A Hybrid Data Association Framework for Robust Online Multi-Object Tracking

Min Yang, Yuwei Wu, and Yunde Jia, *Member, IEEE*

Abstract—Global optimization algorithms have shown impressive performance in data-association-based multi-object tracking, but handling online data remains a difficult hurdle to overcome. In this paper, we present a hybrid data association framework with a min-cost multi-commodity network flow for robust online multi-object tracking. We build local target-specific models interleaved with global optimization of the optimal data association over multiple video frames. More specifically, in the min-cost multi-commodity network flow, the target-specific similarities are online learned to enforce the local consistency for reducing the complexity of the global data association. Meanwhile, the global data association taking multiple video frames into account alleviates irrecoverable errors caused by the local data association between adjacent frames. To ensure the efficiency of online tracking, we give an efficient near-optimal solution to the proposed min-cost multi-commodity flow problem, and provide the empirical proof of its sub-optimality. The comprehensive experiments on real data demonstrate the superior tracking performance of our approach in various challenging situations.

Index Terms—Multi-object tracking, data association, optimization, multi-commodity flow.

I. INTRODUCTION

ONLINE multi-object tracking estimates the spatio-temporal trajectories of multiple objects in an online video stream (*i.e.*, the video is provided frame-by-frame), which is a fundamental problem for numerous real-time applications, such as video surveillance, autonomous driving, and robot navigation. Assume that an object detector is available to detect potential locations of multiple objects in each frame, the tracking problem is consequently reduced to a data association procedure which links these individual detections to form consistent trajectories.

Data association is a challenging problem in many situations, especially in complex scenes, due to the presence of occlusions, inaccurate detections, and interactions among similar-looking objects. Standard approaches for data association are to recursively link detections frame by frame [1]–[8], resulting in a bi-partite matching between the existing trajectories and the newly obtained detections, as shown in Fig. 1(a). These approaches are temporally local and computationally efficient, making them suitable for the online setting. However, using only the local information for data association might lead to irrecoverable errors when an object is undetected or confused with clutters. To overcome this shortcoming, the global data association over entire video frames (or a batch of frames) has been devoted to inferring optimal trajectories [9]–[17], as shown in Fig. 1(b). Such a data association problem can be solved in an optimization framework with carefully designed cost functions. Unfortunately, global association methods can not be directly applied to online video streams. Overlapping temporal window is a common choice to handle online data [10], [12], [13], but the connection between consecutive batches remains an open problem.

In this paper, we propose a hybrid data association approach for online multi-object tracking, which characterizes the superiorities of both local and global data association methods. The core of our approach lies on the association between the existing trajectories and the detections from multiple video frames within a temporal window, as shown in Fig. 1(c). We exploit a min-cost multi-commodity flow which is with respect to a cost-flow network constructed by the detections from multiple frames. The proposed mini-cost multi-commodity network is able to formulate a hybrid data association strategy to handle online data with an efficient near-optimal solution.

In our framework, all possible associations among the detections are represented by edges in the network, where the corresponding edge costs account for the association likelihoods. Each existing trajectory is then supposed to be a specific commodity, and its optimal associations can be found by sending specific commodity flows through the network with a minimum cost. The following three issues need to be studied: (i) identifying newly appeared objects automatically; (ii) computing edge cost for different commodities; (iii) solving the min-cost multi-commodity flow problem efficiently. Our contributions are three-fold.

- We introduce a dummy commodity into our network to automatically identify a new object. The dummy commodity corresponds to a target-independent model, and its commodity flows indicate the permissible tracks of objects newly appeared in a temporal window.
- We present an online discriminative appearance modeling approach to build target-specific models for different existing trajectories. The edge costs of multiple commodities in the network are estimated by exploiting the target-specific information to discriminate a specific target from both other targets and the background.
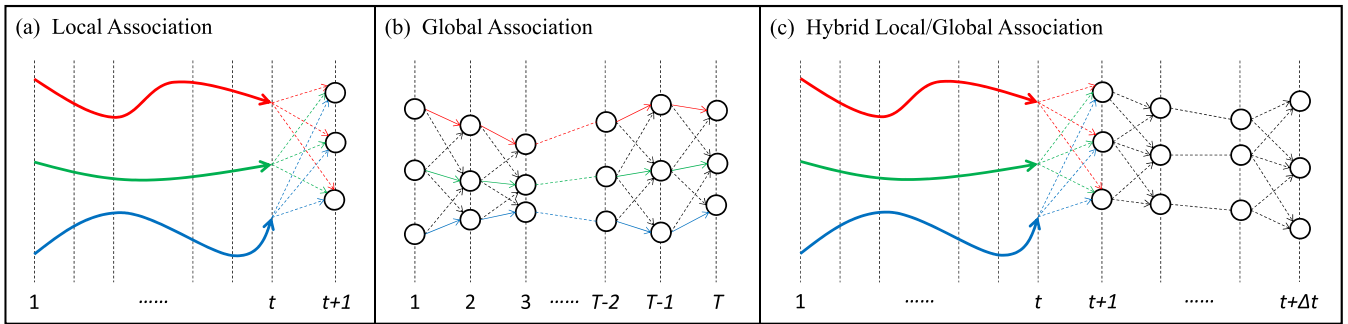
Fig. 1. Illustration of our hybrid data association approach. (a) Local association is performed between two consecutive frames t and $(t+1)$, and a bi-partite matching between the existing trajectories (marked as color arrows) at the current frame t and the detections (marked as circles) from the next frame $(t+1)$ is usually solved. (b) Global association is performed over a batch of frames (length T in this example), and an optimization problem is usually solved to infer optimal trajectories based on the pairwise affinity between detections. (c) The hybrid association finds globally optimal associations for the existing trajectories within a temporally local window (length Δt in this example), and the target-specific information from the existing trajectories provides helpful local constraints to guide the global optimization.

- We propose a near-optimal solution algorithm to the min-cost multi-commodity flow problem, and provide an empirical proof of its sub-optimality. By using the column generation strategy, our solution is extremely efficient and performs well in multi-object tracking.

The proposed hybrid strategy offers several advantages over existing methods. First, it makes the global optimization of trajectories applicable to online data. The local association between consecutive frames is extended to account for more hypotheses from multiple frames. Irrecoverable errors caused by noisy detections or frequent occlusions can be alleviated to improve online tracking performance. Second, the target-specific information from the existing trajectories is explicitly modeled to guide the global optimization over the current batch of frames. In practice, it enforces local constraints to reduce the complexity of the optimization problem as the associated detections are restricted to be consistent with the target-specific models. Both qualitative explanation and experimental confirmation are provided to support this claim.

The rest of the paper is organized as follows. Section II reviews the related work. Section III describes details of our online multi-object tracking method using the hybrid data association including min-cost multi-commodity flow formulation and its edge costs. Section IV presents the globally-optimal solution of our model. We report and discuss the experimental results in Section V, and conclude the paper in Section VI.

II. RELATED WORK

In multi-object tracking, data association based methods fall into a sub-domain known as the *tracking-by-detection* technique, which has shown impressive tracking performance in unconstrained environments. A thorough review can be found in [18]. As evidenced in Section I, the local association method has aroused considerable research interests. Especially with the success of recurrent neural networks (RNNs) in computer vision community [19], RNNs-based methods have witnessed significant advances on MOT problems. Based on the pioneer work introduced by Ondruska and Posner [20], RNNs-based method quickly sparked significant interest to model the local

association, and inspired a number of extensions including [21]–[23]. Nevertheless, the RNNs usually come with high computational and memory demands both during the model training and inference. We here enforce the locality constraint into the global data association formulation, and introduce a hybrid data association framework that is able to integrate the advantages of both local and global association methods.

Maintaining locality for global data association is critical for multi-object tracking performance, since global optimization might scale poorly for the complex scenario and long batches without local constraints. Many global association methods enforce locality by iteratively optimizing trajectories [24]–[27], or using tracklets (*i.e.*, short-term trajectory fragments) instead of individual detections [9], [15], [28]. However, these strategies are hardly applied to online video streams. Alternatively, one can divide an online video stream into consecutive batches with temporal sliding windows, and apply global data association to each video batch [10], [12], [13]. In order to produce consistent trajectories, the connection between optimized trajectories from adjacent batches need to be considered. However, most existing methods adopt heuristic strategies to connect adjacent batches and can not ensure the optimality of the trajectories.

To retain the ability of handling online data, we turn to explicitly model the target-specific information from previous observations, similar to local data association methods, to cooperate with the global data association over multiple frames. Integrating local and global data association is rarely mentioned in the literature. Lenz *et al.* [29] proposed an approximate online solution to the min-cost network flow problem with bounded memory and computation. The local consistency, however, is ignored in the optimization of trajectories. Choi [30] proposed an online multi-object tracking method to formulate the data association between previously tracked objects and detections in a temporal window. The method has a similar problem setting with our method, while the difference is that a highly non-convex formulation is adopted to select appropriate hypotheses for the objects. The solution heavily relies on both the affinity measures and the generated trajectory hypotheses. In contrast, we use a more
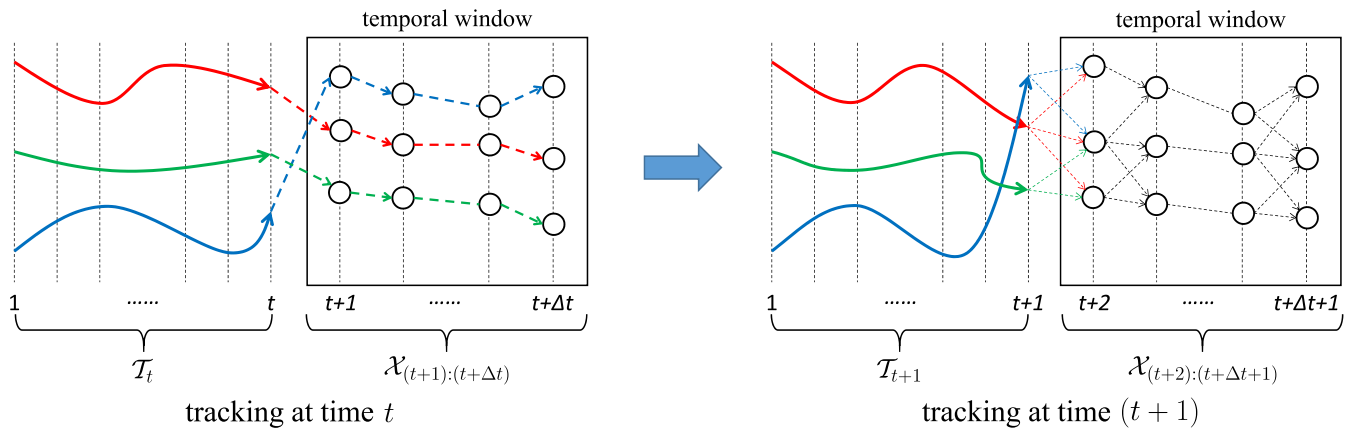
Fig. 2. Illustration of the multi-object tracking process with our hybrid data association. At each time step t, we solve a data association problem between the set of existing trajectories $\mathcal{T}_t$ and the set of detection responses $\mathcal{X}_{(t+1):(t+\Delta t)}$ in a temporal window $[t+1, t+\Delta t]$. After that, the trajectory set $\mathcal{T}_t$ is updated to $\mathcal{T}_{t+1}$ by incorporating the associated detections at the frame $(t+1)$, and the temporal window moves *one* time step forward.

compact formulation, *i.e.*, the min-cost multi-commodity flow, to address the hybrid data association. The target-specific information contained in the existing trajectories is incorporated into the flow costs in a natural way, ensuring that the objective is still convex. We also propose an optimization algorithm to the network flow problem, and show its effectiveness in multi-object tracking.

Recently, multi-commodity flow has been introduced into multi-object tracking in [31] and [32]. Ben Shitrit *et al.* [31] employed the multi-commodity network to account for different appearance groups which are fixed beforehand. Each appearance group (*e.g.*, a basketball team) is supposed to be a specific commodity in the network, and solving multi-commodity flow problems is able to distinguish different appearance groups during the optimization process. Dehghan *et al.* [32] have focused on integrating object detector learning and multi-object tracking, where the multi-commodity network is used to track a fixed number of objects in a short video batch. Our approach is different from these methods in that we use a multi-commodity network to formulate a hybrid data association strategy to handle online data. Furthermore, a high-quality near-optimal solution to the min-cost multi-commodity flow problem can be achieved by an efficient algorithm, especially when the number of objects (commodities) is relatively large. Thus we do not need to heuristically prune the graph [31] or iteratively relax the hard constraints [32].

III. HYBRID DATA ASSOCIATION

Let $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^N$ denote the set of detections from the video with $\mathbf{x}_i$ the i-th detection and N the number of detections. Assume that, at each time step t, we have a set of existing trajectories $\mathcal{T}_t$ and observe multiple video frames in a temporal window $[t+1, t+\Delta t]$. A set of detection responses $\mathcal{X}_{(t+1):(t+\Delta t)}$ is obtained by applying an object detector to each video frame within the temporal window. The task of *hybrid data association* is to find globally optimal associations of $\mathcal{T}_t$ over the detections $\mathcal{X}_{(t+1):(t+\Delta t)}$, and simultaneously identify newly appeared objects. Then the trajectory set $\mathcal{T}_t$ is updated to $\mathcal{T}_{t+1}$ by incorporating the associated detections at the

frame $(t+1)$, and the temporal window moves *one* time step forward, as shown in Fig. 2. In practice, it causes a latency of $(\Delta t - 1)$ to output tracking results, as the trajectories at the frame $(t+1)$ are not updated until the frame $(t+\Delta t)$ is observed. Nevertheless, our approach operates in an online manner to handle online data. Note that the traditional local or global data association methods can be regarded as special cases of the proposed hybrid framework by adjusting the length of the temporal window as $\Delta t = 1$ or $\Delta t = T$ (total length of the video), respectively.

In this section, the data association between $\mathcal{T}_t$ and $\mathcal{X}_{(t+1):(t+\Delta t)}$ is formulated as a *min-cost multi-commodity flow* problem, as shown in Fig. 3. For the convenience of discussion, we drop the time index in the following description, and denote the current set of existing trajectories as $\mathcal{T} = \{T_k\}_{k=1}^K$, where T_k is the k-th existing trajectory and K is the number of existing trajectories.

A. Our Min-Cost Multi-Commodity Flow

Given the set of existing trajectories $\mathcal{T}$ and the set of detections $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^N$, we introduce a directed network $G(\mathcal{X})$ with multiple sources s_k and sinks n_k, $k \in \{0, 1, \dots, K\}$. The directed network $G(\mathcal{X})$ is constructed by the set of detections $\mathcal{X}$. Each detection $\mathbf{x}_i \in \mathcal{X}$ corresponds to a pair of nodes (u_i, v_i) in G connected by an *observation edge* with cost c_i and flow f_i. The cost c_i indicates the confidence of observing the detection $\mathbf{x}_i$, and the flow f_i encodes the selection of the detection $\mathbf{x}_i$ in some tracks. Each transition between a pair of detections $(\mathbf{x}_i, \mathbf{x}_j)$ is represented by a *transition edge* (v_i, u_j) with cost c_{ij} and flow f_{ij}. The cost c_{ij} represents the coherence between detections $\mathbf{x}_i$ and $\mathbf{x}_j$, and the flow f_{ij} indicates that the two detections are connected through the same track. The set of permissible transitions between detections is denoted as E. It could be a subset of all pairs of detections in successive frames by using choice heuristics (*e.g.*, spatial proximity). Finally, the source s and sink n are introduced with *track start edges* (s, u_i) (with cost c_{si} and flow f_{si}) and *track termination edges* (v_i, n) (with cost c_{in} and flow f_{in}). Then the multi-object tracking problem

Observation edges   **Transition edges**   **Track start/termination edges**
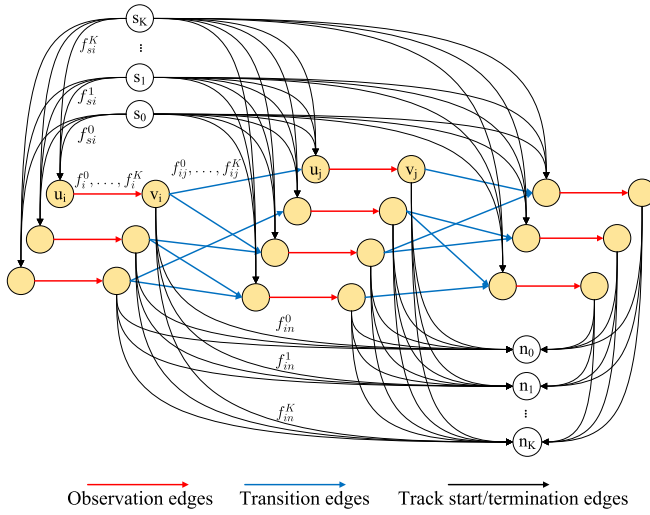
Fig. 3. An example of the directed network with multiple sources and sinks. Each detection $\mathbf{x}_i \in \mathcal{X}$ is represented by a pair of nodes connected by an observation edge. Possible transitions between detections are modeled by transition edges. To allow tracks to start and terminate at any detections from the video, each detection is connected to both a source s and a sink n. We use f_i^k, f_{ij}^k, f_{si}^k, and f_{in}^k to represent the amount of the k-th commodity flows on the observation edge (u_i, v_i), the transition edge (v_i, u_j), the track start edges (s_k, u_i), and the track terminate edge (v_i, n_k), respectively. We add a dummy commodity 0 with the source s_0 and sink n_0 to represent a target-independent model.

is formulated as sending a set of flows from the source s to sink n, which minimizes the total cost

$$C(f) = \sum_i c_i f_i + \sum_i c_{si} f_{si} + \sum_{ij \in E} c_{ij} f_{ij} + \sum_i c_{in} f_{in}. \quad (1)$$

In this work, each existing trajectory T_k is supposed to be a target-specific commodity k which corresponds to a source-sink pair (s_k, n_k). Specifically, sources s_k and sinks n_k are introduced with track start edges (s_k, u_i) and track termination edges (v_i, n_k) connected to all detections, indicating that the existing trajectories or newly appeared trajectories are allowed to start and terminate at any detection from the temporal window. For each commodity k, sending flows from s_k to n_k through the network incurs a specific set of edge costs. Formally, we use f_i^k, f_{ij}^k, f_{si}^k, and f_{in}^k to represent the amount of the k-th commodity flows on the observation edge (u_i, v_i), the transition edge (v_i, u_j), the track start edges (s_k, u_i), and the track terminate edge (v_i, n_k), respectively. The corresponding edge costs, in a similar way, are denoted as c_i^k, c_{ij}^k, c_{si}^k, and c_{in}^k.

To identify newly appeared objects, we add a dummy commodity 0 with the source s_0 and sink n_0 to represent a target-independent model. We call a flow sent from s_k to n_k as the k-th commodity flow. That is, the source and sink are extended to account for multiple commodities (see an example in Fig. 3). Then the optimal associations of T_k over $\mathcal{X}$ can be found by sending the k-th commodity flow through the network. It leads to a multi-commodity flow problem in the community of network flow [33].

With the network $G(\mathcal{X})$, the hybrid data association problem is formulated as finding an optimal set of flows between

multiple source and sink pairs $\{(s_k, n_k)\}_{k=0}^K$, which **minimizes** the total cost

$$\sum_{k=0}^K \left(\sum_i c_{si}^k f_{si}^k + \sum_i c_i^k f_i^k + \sum_{ij \in E} c_{ij}^k f_{ij}^k + \sum_i c_{in}^k f_{in}^k \right). \quad (2)$$

Intuitively, each flow path connects a set of coherent detections over time and thus can be interpreted as an object track. In practice, the flow should subject to the following constraints to satisfy the physical conditions in a real world:

$$\forall k, \quad f_i^k, f_{ij}^k, f_{si}^k, f_{in}^k \in \{0, 1\}, \quad (3)$$

$$\forall k, \quad f_{si}^k + \sum_{j:ji \in E} f_{ji}^k = f_i^k = \sum_{j:ij \in E} f_{ij}^k + f_{in}^k, \quad (4)$$

$$\forall e \in \{i, ij, si, in\}, \quad \sum_{k=0}^K f_e^k \leq 1, \quad (5)$$

$$\forall k, \quad \sum_i f_{si}^k = d_k = \sum_i f_{in}^k. \quad (6)$$

The constraint (3) is an *edge capacity constraint* which means that each detection belongs to at most one track. The *flow conservation constraint* (4) encodes that the sum of flows arriving at any detection $\mathbf{x}_i^k$ is equal to the flow of its observation edge f_i^k, which also is the sum of outgoing flows from the detection $\mathbf{x}_i^k$. The constraints (3), (4), and (5) ensure that all permissible flows in the network come in the form of flow paths from sources to sinks, and also ensure that there is no overlap between multiple paths. The flow variables f_i^k, f_{ij}^k, f_{si}^k, f_{in}^k act as binary indicators taking the value 1 when the corresponding edge is selected in a flow path of the commodity k. The constraint (6) restricts the total amount of flows sent from s_k to n_k to be a certain value d_k. Consequently, each flow path in the network can be interpreted as an object track which connects a set of coherent detections over time. A flow path of commodity k with $k \neq 0$ is the success track of the existing trajectory T_k within the temporal window. We thus set $d_k = 1$ for $k \neq 0$ to ensure that each existing trajectory has only one success track. For the dummy commodity, we set $d_0 = 20$ to capture a sufficient number of new objects.

To simplify the notation, we collect the flow variables f_i^k, f_{ij}^k, f_{si}^k, f_{in}^k in a long vector $\mathbf{f}^k$ and the edge cost variables c_i^k, c_{ij}^k, c_{si}^k, and c_{in}^k in a long vector $\mathbf{c}^k$, respectively. Then the optimization problem that minimizes the cost (2) with constraints (3), (4), (5), and (6) can be rewritten as

$$\min_{\mathbf{f}} \sum_{k=0}^K \left(\mathbf{c}^k \right)^\top \mathbf{f}^k$$

$$s.t. \ \forall k, \ \mathbf{f}^k \geq \mathbf{0},$$

$$\forall k, \ U\mathbf{f}^k = \mathbf{0},$$

$$\forall k, \ V\mathbf{f}^k = d_k \mathbf{1},$$

$$\forall k, \ \mathbf{f}^k \ \text{integer},$$

$$\sum_{k=0}^K \mathbf{f}^k \leq \mathbf{1}, \quad (7)$$

where the constraints are rearranged into the matrix form. The vectors with all zero and one entries are denoted as $\mathbf{0}$ and $\mathbf{1}$, respectively.

B. Computing Edge Costs

In our min-cost multi-commodity flow formulation, sending flows of a commodity k through the network incurs a specific set of edge costs $\mathbf{c}^k$. Therefore, local information contained in the existing trajectories can be incorporated into the edge costs in a natural way, and thus guides the global data association over multiple video frames. In this subsection, we show that the edge costs can be computed by exploiting the target-specific information from the existing trajectories.

1) Observation Cost: Given an existing trajectory T_k and a detection $\mathbf{x}_i$, the observation cost c_i^k encodes the possibility of $\mathbf{x}_i$ belonging to T_k. c_i^k is computed by

$$c_i^k = -\phi_k(\widetilde{\mathbf{a}}_k, \mathbf{a}_i), \tag{8}$$

where $\phi_k(\cdot, \cdot)$ is the similarity function used to recognize the specific object corresponding to T_k, and $\widetilde{\mathbf{a}}_k$ and $\mathbf{a}_i$ are the appearance feature of the existing trajectory T_k and the detection $\mathbf{x}_i$, respectively. We use Convolutional Neural Network (CNN) features to capture the appearance information of an object, as described in Section V-C. The appearance feature of T_k is represented by the average feature vector over the last 10 frames, and the appearance feature of $\mathbf{x}_i$ is extracted from the image region corresponding to its location. The similarity function $\phi_k(\cdot, \cdot)$ assigns high similarity scores to pairs of appearance features when both of them originate from the same object corresponding to T_k, while producing low similarity scores when more than one of them originate from other objects. We utilize an online similarity learning approach to learn the target-specific similarity function $\phi_k(\cdot, \cdot)$, as described in Section III-C. For the dummy commodity, we set c_i^0 to the negative detector score of the detection $\mathbf{x}_i$.

Note that the observation costs take negative values when the appearance similarity scores or the detector scores are larger than zero, which facilitates the generation of long trajectories. Furthermore, the observation costs taking negative values ensure the appearance consistency for each trajectory since the total cost of the network flows is minimized in our model.

2) Transition Cost: The transition cost c_{ij}^k indicates the confidence of connecting the detections $\mathbf{x}_i$ and $\mathbf{x}_j$ in the same success track of T_k, which can be computed by

$$c_{ij}^k = -\phi_k(\mathbf{a}_i, \mathbf{a}_j), \tag{9}$$

where $\mathbf{a}_i$ and $\mathbf{a}_j$ are the appearance feature of the detection $\mathbf{x}_i$ and the detection $\mathbf{x}_j$, respectively. For the dummy commodity, the transition cost c_{ij}^0 is computed by using the cosine of the angle between two feature vectors as a target-independent similarity function.

3) Track Start/Termination Cost: The track start cost c_{si}^k encodes the possibility that a success track T_k starts at the detection $\mathbf{x}_i$. Given the frame index t_i of the detection $\mathbf{x}_i$, we use a constant velocity model to obtain a prediction of T_k at the frame t_i, denoted as $p(T_k, t_i)$. Then the track start cost c_{si}^k is given by

$$c_{si}^k = -\eta^{t_i - \psi(T_k)} \cdot o\left(p(T_k, t_i), \mathbf{x}_i\right), \tag{10}$$

where η is a decay factor (set to 0.95) which discounts long term prediction, $\psi(T_k)$ is the last associated frame of T_k,

and the function o denotes the overlap rate between two bounding boxes. For the dummy commodity, we set the track start cost c_{si}^0 to be a large positive value (10 in our implementation) to reduce the priority of identifying new objects while facilitating the association of the existing trajectories.

Similarly, the track termination cost c_{in}^k encodes the possibility that a success track of the T_k ends at the detection $\mathbf{x}_i$. Suppose that an object trajectory ends at all detections with the same probability, we simply set $c_{in}^k = 10$ for all k.

C. Online Similarity Learning

Given an existing trajectory T_k, we learn a target-specific similarity function $\phi_k(\cdot, \cdot)$ to distinguish the corresponding object from the others. Formally, we use a parametric similarity function that has a bi-linear form to estimate the appearance similarity between two appearance features $\mathbf{x}_i$ and $\mathbf{x}_j$,

$$\phi_k(\mathbf{a}_i, \mathbf{a}_j) = \mathbf{a}_i^\top \mathbf{W}_k \mathbf{a}_j, \tag{11}$$

where $\mathbf{W}_k \in \mathbb{R}^{m \times m}$ with m the dimensionality of appearance features. The task of online similarity learning is to estimate an appropriate parameter matrix $\mathbf{W}_k$ for the existing trajectory T_k in the process of online tracking.

At each time t, we assume that a detection from time $(t+1)$, whose appearance feature is denoted as $\mathbf{a}_k^{(t+1)}$, is associated with the existing trajectory $T_k^{(t)}$. The parameter matrix $\mathbf{W}_k^{(t)}$ of $T_k^{(t)}$ at the current time t is needed to be updated to account for the newly observed appearance feature $\mathbf{a}_k^{(t+1)}$. The principle of updating $\mathbf{W}_k^{(t)}$ is to recognize $\mathbf{a}_k^{(t+1)}$ as a relevant appearance and $\{\mathbf{a}_l^{(t+1)} | l \neq k\}$ as irrelevant appearances. We therefore construct a set of triplets $\mathcal{S}_k^{(t+1)} = \{(\widetilde{\mathbf{a}}_k^{(t)}, \mathbf{a}_k^{(t+1)}, \mathbf{a}_l^{(t+1)}) | l \neq k\}$, where $\widetilde{\mathbf{a}}_k^{(t)}$ is the appearance feature of $T_k^{(t)}$ at the current time t. Each triplet (a, b, c) indicates that the similarity between a and b is apparently larger than the similarity between a and c. Forcing the current matrix $\mathbf{W}_k^{(t)}$ to satisfy the triplet set $\mathcal{S}_k^{(t+1)}$ leads to the updated matrix $\mathbf{W}_k^{(t+1)}$ at time $(t+1)$.

We here present an incremental update algorithm to satisfy the triplets sequentially [34]. Without loss of generality, assume that we have a parameter matrix $\mathbf{W}^\tau$ at the τ-th iteration and observe a triplet $(\mathbf{a}_\tau, \mathbf{a}_\tau^+, \mathbf{a}_\tau^-)$. The goal of incremental updating is to obtain a new matrix $\mathbf{W}$ satisfying

$$(\mathbf{a}_\tau)^\top \mathbf{W}(\mathbf{a}_\tau^+) > (\mathbf{a}_\tau)^\top \mathbf{W}(\mathbf{a}_\tau^-) + 1, \tag{12}$$

which means that it fulfills the definition of a triplet with a safety margin of 1. Meanwhile, applying the Passive-Aggressive algorithm [35] to maintain smoothness, the new matrix is selected to remain close to the previous matrix $\mathbf{W}^\tau$.

We define a hinge loss function to measure the confidence that a matrix $\mathbf{W}$ satisfies the triplet $(\mathbf{a}_\tau, \mathbf{a}_\tau^+, \mathbf{a}_\tau^-)$,

$$L_\mathbf{W}(\mathbf{a}_\tau, \mathbf{a}_\tau^+, \mathbf{a}_\tau^-) = \max\left\{0, 1 - (\mathbf{a}_\tau)^\top \mathbf{W}(\mathbf{a}_\tau^+) + (\mathbf{a}_\tau)^\top \mathbf{W}(\mathbf{a}_\tau^-)\right\}. \tag{13}$$

Then the problem of incremental updating can be expressed as

$$\mathbf{W}^{\tau+1} = \arg\min_{\mathbf{W}} \frac{1}{2}\|\mathbf{W} - \mathbf{W}^{\tau}\|_F^2 + C\xi$$
$$s.t.\ L_{\mathbf{W}}(\mathbf{a}_{\tau}, \mathbf{a}_{\tau}^+,\ \mathbf{a}_{\tau}^-) \leq \xi,\ \xi \geq 0, \qquad (14)$$

where $\|\cdot\|_F$ is the Frobenius norm, ξ is a slack variable, and C is a parameter that controls the trade-off between preserving smoothness and minimizing the loss on the current triplet.

Since Eq. (14) is a constrained convex optimization problem, we can directly derive its optimal solution by using the Karush-Kuhn-Tucker (KKT) conditions,

$$\begin{cases} \mathbf{W}^{\tau+1} = \mathbf{W}^{\tau} + \alpha_{\tau}\mathbf{V}^{\tau}, \\ \mathbf{V}^{\tau} = \mathbf{a}_{\tau}(\mathbf{a}_{\tau}^+ - \mathbf{a}_{\tau}^-)^{\top}, \\ \alpha_{\tau} = \min\left\{C, \dfrac{L_{\mathbf{W}^{\tau}}(\mathbf{a}_{\tau}, \mathbf{a}_{\tau}^+, \mathbf{a}_{\tau}^-)}{\|\mathbf{V}^{\tau}\|^2}\right\}. \end{cases} \qquad (15)$$

According to Eq. (15), the update only happens when the hinge loss $L_{\mathbf{W}^{\tau}}(\mathbf{a}_{\tau}, \mathbf{a}_{\tau}^+, \mathbf{a}_{\tau}^-)$ on the triplet is larger than zero.

To summarize, for each existing trajectory $T_k^{(t)}$ at time t, we incrementally update the similarity function parameterized by the matrix $\mathbf{W}_k^{(t)}$ through the following steps:

- construct the triplet set $\mathcal{S}_k^{(t+1)}$;
- sequentially update the matrix by using the triplet in $\mathcal{S}_k^{(t+1)}$ one-by-one with Eq. (15);
- obtain the updated matrix $\mathbf{W}_k^{(t+1)}$ at the time $(t+1)$.

Note that the parameter matrix $\mathbf{W}_k$ of the existing trajectory T_k is initialized to an identity matrix when the trajectory is initialized. The incremental update on each iteration, as defined by Eq. (15), only involves few matrix operations and thus is extremely efficient. Moreover, the entire online similarity learning process for each trajectory is independent and can be performed parallelly to further improve the computational efficiency.

IV. OPTIMIZATION

Finding a global minimum of the formulation (7) is exactly an Integer Linear Program (ILP) which is NP-hard. In addition, the optimal solution to its Linear Program (LP) relaxation is not guaranteed to be integral, which serves as an important requirement for the generation of reasonable object trajectories. In this section, by exploring the special structure of the constraints, we propose an efficient optimization algorithm that is able to provide near-optimal integer solutions with empirical sub-optimality certificates.

A. Dantzig-Wolfe Decomposition

Most constraints in the problem (7) only involve a single commodity, we use the Dantzig-Wolfe decomposition [36] to reformulate the "relatively easy" constraints. Specifically, we consider the nonnegativity constraints $\mathbf{f}^k \geq \mathbf{0}$ and the flow conservation constraints $U\mathbf{f}^k = \mathbf{0}$ that are exactly identical for each commodity k. All feasible flow vectors can be treated as points lying on the polyhedron $P = \{\mathbf{f} \geq \mathbf{0} \mid U\mathbf{f} = \mathbf{0}\}$. It is a *cone* and has a single vertex $\mathbf{0}$ and a finite number of rays

$\{\mathbf{r}^1, \ldots, \mathbf{r}^G\}$. By the Minkowski-Weyl theorem [37], we can represent a flow vector $\mathbf{f}^k \in P$ as

$$\mathbf{f}^k = \sum_{g=1}^{G} \lambda_{k,g}\mathbf{r}^g, \qquad (16)$$

where $\lambda_{k,g} \geq 0$ is the associated non-negative coefficient. In our case, the rays $\{\mathbf{r}^1, \ldots, \mathbf{r}^G\}$ form the basis of the null space defined by the constraint matrix U in the flow conservation constraints $U\mathbf{f} = \mathbf{0}$, which correspond to indicator vectors of all possible paths from the source to the sink in our network.

Substituting the equation (16) into (7), (7) can be rewritten as

$$\min_{\boldsymbol{\lambda}} \sum_{k=0}^{K}\sum_{g=1}^{G} \lambda_{k,g}\left(\left(\mathbf{c}^k\right)^{\top}\mathbf{r}^g\right)$$
$$s.t. \sum_{k=0}^{K}\sum_{g=1}^{G} \lambda_{k,g}\mathbf{r}^g \leq \mathbf{1},$$
$$\forall k,\ \sum_{g=1}^{G} \lambda_{k,g} = d_k,$$
$$\forall k,\ \forall g,\ \lambda_{k,g} \geq 0,$$
$$\forall k,\ \forall g,\ \lambda_{k,g}\ \text{integer}. \qquad (17)$$

The formulation (17) can be seen as a path flow problem that is equivalent to the original edge flow formulation (7). The variable $\lambda_{k,g}$ is interpreted as the k-th commodity flow on the path corresponding to $\mathbf{r}^g$, indicating whether the path $\mathbf{r}^g$ is selected by the k-th commodity or not.

B. Column Generation

Enumerating all possible paths to construct the complete set $\{\mathbf{r}^1, \ldots, \mathbf{r}^G\}$ leads to a very large number of variables for optimization. Actually, only a few paths among $\{\mathbf{r}^1, \ldots, \mathbf{r}^G\}$ are needed to achieve the optimal solution in practice. We thus use the column generation algorithm [38] to dynamically find the critical paths. In the following, we consider the LP relaxation of the formulation (17), denoted as the master LP (MLP), by removing the integer constraints, and show later how to obtain a near-optimal integer solution.

Formally, the MLP problem can be expressed as

$$\text{(MLP)}\ \min_{\boldsymbol{\lambda}} \sum_{k=0}^{K}\sum_{g\in\mathcal{I}} \lambda_{k,g}\left(\left(\mathbf{c}^k\right)^{\top}\mathbf{r}^g\right)$$
$$s.t. \sum_{k=0}^{K}\sum_{g\in\mathcal{I}} \lambda_{k,g}\mathbf{r}^g \leq \mathbf{1},$$
$$\forall k,\ \sum_{g\in\mathcal{I}} \lambda_{k,g} = d_k,$$
$$\forall k,\ \forall g\in\mathcal{I},\ \lambda_{k,g} \geq 0, \qquad (18)$$

where $\mathcal{I} = \{1, \ldots, G\}$ is the whole index set of all possible paths. The dual problem of the MLP, denoted as DMLP,

has the form

$$\text{(DMLP)} \quad \max_{\boldsymbol{\pi}, \boldsymbol{\sigma}} \quad -\mathbf{1}^\top \boldsymbol{\pi} + \sum_{k=0}^{K} d_k \sigma_k$$

$$s.t. \ \forall k, \ \forall g \in I, \ -\boldsymbol{\pi}^\top \mathbf{r}^g + \sigma_k \leq \left(\mathbf{c}^k\right)^\top \mathbf{r}^g,$$

$$\boldsymbol{\pi} \geq \mathbf{0}, \tag{19}$$

where $(\boldsymbol{\pi}, \sigma_k)$ are the dual variables of the primal variables $\lambda_{k,g}$. According to the duality theory, any dual feasible solution of the DMLP provides a lower bound on the MLP, being the fundamental of the column generation algorithm.

Assume that, at the iteration τ, only a subset of paths $\{\mathbf{r}^g\}_{g \in \mathcal{I}_\tau}$ with $\mathcal{I}_\tau \subset \mathcal{I}$ is available. Solving the MLP on the subset $\mathcal{I}_\tau$ gives rise to the restricted master linear program (RMLP),

$$\text{(RMLP)} \quad \min_{\boldsymbol{\lambda}} \quad \sum_{k=0}^{K} \sum_{g \in \mathcal{I}_\tau} \lambda_{k,g} \left(\left(\mathbf{c}^k\right)^\top \mathbf{r}^g \right)$$

$$s.t. \quad \sum_{k=0}^{K} \sum_{g \in \mathcal{I}_\tau} \lambda_{k,g} \mathbf{r}^g \leq \mathbf{1},$$

$$\forall k, \quad \sum_{g \in \mathcal{I}_\tau} \lambda_{k,g} = d_k,$$

$$\forall k, \ \forall g \in \mathcal{I}_\tau, \ \lambda_{k,g} \geq 0. \tag{20}$$

Let $\lambda_{k,g}^*$ and $(\boldsymbol{\pi}^*, \sigma_k^*)$ be the optimal primal and dual solution to the RMLP, respectively. We need to check whether the optimal solution to the RMLP is also optimal for the MLP, and decide whether the current path set $\mathcal{I}_\tau$ is needed to be augmented. It can be realized by solving the following *pricing* problem:

$$\zeta_k = \min \left\{ \left(\mathbf{c}^k + \boldsymbol{\pi}^*\right)^\top \mathbf{r}^g \mid g \in \mathcal{I} \right\}. \tag{21}$$

In our case, the pricing problem turns into a *shortest path* problem with regard to the modified edge cost $(\mathbf{c}^k + \boldsymbol{\pi}^*)$ for the commodity k, which can be solved very efficiently by dynamic programming. With the optimal solution ζ_k to the pricing problem, we have the following proposition.

Proposition 1: If $\zeta_k \geq \sigma_k^*$ holds for all k, the optimal primal solution to the RMLP $\lambda_{k,g}^*$ optimally solves the MLP.

Proof: Given the optimal primal solution to the RMLP $\lambda_{k,g}^*$, we can validate that $\lambda_{k,g}^*$ is a feasible solution to the MLP by setting $\lambda_{k,g} = 0$ for those paths not in the current set $\mathcal{I}_\tau$. Therefore, the optimal value of the RMLP gives an upper bound on the MLP,

$$\upsilon(RMLP) \geq \upsilon(MLP), \tag{22}$$

where $\upsilon(RMLP)$ and $\upsilon(MLP)$ are the optimal values of the RMLP and the MLP, respectively.

According to the definition of the pricing problem (21), when $\zeta_k \geq \sigma_k^*$ holds for all $k \in \{0, 1, \ldots, K\}$, we have

$$\forall k, \quad \zeta_k = \min \left\{ \left(\mathbf{c}^k + \boldsymbol{\pi}^*\right)^\top \mathbf{r}^g \mid g \in \mathcal{I} \right\} \geq \sigma_k^*. \tag{23}$$

(23) can be rewritten as

$$\forall k, \ \forall g \in \mathcal{I}, \ -\boldsymbol{\pi}^{*\top} \mathbf{r}^g + \sigma_k^* \leq \left(\mathbf{c}^k\right)^\top \mathbf{r}^g, \tag{24}$$

which implies that the optimal dual solution to the RMLP $(\boldsymbol{\pi}^*, \sigma_k^*)$ is also a feasible solution to the DMLP given by (19). According to the duality theory, the solution $(\boldsymbol{\pi}^*, \sigma_k^*)$ provides a lower (dual) bound on the MLP, we therefore have

$$\upsilon(RMLP) \leq \upsilon(MLP). \tag{25}$$

Note that the above equation uses the fact that the optimal primal solution $\lambda_{k,g}^*$ and the optimal dual solution $(\boldsymbol{\pi}^*, \sigma_k^*)$ to the RMLP gives the exactly same optimal value of the objective function.

With the equations (22) and (25), we can conclude that the RMLP and the MLP have the same optimal value if $\zeta_k \geq \sigma_k^*$ holds for all k. The optimal primal solution to the RMLP $\lambda_{k,g}^*$ optimally solves the MLP, thereby completing the proof. ∎

If the condition of the Proposition 1 is not satisfied, *i.e.*, $\zeta_k < \sigma_k^*$ for some k, the shortest path $\tilde{\mathbf{r}}_k$ provided by the pricing problem (21) has a *negative reduced cost*. We introduce $\tilde{\mathbf{r}}_k$ into the subset $\mathcal{I}_\tau$, and repeat the process for the next iteration to decrease the objective value of the MLP.

To obtain a near-optimal integer solution to the ILP (17), one can retain the feasible solution with the minimum objective value once the RMLP provides an integer solution during the column generation process. Since the optimal solution to the MLP gives a lower bound for the ILP, the difference between the objective value of the returned integer solution and the lower bound is thus an upper bound certificate on its sub-optimality. In our experiments, we obtained small sub-optimality certificates for the returned integer solutions, indicating that our optimization algorithm based on column generation is stable. We summarize our method in Algorithm 1.

V. Experiments

In this section, we evaluate our approach on real world videos to demonstrate its effectiveness. Specifically, the performance of our approach is analyzed in three aspects. (i) We evaluate the influence of the length of the temporal window, *i.e.*, Δt on multi-object tracking performance for our hybrid data association framework; (ii) We compare the column generation (CG) solver introduced in this paper and the exact integer linear programming (ILP) solver in terms of sub-optimality, convergence speed, and MOTA score; (iii) We show that our approach produces superior tracking results over the state-of-the-art via both quantitative and qualitative evaluation.

A. Datasets

We use two publicly available benchmark datasets, *i.e.*, the *PETS 2009* dataset and the *MOTChallenge 2015* dataset, for performance evaluation. The details are listed as follows.

1) PETS 2009: The *PETS 2009* dataset [39] shows an outdoor scene where numerous pedestrians enter, exit, and interact with each other frequently. The images of the dataset are recorded in 768×576 pixels at 7 fps. The major challenges of this dataset are frequent occlusions either caused by people interaction or static occlusions due to a traffic sign. Apart from the widely used *S2L1* and *S2L2* sequence, we also evaluate our approach on the more challenging *S2L3* sequence that captures

Algorithm 1 The Hybrid Data Association via Column Generation

---

Input: the edge costs $\mathbf{c}^k$ and track numbers d_k for all commodities $k = 0, 1, \ldots, K$.

Output: the near-optimal integer solution $\{\mathbf{f}^k\}_{k=0}^K$ to the problem (7) and its sub-optimality certificate ϵ.

1: **Initialize:** the initial path set $\mathcal{I}_1$ consists of the shortest paths of all commodities with regard to the edge cost $\mathbf{c}^k$

2: **for** $\tau = 1$ to ITERMAX **do**

3:     solve the RMLP defined by (20) on $\mathcal{I}_\tau$ to get the optimal primal and dual solution $\lambda_{k,g}^*$, $(\boldsymbol{\pi}^*, \sigma_k^*)$;

4:     /* retain the integer solution */

5:     **if** $\lambda_{k,g}^*$ are integer **then**

6:         $v(ILP) = v(RMLP)$;

7:         $\tilde{\lambda}_{k,g} = \lambda_{k,g}^*$;

8:     **end if**

9:     /* find shortest paths */

10:     **for** $k = 0, \ldots, K$ **do**

11:         $\tilde{\mathbf{r}}_k = \arg\min_{\{\mathbf{r}^g | g \in \mathcal{I}\}} \left(\mathbf{c}^k + \boldsymbol{\pi}^*\right)^\top \mathbf{r}^g$;

12:         $\zeta_k = (\mathbf{c}^k + \boldsymbol{\pi}^*)^\top \tilde{\mathbf{r}}_k$;

13:     **end for**

14:     /* optimality check */

15:     **if** $\zeta_k \geq \sigma_k^*$ holds for all k **then**

16:         break;

17:     **end if**

18:     /* augment the path set */

19:     $\mathcal{I}_{\tau+1} = \mathcal{I}_\tau \cup \{\tilde{\mathbf{r}}_k | \forall k, \zeta_k < \sigma_k^*\}$.

20: **end for**

21: **return** $\mathbf{f}^k = \sum_{g \in I_\tau} \tilde{\lambda}_{k,g} \mathbf{r}^g$, $\epsilon = v(ILP) - v(RMLP)$.

---

much denser crowds. The input detections and ground truth of these sequences are from Milan *et al.* [26].

In our experiments, we use the *PETS 2009* dataset for diagnosis analysis, including the investigation of the influence of the critical parameter Δt (see Section V-D) and the comparison between the proposed CG solver and the ILP solver (see Section V-E). The reason is that the *S2L1*, *S2L2*, and *S2L3* sequences from the *PETS 2009* dataset, respectively, correspond to three representative application scenarios of multi-object tracking with low, high, and crowded object densities.

2) MOTChallenge 2015: The *MOTChallenge 2015* dataset gathers various existing and new challenging video sequences to evaluate the performance of multi-object tracking methods. Since our method performs tracking on the image coordinate, we use the *2D MOT 2015* sequences in the *MOTChallenge 2015*. The sequences are composed of 11 training and 11 testing video sequences in which the challenges include camera motion, low viewpoint, varying frame rates, and severe weather condition. The training sequences contain over 5500 frames ($\sim$ 7 minutes) and 500 annotated trajectories (39905 bounding boxes). The benchmark releases the ground truth of the training sequences publicly, and thus one can use the training sequences to determine the set of system parameters. The testing sequences contains over 5700 frames ($\sim$ 10 minutes) and 721 annotated trajectories

(61440 bounding boxes), while the annotations are not available to avoid (over)fitting of the competing methods to the specific sequences.

Since it is hard for methods to finetune on such a large amount of data, we use the 11 testing sequences from the *MOTChallenge 2015* dataset for quantitative comparison against various state-of-the-art trackers in our experiments (see Section V-F). Moreover, the tracking results of all competing methods are automatically evaluated by the benchmark and the performance scores are publicly online, making the quantitative comparison strictly fair.

B. Evaluation Metrics

We use the widely accepted CLEAR MOT performance metrics [40] for performance evaluation which include the multiple object tracking precision (MOTP↑) that measures average overlap rate between estimated trajectories and the ground truth, the multiple object tracking accuracy (MOTA↑) that is a cumulative accuracy combining false positives (FP↓), false negatives (FN↓) and identity switches (IDS↓). We also report performance scores defined by Li *et al.* [41], including the percentage of mostly tracked (MT↑) ground truth trajectories, the percentage of mostly lost (ML↓) ground truth trajectories, and the number of times that a ground truth trajectory is interrupted (Frag↓). To be specific, a ground truth trajectory is determined to be mostly tracked if and only if it is covered by the estimated trajectories with percentage larger than 80%, while a ground truth trajectory is determined to be mostly lost when the coverage percentage is less than 20%. Additionally, we report the false positive ratio to account for the accuracy of identifying true targets, which is measured by the number of false alarms per frame (FAF↓). Here, ↑ means that higher scores indicate better results, and ↓ represents that lower is better.

C. Appearance Feature

As for the appearance features, we utilize the region-based CNN features proposed in [42], where the deep neural network is trained on the ImageNet dataset and fine-tuned on the PASCAL VOC dataset. To obtain a more generic deep representation, we follow the strategy in [43] to use sum pooling to aggregate the output of the last convolutional layer, rather than directly use the features from the last fully-connected layer. For each detection region, the final feature vector is 256-dimensional with better time and space complexity. Considering that objects of interest tend to be located close to the geometrical center of an image, we also apply the centering prior to the sum pooling strategy to improve the accuracy, which assigns larger weights to the features from the center of the region.

D. Influence of Large Temporal Window

The length of the temporal window (Δt) determines the number of video frames in which the existing trajectories can find their associations, and thus is critical for the proposed hybrid association framework. Intuitively, taking more frames
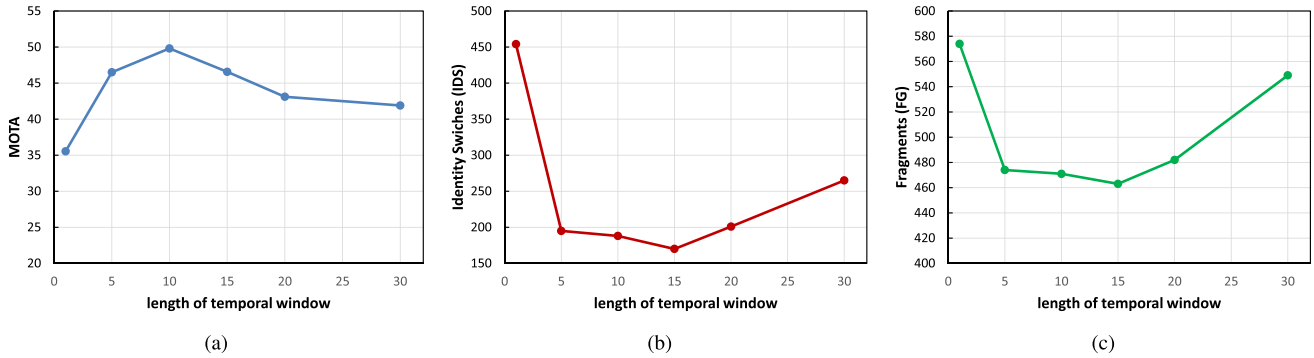
Fig. 4. Influence of the length of the temporal window (Δt) on tracking performance. (a) MOTA, (b) IDS, and (c) FG scores on the PETS dataset are plotted.
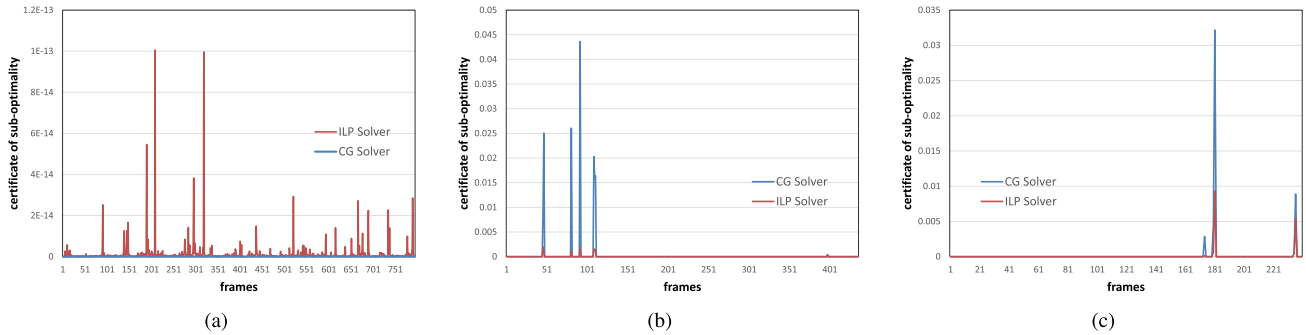

Fig. 5. Sub-optimality comparison between the CG and ILP solvers. The sub-optimality certificates are reported for each frame of (a) PETS-S2L1, (b) PETS-S2L2, and (c) PETS-S2L3 sequences, respectively. The certificates provided by the CG solver are quite small (equal to zero in most of the cases) and comparable to the ILP solver, indicating that the CG solver is stable.

into account should be helpful for handling inaccurate detections and occlusions. To study the influence of Δt on multi-object tracking performance, we conduct an experiment with $\Delta t = \{1, 5, 10, 15, 20, 30\}$ on the PETS dataset. Fig. 4 shows the MOTA, IDS, and FG scores as a function of Δt.

We can observe from Fig. 4 that enlarging the temporal window improves the overall performance and apparently reduces the number of ID switches and trajectory fragments, especially compared with the purely local method when the length of temporal window is set to $\Delta t = 1$. This result indicates the importance of the data association across multiple frames which our hybrid data association framework can leverage. As we claimed, integrating the local target-specific model with the global optimization over multiple frames is able to alleviate the irrecoverable errors caused by making decision with only local information. Inaccuracy brought by false alarms and short-term occlusions can be exactly resolved to improve the multi-object tracking performance.

On the other hand, the performance decreases when the temporal window is unduly large (> 20). The reason is that the local consistency enforced by target-specific models becomes inaccurate with a long temporal distance. Specifically, due to appearance variations, the target-specific similarity functions obtained by online learning might be inaccurate when they are used to evaluate the object appearances coming from the future. Minimizing the edge costs in the multi-commodity network is therefore unstable to produce consistent flows (trajectories). Similarly, the constant velocity model used

TABLE I
COMPARISON OF TRACKING PERFORMANCE AND CONVERGENCE SPEED OF THE CG AND ILP SOLVERS ON THE PETS DATASET

Video	CG Solver		ILP Solver	
	Run time (s)	MOTA (%)	Run time (s)	MOTA (%)
PETS-S2L1	0.0276	82.6	0.0296	82.0
PETS-S2L2	0.1571	44.2	0.2131	41.6
PETS-S2L3	0.2381	29.5	0.8156	29.3

to estimate the track start cost might provide unstable long term predictions and thus degrades the tracking accuracy. To achieve a tradeoff between local consistency and global association, we set $\Delta t = 10$ for our hybrid data association approach and keep it fixed throughout the following experiments.

E. Solver Comparison

In this paper, we introduce a column generation (CG) based solver to the min-cost multi-commodity flow problem in terms of multi-object tracking. Alternatively, one can solve the problem directly using existing integer linear programming packages. To demonstrate the superiority of the proposed CG solver over the standard ILP solver, we report the sub-optimality certificates of the solutions provided by both the CG solver and the ILP solver for the *PETS* dataset in Fig. 5. The sub-optimality certificates are computed as described in Section IV-B. For the ILP solver, we employ the commercial software Gurobi which represents the state of the art in ILP.

TABLE II

QUANTITATIVE COMPARISON RESULTS OF OUR APPROACH (DENOTED AS HYBRIDDAT) WITH OTHER STATE-OF-THE-ART METHODS ON THE *MOTChallenge 2015* DATASET. WE GROUP THE RESULT LISTINGS INTO LOCAL, GLOBAL, AND HYBRID METHODS. BOLD SCORES HIGHLIGHT THE BEST RESULTS WHILE ITALIC SCORES INDICATE THE SECOND BEST ONES. (ACCESSED ON 7/6/2016)

	Method	MOTA[%]↑	MOTP[%]↑	FAF↓	MT[%]↑	ML[%]↓	FP↓	FN↓	IDS↓	FG↓
Local	TC_ODAL [44]	15.1 ± 15.0	70.5	2.2	3.2	55.8	12,970	38,538	637	1,716
	RMOT [45]	18.6 ± 17.5	69.6	2.2	5.3	53.3	12,473	36,835	684	1,282
	MDP [4]	30.3 ± 14.6	71.3	1.7	13.0	**38.4**	9,717	32,422	680	1,500
	TDAM [6]	33.0 ± 9.8	**72.8**	1.7	13.3	*39.1*	10,065	**30,617**	464	1,506
	SCEA [7]	29.1 ± 12.2	71.1	**1.0**	8.9	47.3	**6,060**	36,912	604	1,182
Global	DP_NMS [46]	14.5 ± 13.9	70.8	2.3	6.0	40.8	13,171	34,814	4,537	3,090
	SMOT [47]	18.2 ± 10.3	71.2	1.5	2.8	54.8	8,780	40,310	1,148	2,132
	TBD [48]	15.9 ± 17.6	70.9	2.6	6.4	47.9	14,943	34,777	1,939	1,963
	CEM [26]	19.3 ± 17.5	70.7	2.5	8.5	46.5	14,180	34,591	813	1,023
	MotiCon [49]	23.1 ± 16.4	70.9	1.8	4.7	52.0	10,404	35,844	1,018	1,061
	SegTrack [50]	22.5 ± 15.2	71.7	1.4	5.8	63.9	7,890	39,020	697	**737**
	MHT_DAM [51]	32.4 ± 15.6	71.8	1.6	**16.0**	43.8	9,064	32,060	435	826
	JPDA_m [52]	23.8 ± 15.1	68.2	*1.1*	5.0	58.1	*6,373*	40,084	*365*	869
	TSMLCDE [15]	*34.3 ± 13.1*	71.7	1.4	*14.0*	39.4	7,869	31,908	618	959
Hybrid	NOMT [30]	33.7 ± 16.2	71.9	1.3	12.2	44.0	7,762	32,547	442	*823*
	HybridDAT	**35.0 ± 15.0**	*72.6*	1.5	11.4	42.2	8,455	*31,140*	**358**	1,267

Overall, the certificates provided by the CG solver are quite small (equal to zero in most of the cases) and comparable to the ILP solver, indicating that the CG solver is stable. As can be observed in Fig. 5(a), the CG solver provides zero certificates on each frame of the *PETS-S2L1* sequence, while the ILP solver provides certificates much closer to zero. It demonstrates that the CG solver exactly finds the optimal integer solution to the min-cost multi-commodity flow problem when the ILP has a tight relaxation to a LP. For the situations where the ILP is not equivalent to a LP, caused by the close interactions of multiple objects, the CG solver provides a near-optimal solution in an efficient way by using a column generation process, as shown in Fig. 5(b) and Fig. 5(c).

To further demonstrate the superiority of the CG solver in terms of multi-object tracking, we report the tracking performance (the MOTA score) and convergence speed (the average run time per frame) of both the CG solver and the ILP solver for the three sequences with varying object densities in the *PETS* dataset. Results are shown in Table I. As can be observed, the CG solver achieves better results compared with ILP with a fast speed. For each sequence, the CG solver achieves higher MOTA scores than the ILP solver, indicating that the near-optimal solutions produced by the CG solver are more meaningful for multi-object tracking. It owes to the path-flow reformulation involved in the CG solver which conducts a direct connection between the solution and the estimated trajectories. Furthermore, favorable convergence speed is provided by the CG solver even though the number of objects increases quickly from the sequence PETS-S2L1 ($\sim$ 5 objects per frame) to PETS-S2L3 ($\sim$ 30 objects per frame).

F. Comparison With the State-of-the-Art

We now compare our approach with the state-of-the-art methods on the *MOTChallenge 2015* dataset. The state-of-the-art methods are selected with available corresponding publications at the time of our submission to the test bench, including TC_ODAL [44], RMOT [45], MDP [4], SCEA [7], TDAM [6], DP_NMS [46], SMOT [47], TBD [48], CEM [26], MotiCon [49], SegTrack [50], MHT_DAM [51], JPDA_m [52], TSMLCDE [15], and NOMT [30]. Note that the TC_ODAL, RMOT, MDP, SCEA and TDAM trackers are local data-association methods, the NOMT tracker and our approach perform data association in a hybrid way, while the other trackers are global data-association methods.

Table II lists detailed quantitative comparison results on the *MOTChallenge 2015* dataset, where the results are grouped into local, global, and hybrid data-association methods.[1] With only the provided detections and a simple dynamic model, our approach shows very competitive performance with the best MOTA score. It demonstrates that our approach performs favorably over the state-of-the-art and is suitable for various unconstrained environments. In particular, the MOTA score and the number of ID switches are substantially improved compared with both local and global data-association methods. It is ascribed to the hybrid data association framework that is able to find optimal associations for the existing trajectories over multiple video frames. Errors caused by inaccurate detections and occlusions, which are the most challenging issues in complex scenes, are significantly alleviated by our approach to produce consistent trajectories.

As expected, hybrid data-association methods performs better than both local and global methods by a large margin. This superior performance is mainly due to the integration of local target-specific models and global optimization over multiple frames. Compared with the local methods, hybrid data association takes multiple frames into account and therefore is much more stable against noise. Moreover, compared with the global methods, hybrid data association utilizes the local target-specific models to ensure the local consistency of estimated trajectories, meanwhile retaining the ability to handle online data. Benefitting from the superiority of hybrid data association, the NOMT tracker also achieves good scores on the challenging dataset, as we can observe in Table II.

---

[1]The comparison is also available at the website of the MOTChallenge http://motchallenge.net/results/2D_MOT_2015/.

Fig. 6. Sample tracking results of our approach on five representative testing video sequences of the MOTChallenge 2015 dataset (*i.e.*, *PETS09-S2L2*, *ETH-Jelmoli*, *ADL-Rundle-1*, *Venice-1*, and *KITTI-19*). At each frame, we show the bounding boxes together with the past trajectories (last 30 frames). The color of the bounding boxes and trajectories indicates the ID of the tracked objects. Best viewed in color. (Refer to the tracking videos for more detailed results.)

In contrast, our approach produces apparently lower FN and IDS scores with a reasonable number of false alarms, and thus provides a better MOTA score. The reason is that our min-cost multi-commodity flow formulation models the multi-object tracking problem in a compact form and enables the efficient near-optimal solution to obtain more accurate trajectories.

On the other hand, our approach produces slightly more fragmented trajectories. The reason is that our approach can perform multi-object tracking in an online manner, even though the global optimization over multiple frames

is involved. Our approach tends to terminate the trajectory when it has no associated detections in the future frames and thus increases the FG scores. The number of ID switches is significantly reduced due to the consideration of multiple future frames, as shown in Table II.

Several qualitative examples of tracking results produced by our approach on the *MOTChallenge 2015* are shown in Fig. 6. Consistency of the estimated trajectories is indicated by bounding boxes of the same color on the same object over time. Our method is able to accurately track

the objects against the influence of abundant false positive detections, short-term occlusions, abrupt motions etc. (Videos suitable for qualitative evaluation of the results across all frames are available at the website of the MOTChallenge http://motchallenge.net/results/2D_MOT_2015/, as well as the detailed tracking results provided by our approach and the state-of-the-art algorithms.)

VI. Conclusions

In this paper, we have proposed a hybrid data association approach for multi-object tracking. Instead of only considering local associations between adjacent video frames, we explored the superior abilities of global optimization over multiple frames to carry out online tracking. It was formulated as a min-cost multi-commodity flow problem where the local target-specific information is modeled to cooperate with the global association. We employed a powerful online similarity learning algorithm to explicitly build target-specific appearance models to compute the edge costs of our multi-commodity network, improving the discriminative ability of the framework. In addition, we introduced an efficient and effective solution with empirical sub-optimality certificates, and validated its superiority in terms of multi-object tracking. Extensive experiments on various challenging datasets have demonstrated that our approach outperforms the state-of-the-art methods.

Our future work will explore more effective approaches to learn edge costs for the multi-commodity network since it is the most critical issue for good performance. Online similarity learning is just one example of using the appearance cue to compute edge costs, and we believe that our hybrid data association framework can be further improved in terms of multi-object tracking by introducing more useful cues such as motion and shape.

References

[1] M. D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, and L. van Gool, "Online multiperson tracking-by-detection from a single, uncalibrated camera," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 9, pp. 1820–1833, Sep. 2011.

[2] 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 Recognit. (CVPR)*, Jun. 2012, pp. 1815–1821.

[3] H. Possegger, T. Mauthner, P. M. Roth, and H. Bischof, "Occlusion geodesics for online multi-object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 1306–1313.

[4] Y. Xiang, A. Alahi, and S. Savarese, "Learning to track: Online multi-object tracking by decision making," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4705–4713.

[5] F. Solera, S. Calderara, and R. Cucchiara, "Learning to divide and conquer for online multi-target tracking," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4373–4381.

[6] M. Yang and Y. Jia, "Temporal dynamic appearance modeling for online multi-person tracking," *Comput. Vis. Image Understand.*, vol. 153, pp. 16–28, Dec. 2016.

[7] J. Hong Yoon, C.-R. Lee, M.-H. Yang, and K.-J. Yoon, "Online multi-object tracking via structural constraint event aggregation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1392–1400.

[8] A. Milan, S. H. Rezatofighi, A. Dick, I. Reid, and K. Schindler. (Apr. 2016). "Online multi-target tracking using recurrent neural networks." [Online]. Available: http://arxiv.org/abs/1604.03635

[9] L. Zhang, Y. Li, and R. Nevatia, "Global data association for multi-object tracking using network flows," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2008, pp. 1–8.

[10] 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.

[11] W. Luo, B. Stenger, X. Zhao, and T.-K. Kim, "Automatic topic discovery for multi-object tracking," in *Proc. AAAI Conf. Artif. Intell.*, 2015, pp. 3820–3826.

[12] V. Chari, S. Lacoste-Julien, I. Laptev, and J. Sivic, "On pairwise costs for network flow multi-object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5537–5545.

[13] A. Dehghan, S. M. Assari, and M. Shah, "GMMCP tracker: Globally optimal generalized maximum multi clique problem for multiple object tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 4091–4099.

[14] S. Tang, B. Andres, M. Andriluka, and B. Schiele, "Subgraph decomposition for multi-target tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5033–5041.

[15] B. Wang, G. Wang, K. L. Chan, and L. Wang, "Tracklet association by online target-specific metric learning and coherent dynamics estimation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 39, no. 3, pp. 589–602, Mar. 2017.

[16] S. Tang, B. Andres, M. Andriluka, and B. Schiele, "Multi-person tracking by multicut and deep matching," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2016, pp. 100–111.

[17] A. Maksai, X. Wang, F. Fleuret, and P. Fua. (2016). "Globally consistent multi-people tracking using motion patterns." [Online]. Available: https://arxiv.org/abs/1612.00604

[18] W. Luo, X. Zhao, and T.-K. Kim. (2014). "Multiple object tracking: A review." [Online]. Available: https://arxiv.org/abs/1409.7618

[19] A. Karpathy, J. Johnson, and L. Fei-Fei. (2015). "Visualizing and understanding recurrent networks." [Online]. Available: https://arxiv.org/abs/1506.02078

[20] P. Ondruska and I. Posner, "Deep tracking: Seeing beyond seeing using recurrent neural networks," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, Feb. 2016, pp. 3361–3367.

[21] A. Milan, S. H. Rezatofighi, A. Dick, I. Reid, and K. Schindler, "Online multi-target tracking using recurrent neural networks," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, Feb. 2017, pp. 4225–4232.

[22] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese, "Social lstm: Human trajectory prediction in crowded spaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 961–971.

[23] A. Sadeghian, A. Alahi, and S. Savarese. (2017). "Tracking the untrackable: Learning to track multiple cues with long-term dependencies." [Online]. Available: https://arxiv.org/abs/1701.01909

[24] C. Huang, B. Wu, and R. Nevatia, "Robust object tracking by hierarchical association of detection responses," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*, 2008, pp. 788–801.

[25] B. Yang and R. Nevatia, "Multi-target tracking by online learning of non-linear motion patterns and robust appearance models," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2012, pp. 1918–1925.

[26] A. Milan, S. Roth, and K. Schindler, "Continuous energy minimization for multitarget tracking," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 1, pp. 58–72, Jan. 2014.

[27] A. Milan, K. Schindler, and S. Roth, "Multi-target tracking by discrete-continuous energy minimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 10, pp. 2054–2068, Oct. 2016.

[28] B. Yang and R. Nevatia, "Multi-target tracking by online learning a CRF model of appearance and motion patterns," *Int. J. Comput. Vis.*, vol. 107, no. 2, pp. 203–217, 2014.

[29] P. Lenz, A. Geiger, and R. Urtasun, "FollowMe: Efficient online min-cost flow tracking with bounded memory and computation," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4364–4372.

[30] W. Choi, "Near-online multi-target tracking with aggregated local flow descriptor," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3029–3037.

[31] H. B. Shitrit, J. Berclaz, F. Fleuret, and P. Fua, "Multi-commodity network flow for tracking multiple people," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 8, pp. 1614–1627, Aug. 2014.

[32] A. Dehghan, Y. Tian, P. H. Torr, and M. Shah, "Target identity-aware network flow for online multiple target tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 1146–1154.

[33] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*. Upper Saddle River, NJ, USA: Prentice–Hall, 1993.

[34] G. Chechik, V. Sharma, U. Shalit, and S. Bengio, "Large scale online learning of image similarity through ranking," *J. Mach. Learn. Res.*, vol. 11, pp. 1109–1135, Jan. 2010.

[35] K. Crammer, O. Dekel, J. Keshet, S. Shalev-Shwartz, and Y. Singer, "Online passive-aggressive algorithms," *J. Mach. Learn. Res.*, vol. 7, pp. 551–585, Dec. 2006.

[36] G. B. Dantzig and P. Wolfe, "Decomposition principle for linear programs," *Oper. Res.*, vol. 8, no. 1, pp. 101–111, 1960.

[37] A. Schrijver, *Theory Linear Integer Programming*. Hoboken, NJ, USA: Wiley, 1998.

[38] L. R. Ford, Jr., and D. R. Fulkerson, "A suggested computation for maximal multi-commodity network flows," *Manage. Sci.*, vol. 5, no. 1, pp. 97–101, 1958.

[39] A. Ellis, A. Shahrokni, and J. M. Ferryman, "PETS2009 and Winter-PETS 2009 results: A combined evaluation," in *Proc. IEEE Int. Workshop Perform. Eval. Tracking Surveillance (PETS-Winter)*, Dec. 2009, pp. 1–8.

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

[41] Y. Li, C. Huang, and R. Nevatia, "Learning to associate: Hybridboosted multi-target tracker for crowded scene," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2009, pp. 2953–2960.

[42] R. Girshick, J. Donahue, T. Darrell, and J. Malik, "Region-based convolutional networks for accurate object detection and segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 1, pp. 142–158, Jan. 2016.

[43] A. Babenko and V. Lempitsky, "Aggregating local deep features for image retrieval," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 1269–1277.

[44] S. Bae and K. Yoon, "Robust online multi-object tracking based on tracklet confidence and online discriminative appearance learning," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 1218–1225.

[45] J. H. Yoon, M.-H. Yang, J. Lim, and K.-J. Yoon, "Bayesian multi-object tracking using motion context from multiple objects," in *Proc. IEEE Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2015, pp. 33–40.

[46] H. Pirsiavash, D. Ramanan, and C. C. Fowlkes, "Globally-optimal greedy algorithms for tracking a variable number of objects," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2011, pp. 1201–1208.

[47] C. Dicle, O. I. Camps, and M. Sznaier, "The way they move: Tracking multiple targets with similar appearance," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2013, pp. 2304–2311.

[48] A. Geiger, M. Lauer, C. Wojek, C. Stiller, and R. Urtasun, "3D traffic scene understanding from movable platforms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 5, pp. 1012–1025, May 2014.

[49] L. Leal-Taixé, M. Fenzi, A. Kuznetsova, B. Rosenhahn, and S. Savarese, "Learning an image-based motion context for multiple people tracking," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2014, pp. 3542–3549.

[50] A. Milan, L. Leal-Taixé, K. Schindler, and I. Reid, "Joint tracking and segmentation of multiple targets," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 5397–5406.

[51] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg, "Multiple hypothesis tracking revisited," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 4696–4704.

[52] S. H. Rezatofighi, A. Milan, Z. Zhang, Q. Shi, A. Dick, and I. Reid, "Joint probabilistic data association revisited," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Dec. 2015, pp. 3047–3055.

Min Yang received the B.S. and Ph.D. degrees from the Beijing Institute of Technology in 2010 and 2016, respectively. His research interests include computer vision, pattern recognition and machine learning.

Yuwei Wu received the Ph.D. degree in computer science from the Beijing Institute of Technology (BIT), Beijing, China, in 2014. From 2014 to 2016, he was a Post-Doctoral Research Fellow with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. He is currently an Assistant Professor with the School of Computer Science, BIT. His research interests are in computer vision and information retrieval. He received outstanding Ph.D. Thesis Award from BIT and the Distinguished Dissertation Award Nominee from the China Association for Artificial Intelligence.

Yunde Jia (M'11) received the M.S. and Ph.D. degrees in mechatronics from the Beijing Institute of Technology (BIT), Beijing, China, in 1986 and 2000, respectively. He has served as the Executive Dean of the School of Computer Science, BIT, from 2005 to 2008. He was a a Visiting Scientist with Carnegie Mellon University, Pittsburgh, PA, USA, from 1995 to 1997, and a Visiting Fellow with The Australian National University, Acton, Australia, in 2011. He is currently a Professor of computer science with BIT, and the Director of the Beijing Laboratory of Intelligent Information Technology, School of Computer Science. His current research interests include computer vision, media computing, and intelligent systems.