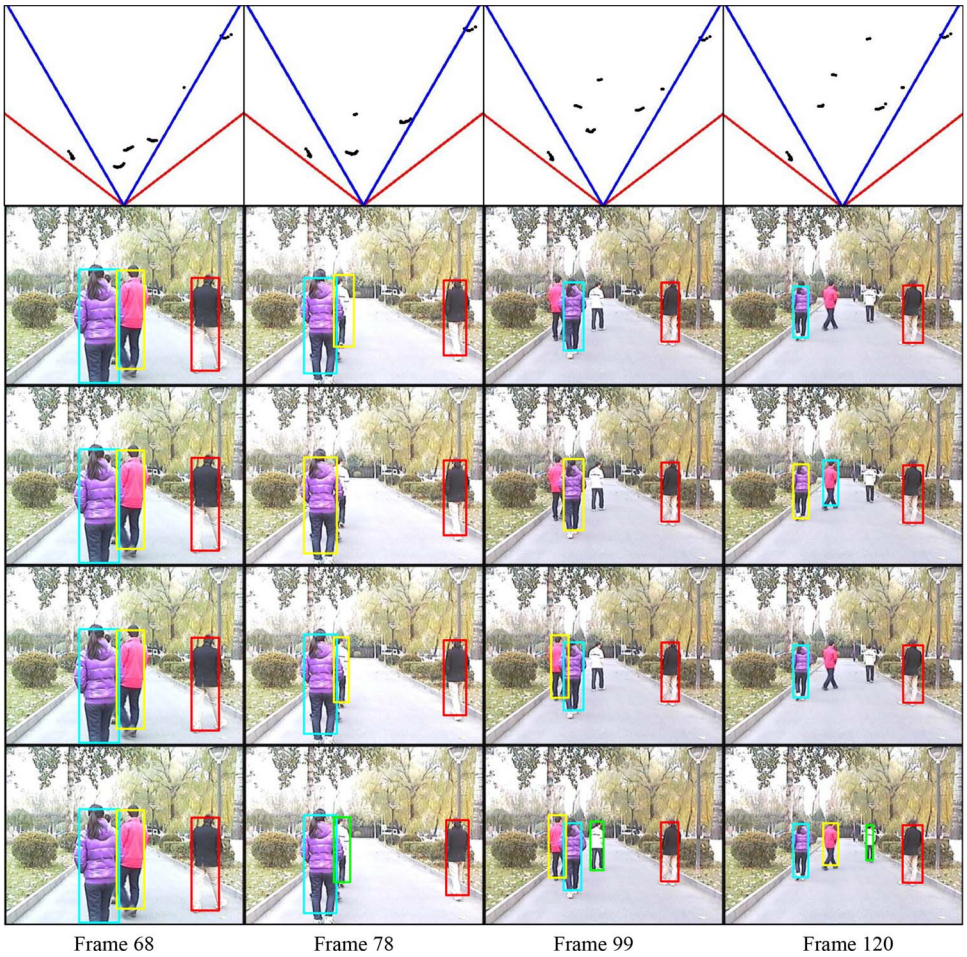


Fig. 9. Tracking examples in the SDL-1 data set. The first row shows observations in the depth domain. The red and blue lines denote the scanning range from the D and RGB sensors, respectively. The second row shows the tracking result of the "NN" method. There are several IDS and losing-object errors after occlusion. The third row shows the tracking result of the vision-based method [29]. The fourth row shows the tracking result of the K-partite graph method. The fifth row shows the tracking result of the proposed LGM method.

SDL 1-2 Data Sets: The SDL-1 and SDL-2 data sets [45] are recorded on a straight road and a crossroad. The vision and depth sensors are mounted at a height of 0.9 m to scan pedestrians at waist level. as shown in Fig. 7. In addition to pedestrians naturally walking on the road, another ten persons intentionally cross in front of the platform, in order to cast complex occlusion situations. Figs. 9 and 10 show the tracking results of the LGM. In Table II, it is found that the items Recall and Prec. increase compared with the results in the SYNC data set because the increase of pedestrians poses more serious occlusions, especially full occlusions, in a head-up view. The methods based on trajectory-analysis in the first baseline perform worse than the LGM. The trajectory-level tracking models often lose the objects during full occlusions, and swap the labels of pedestrians after serious occlusions, which results in more Frag. and IDS errors. At the same time, the item PL increases. However, the depth data prevents the tracker adding false observations without depth value into a complete trajectory. In the LGM, the layer-level constraint guarantees that the observations search in a consistent depth space. The depth span increases as well as the edge cost. The observations without depth data are assigned the maximum costs in the layered graph, which eliminated the edges between them. The LGM method has the least Frag. and no IDS errors in the SDL-1 data set, as well as less Frag. and IDS errors on SDL-2 data set. Therefore, the depth data provides another solid reference for the conventional 2D image, and improves the accuracy of the RGB-based tracking model.

SDL-Campus Data Set: We test our approach on the SDL-campus data set [45]. Compared with other data sets, the pedestrians in that data set are far from the platform, so the observation regions are smaller than those in the above data sets. We compare our models with Zhang *et al.*'s [29] network flow method and Milan *et al.*'s [34] detection- and trajectory-level exclusion method. Table II shows that the proposed LGM significantly improves the Recall and Prec. items. The samples of the tracking result are shown in Fig. 11.

LIPD Data Set: The LIPD data set [46] was recorded from the sensor acquisition system mounted on an instrument-equipped Yamaha vehicle, driving in an urban environment. It was equipped with an Ibeo laser scanner, and a monocular Guppy camera. Due to the fact that the data set was obtained around dusk, another challenge was marked light variation. The data set contains 4823 frames. We test all three baselines of methods using the frames containing multiple pedestrians. The comparative results are shown in Table II, which suggests that the LGM method is also consistent in poor light conditions.

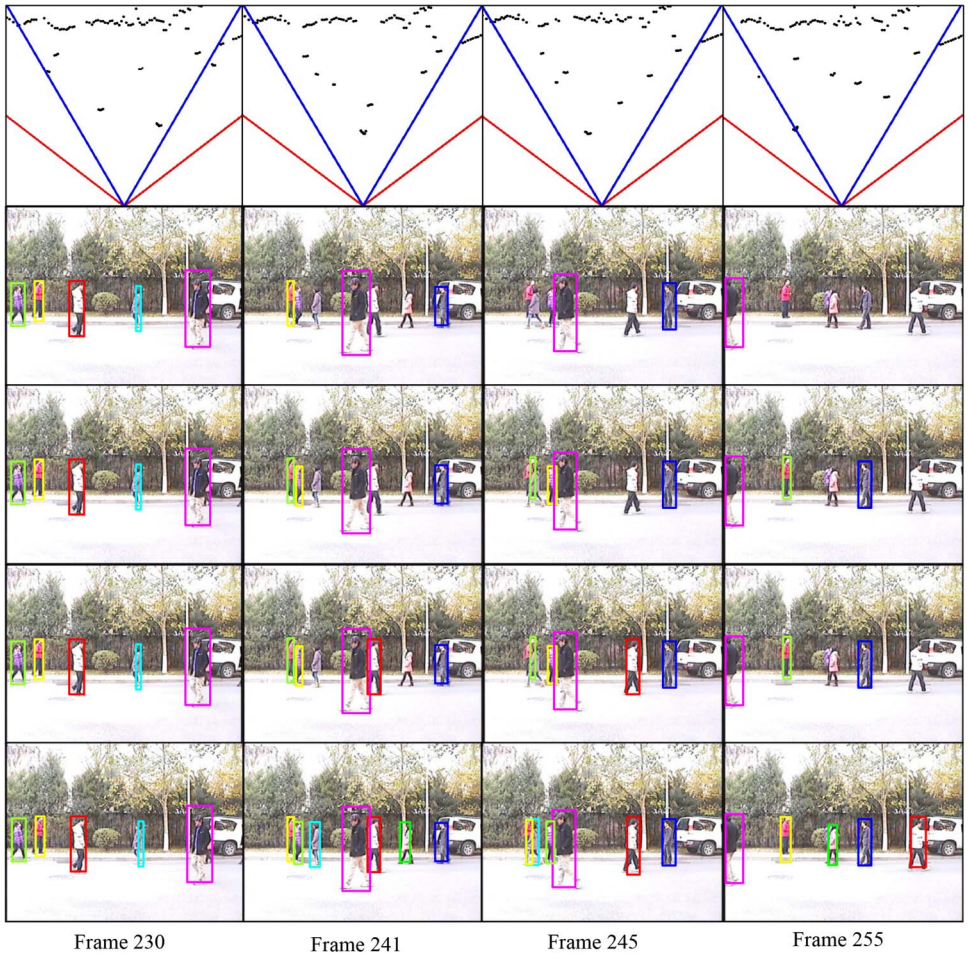


Fig. 10. Tracking examples in the SDL-2 data set. The first row shows observations the depth domain. The red and blue lines denote the scanning range from the D and RGB sensors, respectively. The second row shows the tracking result of the "NN" method. The third row shows the tracking result of the vision-based method [30]. The fourth row shows the tracking result of the K-partite graph method. The fifth row shows the tracking result of the proposed LGM method.

Fig. 11. Tracking examples in the SDL-Campus data set from the frame 1290 to 1413, which includes many pedestrians more than 25 meters away from the moving platform, their observation regions are smaller than those in the above data sets. The results come from the proposed LGM method.

In this condition, the RGB-based appearance feature does not make sense, because the appearances of targets have nearly fused with the dark background. Our model utilized the motion cue from the depth data to conduct the layer division, then obtains the orientation and 3-D motion affinity in Eq. (5), which promises that the orientation and motion affinity costs can be obtained and targets could be tracked, when lacking of the appearance feature. This further validates the robustness and efficiency of our model. Sample tracking is shown in Fig. 12.

Real-Time Performance: The proposed LGM approach is based on a layered graph, wherein the layer-level constraint decreases much search space in tracking. It seems that the feature extraction in the depth domain is time-consuming, however, the depth data provides a large advantage in decreasing the computing complexity in the data association process. Our experiments are performed on an Intel 3.4GHz PC with 4G memory, and the codes are implemented in Matlab. Without code optimization, our method achieves a rate of 40 fps for

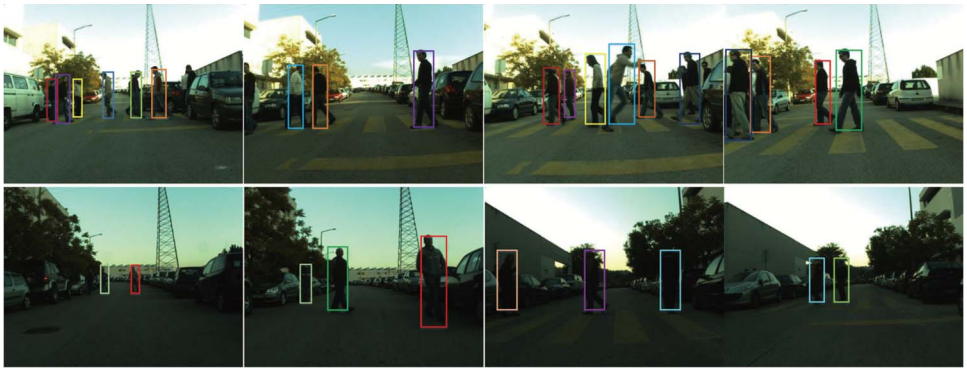


Fig. 12. Tracking examples in the LIPD data set. The results come from the proposed LGM model. The first row shows the tracking results in a zebra crossing containing frequent full-occlusion issues, and second row presents the results in a poor light condition.

robust tracking more than 10 pedestrians in crowded traffic scenarios, which is real-time performance.

Function of Depth Data: From the experimental results, the methods that utilize depth data outperform, especially when the pedestrians are under serious occlusion and in cluttered backgrounds. Pedestrian appearance changes significantly after rotation or deformation, making distinguishing difficult. The pedestrian motion feature is not reliable after multiple serious occlusions, which are the main reason of tracking failure for conventional RGB-based multiple object tracking. However, the depth feature remains distinguishable when the similarity vanishes in the RGB domain. For example, the pedestrian is partially occluded, the RGB-based detector sometimes locates the pedestrian with the foreground and/or background by error. This observation tends to be misconnected with the occluder or discarded by the tracker. However, the depth data provides the space information of the occluder and occludee.

VII. Conclusion

Multi-pedestrian tracking in traffic scenes is challenging due to cluttered backgrounds and long-term serious occlusions. Existing approaches that used RGB-D data in detection and tracking fail to form a complete association model with the depth data in traffic scenes. In this paper, we implemented an RGB-D multi-pedestrian tracking method by integrating the depth and vision data with a layer-level constraint, which was formulated as a layered graph model and solved by a heuristic searching algorithm. A 3-D occlusion handling strategy is proposed to solve the serious occlusion problem in and out of the layers with respect to the LGM. Extensive experiments on five public data sets demonstrate that our method is effective and accurate for multi-pedestrian tracking tasks in traffic scenes and thus advances the state-of-the-art.

Although the proposed LGM method is accurate and efficient, the unstable detection responses probably introduce false pedestrian observations. In the future, it may be effective to use a detector which combines the RGB and depth data together to improve the detection accuracy and thereby provide an integrated multiple pedestrian detection and tracking system. In addition, we will implement the LGM method in more datasets considering different weather factors, especially in rain and snow.

References

[1] M. M. Trivedi, T. Gandhi, and J. McCall, "Looking-in and looking-out of a vehicle: Computer-vision-based enhanced vehicle safety," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 1, pp. 108–120, Mar. 2007.

[2] C. Wojek, S. Walk, S. Roth, K. Schindler, and B. Schiele, "Monocular visual scene understanding: Understanding multi-object traffic scenes," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 4, pp. 882–897, Apr. 2013.

[3] L. Spinello and K. O. Arras, "Leveraging RGB-D data: Adaptive fusion and domain adaptation for object detection," in *Proc. IEEE Int. Conf. Robot. Autom.*, 2012, pp. 4469–4474.

[4] A. Ess, B. Leibe, and L. Van Gool, "Depth and appearance for mobile scene analysis," in *Proc. IEEE 11th Int. Conf. on Comput. Vis.*, 2007, pp. 1–8.

[5] M. Bajracharya, B. Moghaddam, A. Howard, S. Brennan, and L. H. Matthies, "A fast stereo-based system for detecting and tracking pedestrians from a moving vehicle," *Int. J. Robot. Res.*, vol. 28, no. 11/12, pp. 1466–1485, 2009.

[6] D. M. Gavrila and S. Munder, "Multi-cue pedestrian detection and tracking from a moving vehicle," *Int. J. Comput. Vis.*, vol. 73, no. 1, pp. 41–59, Jun. 2007.

[7] J. C. McCall and M. M. Trivedi, "Video-based lane estimation and tracking for driver assistance: Survey, system, and evaluation," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 1, pp. 20–37, Mar. 2006.

[8] T. Gandhi and M. M. Trivedi, "Computer vision and machine learning for enhancing pedestrian safety," in *Proc. Comput. Intell. Autom. Appl.*, 2008, pp. 59–77.

[9] A. Ess, B. Leibe, K. Schindler, and L. Van Gool, "Robust multiperson tracking from a mobile platform," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 10, pp. 1831–1846, Oct. 2009.

[10] T. Gandhi and M. M. Trivedi, "Vehicle surround capture: Survey of techniques and a novel omni-video-based approach for dynamic panoramic surround maps," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 3, pp. 293–308, Sep. 2006.

[11] D. Hoiem, A. A. Efros, and M. Hebert, "Putting objects in perspective," *Int. J. Comput. Vis.*, vol. 80, no. 1, pp. 3–15, 2008.

[12] D. Streller, K. Furstenberg, and K. Dietmayer, "Vehicle and object models for robust tracking in traffic scenes using laser range images," in *Proc. IEEE 5th Int. Conf. Intell. Transp. Syst.*, 2002, pp. 118–123.

[13] E. Prassler, J. Scholz, and A. Elfes, "Tracking people in a railway station during rush-hour," in *Proc. Comput. Vis. Syst.*, 1999, pp. 162–179.

[14] A. Mendes and U. Nunes, "Situation-based multi-target detection and tracking with laserscanner in outdoor semi-structured environment," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, 2004, vol. 1, pp. 88–93.

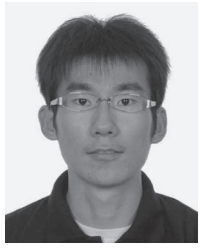
[15] M. Szarvas, U. Sakai, and J. Ogata, "Real-time pedestrian detection using LIDAR and convolutional neural networks," in *Proc. IEEE Conf. Intell. Veh. Symp.*, 2006, pp. 213–218.

[16] A. Broggi, P. Cerri, S. Ghidoni, P. Grisleri, and H. G. Jung, "A new approach to urban pedestrian detection for automatic braking," *IEEE Trans. Intell. Transp. Syst.*, vol. 10, no. 4, pp. 594–605, Dec. 2009.

[17] M. Mahlisch, R. Schweiger, W. Ritter, and K. Dietmayer, "Sensorfusion using spatio-temporal aligned video and lidar for improved vehicle detection," in *Proc. IEEE Conf. Intell. Veh. Symp.*, 2006, pp. 424–429.

[18] H. Noguchi, T. Mori, T. Matsumoto, M. Shimosaka, and T. Sato, "Multiple-person tracking by multiple cameras and laser range scanners in indoor environments," *J. Robot. Mechatron.*, vol. 22, no. 2, p. 221, 2010.

[19] Q. Baig, O. Aycard, T. D. Vu, and T. Fraichard, "Fusion between laser and stereo vision data for moving objects tracking in intersection like scenario," in *Proc. IEEE Intell. Veh. Symp.*, 2011, pp. 362–367.
[20] J. Cui, H. Zha, H. Zhao, and R. Shibasaki, "Multi-modal tracking of people using laser scanners and video camera," *Image Vis. Comput.*, vol. 26, no. 2, pp. 240–252, Feb. 2008.
[21] X. Song *et al.*, "An online system for multiple interacting targets tracking: Fusion of laser and vision, tracking and learning," *ACM Trans. Intell. Syst. Technol.*, vol. 4, no. 1, p. 18, Jan. 2013.
[22] A. R. Zamir, A. Dehghan, and M. Shah, "GMCP-Tracker: Global multi-object tracking using generalized minimum clique graphs," in *Proc. Eur. Conf. Comput. Vis.*, 2012, pp. 343–356.
[23] B. Yang, C. Huang, and R. Nevatia, "Learning affinities and dependencies for multi-target tracking using a CRF model," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2011, pp. 1233–1240.
[24] A. Andriyenko and K. Schindler, "Multi-target tracking by continuous energy minimization," in *IEEE Conf. Comput. Vis. Pattern Recog.*, 2011, pp. 1265–1272.
[25] A. Milan, K. Schindler, and S. Roth, "Challenges of ground truth evaluation of multi-target tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog. Workshops*, 2013, pp. 735–742.
[26] K. Shafique and M. Shah, "A noniterative greedy algorithm for multiframe point correspondence," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 1, pp. 51–65, Jan. 2005.
[27] G. Shu, A. Dehghan, O. Oreifej, E. Hand, and M. Shah, "Part-based multiple-person tracking with partial occlusion handling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2012, pp. 1815–1821.
[28] S. Gao, Z. Han, C. Li, and J. Jiao, "Real-time multi-pedestrian tracking based on vision and depth information fusion," in *Proc. Adv. Multim. Inf. Process.-PCM*, 2013, pp. 708–719.
[29] L. Zhang, Y. Li, and R. Nevatia, "Global data association for multi-object tracking using network flows," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2008, pp. 1–8.
[30] J. Berclaz, F. Fleuret, E. Turetken, and P. Fua, "Multiple object tracking using k-shortest paths optimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1806–1819, Sep. 2011.
[31] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2011, pp. 1201–1208.
[32] W. Brendel, M. Amer, and S. Todorovic, "Multiobject tracking as maximum weight independent set," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2011, pp. 1273–1280.
[33] B. Yang and R. Nevatia, "An online learned CRF model for multi-target tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2012, pp. 2034–2041.
[34] A. Milan, K. Schindler, and S. Roth, "Detection-and trajectory-level exclusion in multiple object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 3682–3689.
[35] L. Wen *et al.*, "Multiple target tracking based on undirected hierarchical relation hypergraph," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2014, pp. 1282– 1289.
[36] M. Enzweiler, A. Eigenstetter, B. Schiele, and D. M. Gavrila, "Multi-cue pedestrian classification with partial occlusion handling," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 990–997.
[37] S. Kwak, W. Nam, B. Han, and J. H. Han, "Learning occlusion with likelihoods for visual tracking," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 1551–1558.
[38] G. Cielniak, T. Duckett, and A. J. Lilienthal, "Data association and occlusion handling for vision-based people tracking by mobile robots," *Robot. Auton. Syst.*, vol. 58, no. 5, pp. 435–443, May 2010.
[39] A. Andriyenko, S. Roth, and K. Schindler, "An analytical formulation of global occlusion reasoning for multi-target tracking," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, 2011, pp. 1839–1846.
[40] V. Ablavsky and S. Sclaroff, "Layered graphical models for tracking partially occluded objects," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1758–1775, Sep. 2011.
[41] [Online]. Available: http://www.cs.berkeley.edu/rbg/latent/
[42] Z. Han, J. Jiao, B. Zhang, Q. Ye, and J. Liu, "Visual object tracking via sample-based Adaptive Sparse Representation (AdaSR)," *Pattern Recognit.*, vol. 44, no. 9, pp. 2170–2183, Sep. 2011.
[43] M. Hofmann, D. Wolf, and G. Rigoll, "Hypergraphs for joint multi-view reconstruction and multi-object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2013, pp. 3650–3657.
[44] L. Oliveira, U. Nunes, P. Peixoto, M. Silva, and F. Moita, "Semantic fusion of laser and vision in pedestrian detection," *Pattern Recognit.*, vol. 43, no. 10, pp. 3648–3659, 2010.
[45] "Sdl Dataset." [Online]. Available: http://www.ucassdl.cn/resource.asp
[46] "LIPD Dataset in Urban Environment." [Online]. Available: http://www2.isr.uc.pt/cpremebida/dataset/
[47] C.-H. Kuo, C. Huang, and R. Nevatia, "Multi-target tracking by on-line learned discriminative appearance models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recog.*, 2010, pp. 685–692.



Shan Gao received the B.S. degree in communication engineering from Nankai University, Tianjin, China, in 2010 and the M.S. degree from University of Chinese Academy of Sciences, Beijing, China, in 2013. He is currently working toward the Ph.D. degree in computer science with the School of Electronic, Electrical, and Communication Engineering, University of Chinese Academy of Sciences. His research interests include object detection and tracking, image processing, and multisensor fusion.

Zhenjun Han received the B.S. degree in software engineering from Tianjin University, Tianjin, China, in 2006 and the M.S. and Ph.D. degrees from University of Chinese Academy of Sciences, Beijing, China, in 2009 and 2012, respectively. Since 2013, he has been an Associate Professor with the School of Electronic, Electrical, and Communication Engineering, University of Chinese Academy of Sciences. His research interests include object tracking and pedestrian reidentification.

Ce Li received the B.S. degree in software engineering from Tianjin University, Tianjin, China, in 2008 and the M.S. degree from University of Chinese Academy of Sciences, Beijing, China, in 2012. She is currently working toward the Ph.D. degree with the School of Electronic, Electrical, and Communication Engineering, University of Chinese Academy of Sciences. Her research interests include visual behavior analysis and intelligent surveillance.

Qixiang Ye (M'10) received the B.S. and M.S. degrees in mechanical and electrical engineering from Harbin Institute of Technology, Harbin, China, in 1999 and 2001, respectively, and the Ph.D. degree from Chinese Academy of Sciences, Beijing, China, in 2006.

From 2006 to 2009, he was an Assistant Professor with University of Chinese Academy of Sciences, where he has been an Associate Professor with the School of Electronic, Electrical, and Communication Engineering since 2009. Since December 2012, he has been a Visiting Assistant Professor with the Institute for Advanced Computer Studies, University of Maryland, College Park, MD, USA. He is the author or coauthor of more than 50 papers in refereed conferences and journals. He pioneered the kernel SVM-based pyrolysis output prediction software that was put into practical application by SINOPEC in 2012. He developed two kinds of piecewise linear SVM methods that were applied in image-based object detection. His research interests include image processing, image-based object detection, and machine learning.

Dr. Ye received the Sony Outstanding Paper Award in 2005.

Jianbin Jiao (M'10) received the B.S., M.S., and Ph.D. degrees in mechanical and electronic engineering from Harbin Institute of Technology (HIT), Harbin, China, in 1989, 1992, and 1995, respectively. From 1997 to 2005, he was an Associate Professor with HIT. Since 2006, he has been a Professor with the School of Electronic, Electrical, and Communication Engineering, University of the Chinese Academy of Sciences, Beijing, China. His research interests include image processing, pattern recognition, and intelligent surveillance.