Pattern Recognition 129 (2022) 108738

Contents lists available at ScienceDirect

Pattern Recognition

journal homepage: www.elsevier.com/locate/patcog

An end-to-end identity association network based on geometry refinement for multi-object tracking

Rui Li[a], Baopeng Zhang[a], Zhu Teng[a,*], Jianping Fan[b]

[a] School of Computer and Information Technology, Beijing Jiaotong University, Beijing, 100044, China
[b] AI Lab, Lenovo Research, Beijing, 100085, China

ARTICLE INFO

Article history:
Received 12 May 2021
Revised 10 October 2021
Accepted 23 April 2022
Available online 7 May 2022

Keywords:
Multi-object tracking
Interactions
Occlusions
Data association
Identity verification

ABSTRACT

In multi-target tracking, object interactions and occlusions are two significant factors that affect tracking performance. To settle this, we propose an identity association network (IANet) that integrates the geometry refinement network (GRNet) and the identity verification (IV) module to perform data association and reason the mapping between the detections and tracklets. In our data association process, the object drifts caused by object interactions are suppressed effectively by encoding the direction and velocity of objects to refine the geometric position of tracklets. The tracklets with refined geometric information are further utilized in the IV module to achieve a sufficient encoding of multivariate spatial cues including both appearance and geometry information, which defeats the misleading impacts of interactions and occlusions dramatically in multi-object tracking. The extensive experiments and comparative evaluations have demonstrated that our proposed method can significantly outperform many state-of-the-art methods on benchmarks of 2D MOT2015, MOT16, MOT17, MOT20, and KITTI by using public detection and online settings.

1. Introduction

Multi-object tracking (MOT) is becoming more and more attractive in the computer vision community due to its applications in video surveillance, virtual reality and human-computer interaction, etc. MOT aims to simultaneously locate multiple objects of interest in a given video, maintain the IDs of these objects, and record their trajectories. In recent years, with the improvements of the technologies for object detection, tracking-by-detection algorithms have become mainstream for MOT, where object detection is first performed on each frame, and data association is then conducted to create the trajectory of each object in a video.

In such tracking-by-detection approaches, learning a discriminative data association method is critical, as it determines whether the tracker can correctly distinguish different trajectories of objects. Most existing methods are mainly based on similarity measures or assignments for data association, but they have complex structures and require a large number of labeled data [1,2]. Some methods such as [3] employ the regressor of an object detector for data association to ease the costs of annotations. However, without handling the interactions and occlusions in tracking, it is difficult to accurately distinguish the trajectory of each object in the crowded and occluded scenes, as in such situations obvious appearance and geometry variations often occur.

Interactions can be regarded as short-term occlusions between objects. That is, all objects with interactive behavior can be detected in adjacent frames, where the direct utilization of regressor [3] will cause the identities to drift, and the IDs of interactive objects will be exchanged with each other. To this end, we design a geometry refinement network (GRNet) for MOT driven by the awareness of interactions. In our work, the optical flow of adjacent frames is calculated and implicitly applied as the direction and velocity of moving objects, and then the relative motion is captured to refine the geometry position of objects in the next frame. As shown in Fig. 1(a) and (b), the spatial topology of ID_1 and ID_2 changes due to the interaction. In this case, the method [3] shown in Fig. 1(c) fails to maintain their identities, whereas ours (Fig. 1(d)) can accurately track the interactive objects.

In contrast to interactions, occlusions specifically refer to the long-term occlusions caused by other objects or surrounding backgrounds, and the occluded objects can not be detected in adjacent frames. The occlusions usually cause the tracking objects to disappear and a new ID will be assigned when they appear again. Many existing methods deal with the occlusions by the re-identification operation. They only consider the distance of appearance features between objects without modeling dependencies of

* Corresponding author.
E-mail address: zteng@bjtu.edu.cn (Z. Teng).

https://doi.org/10.1016/j.patcog.2022.108738
0031-3203/

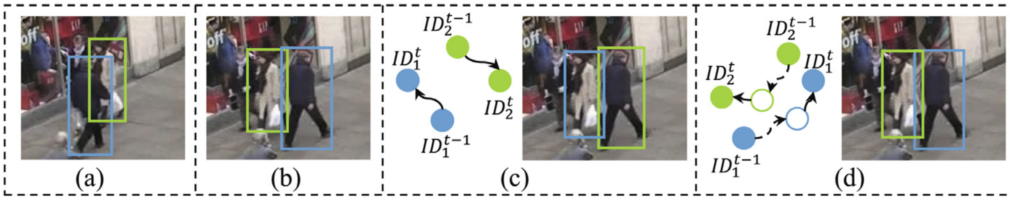

Fig. 1. A stimulus example of our proposed identity association network based on geometry refinement for data association. (a): Interactive objects of the frame $t-1$; (b): The ground truth of the frame t; (c): The tracking result by using regression in the frame t; (d): The tracking result by using IANet in the frame t. When the objects interact and the spatial topology of the interactive objects in two adjacent frames changes as shown in (a) and (b), the data association method in (c) fails to maintain the identities of ID_1 and ID_2. In contrast, our proposed IANet in (d) can still effectively capture the variation of spatial topology and accurately track the interactive objects.

contexts such as the spatial deployments, so that the performance is not satisfactory. In addition, the re-identification network and the data association model are often independent of each other, which leads to a tedious tracking process. In recent years, more and more researchers have begun to explore other clues such as the positional relationship and topology structure of objects to enhance the discrimination of object representations [1,4], yet these methods rely on complex structures to model the spatial multivariate cues, which is incapable of the efficiency requirement of MOT. Based on these observations, we propose an identity verification (IV) module to suppress the misjudgments caused by interactions and occlusions. This model is integrated into the GRNet for end-to-end learning, which fully excavates the spatial relationship and learns the mapping between detections and tracklets through a lightweight network to verify the identities of objects. Here, a tracklet is a track fragment in the process of generating the complete trajectory of an object, and it only contains the tracking information of an object in one or several frames.

In this work, we propose a novel identity association network (IANet) for MOT driven by the awareness of interactions and occlusions, which mainly consists of two parts: (1) designing a geometry refinement network (GRNet) to implicitly encode the direction and velocity of objects for refining the geometry information of tracklets in the data association process; (2) constructing a lightweight identity verification (IV) module to learn the mapping between tracklets with refined geometry information and detections to verify the identities of objects, which is integrated into the GRNet for end-to-end learning.

To summarize, our main contributions are two-fold:

- We propose an end-to-end identity association network (IANet) including a geometry refinement network (GRNet) with an identity verification (IV) module, which fully mines the temporal and spatial clues of objects and directly outputs the tracking results without dells and whistles.
- To alleviate the misleading impacts caused by interactions and occlusions, we propose a GRNet for refining the geometry information of each object in the data association by encoding the direction and velocity of objects, and the relationship reasoning between tracklets with refined geometry information and detections is performed by a lightweight IV module to further verify the identities of tracklets.

Moreover, extensive experiments on five widely-employed benchmarks including 2D MOT2015, MOT 16, MOT17, MOT20, and KITTI are executed and the results demonstrate that the proposed algorithm can significantly outperform numerous state-of-the-art methods.

2. Related work

Tracking-by-Detection Paradigm: Multi-object tracking is still a challenging task due to the crowded and complex environments. The existing works mainly follow the paradigm of tracking-by-detection, and the performance is heavily affected by object detection. Traditional MOT methods are mainly based on Bayesian probabilistic models [5,6], Markov decision making [7] and other methods. However, these methods design the features manually, which may result in complex structures and unsatisfactory performance. Recently, with the development of deep learning, a lot of object detection algorithms are developed based on convolutional neural networks [8,9], and MOT researchers focus their attention on the object association problem by employing these detection algorithms. The existing object trackers can be divided into two groups based on whether they employ the knowledge of future frames in the data association: (a) online methods [1,3,10–12]; and (b) offline methods [13–17].

Most of these online and offline methods concentrate on designing object representations and utilize these features for similarity measures or assignments. For example, in [18], a Siamese network is developed to extract the appearance features of objects and calculate the matching similarity between two detections. The optical flow-like features extracted by deep networks are used to achieve better multi-target tracking results in [2]. A feature fusion algorithm is designed based on Long Short Term Memory (LSTM) network to learn the matching similarity between the historical information of the trajectory and the current detection in [1]. Even though these models design complex structures to deduce object features, their performances are not satisfactory due to not fully exploiting other cues besides appearances from the complex scenes in MOT datasets. In recent years, more and more researchers have considered building simple and efficient methods to deal with the task of multi-object tracking from different perspectives. For instance, [3] proposes a method that utilizes the regression operation of object detection for data association, which is simple and the performance is excellent. Despite noticeable progress, this method is unlikely to distinguish objects in the crowded and occlusive scenes (where interactions, occlusions, obvious appearance variations often occur) due to the lack of the components for calculating the motion vectors of objects and modeling the dependencies of context. Unlike these existing approaches using similarity measures or assignments for data association, we propose an end-to-end identity association network based on geometry refinement, which is aware of interactions and occlusions to effectively maintain the trajectories of objects as completely as possible in a crowded environment.

Interactions and Occlusions in MOT: Due to the crowded and complex scenes in multi-object tracking, there often exist interactions between objects, and objects may be occluded by the surrounding background in the sequence. The interactions and occlusions are very challenging in MOT because they make it difficult to discriminate objects and increase the possibility of ID switches. However, studies on these issues are still insufficient. In the current exhaustive literature, researchers have little specific analysis on these issues and generally treat them as a common occlusion problem. Most methods handle the common occlusion problem by making use of better detectors [19,20] or borrowing models from person re-identification [3]. For example, a short-term re-identification based on a Siamese network is utilized to cope with the re-identification after occlusions [3]. In [21], the occlusions are managed from the perspective of feature learning and a Spatial-Temporal Attention Mechanism (STAM) is proposed to recognize the occlusion situation and identify possible relevant objects. An object re-identification network trained by a triplet loss is employed in [3] to deal with the occlusion problem. These methods only consider the appearance features of objects, or recover salient foreground areas within the detected bounding box, but do not take into account the impacts of the object contexts. Different from the above methods, we deliberate the interactions and occlusions and propose a novel method for MOT driven by the awareness of interactions and occlusions. In the proposed method, we attempt to handle the interactions by the data association process and infer the identities of the objects through an identity verification module based on the relationship matrix.

Recurrent Neural Networks: The Recurrent Neural Network networks have been successfully applied in many fields such as natural language processing, and computer vision to obtain the dependencies between various elements. In computer vision tasks, from video object detection [22], action recognition [23] to video object segmentation [24], RNNs have significantly improved the performance of these tasks, because the potential relationships between multiple objects and the relationships between objects at different periods can be inferred through the relation network. Inspired by this, the application of RNNs in multi-object tracking has attracted more attention [1,15,25]. Kim et al. [25] proposes a novel model Bilinear LSTM to improve the learning of long-term appearance models via a recurrent network, which is utilized for the multi-object tracking task. Xiang et al. [15] propose a deep conditional random field network for MOT in the crowded scenes and employ a bidirectional long short-term memory (LSTM) network to encode the long-term dependencies. Different from these methods, we construct a relationship matrix by excavating the appearance and geometry clues between tracklets with refined geometry information and detections, and then utilize a lightweight identity verification block based on Bi-RNNs to encode the dependencies between node pairs in the matrix to verify the identities of objects, which is integrated into GRNet for end-to-end learning.

3. Our proposed method

Given a frame-by-frame video sequence, the goal of MOT is to predict the trajectories of N objects over time, and we denote the trajectories as $T = \{T_i\}_{i=1}^{N}$. The trajectory of each object can be represented by a series of ordered object bounding boxes $T_i = \left\{b_i^t\right\}_{t=1}^{\mathcal{T}}$, $b_i^t = \left[x_i^t, y_i^t, w_i^t, h_i^t\right]$, where t denotes a frame of the video sequence, x_i^t and y_i^t represent the center location of the ith object at frame t, and w_i^t and h_i^t are the width and height, respectively. For the convenience of explanations, we further define $B^t = \left\{b_1^t, b_2^t, ..., b_m^t\right\}$ as the set of bounding boxes of tracklets in the frame t and use $D^t = \left\{d_1^t, d_2^t, ..., d_k^t\right\}$ to represent the detections in the frame t, where m and k represent the number of tracklets and detections in the frame t.

The pipeline of our proposed method in the MOT inference is presented in Fig. 2. We use noisy detections obtained by the object detector as the starting point and accomplish MOT through two steps. For the current frame t with the detected bounding boxes D^t, we design a GRNet to tackle the data association between the previous frame $t-1$ and the current frame t by modeling the relative motion of objects, which encodes the optical flow as the direction and velocity of moving objects, and utilizes a linear regression to refine the geometry location of tracklets in the current frame. Then, the spatial relationship between detections D^t and tracklets B^t obtained by GRNet is excavated and the IV module is employed to learn the mapping between D^t and B^t in the frame t for verifying the identities of objects. Specially, we develop the GRNet that encodes the optical flow into the geometry information of objects (described in Section 3.1). The optical flow perceives the subtle variations of moving objects in our data association stage, which preliminarily suppresses drifts caused by interactions of objects and brings a more accurate localization. To discriminate against each object and alleviate the impacts caused by interactions and occlusions in MOT, relationship learning is conducted based on the lightweight IV module (see more details in Section 3.2.2), which automatically learns the mapping between the tracklets with refined geometry information and the detections by encoding the constructed relationship matrix (described in Section 3.2.1). Finally, we elaborate on the end-to-end learning of the whole IANet in the training phase (described in Section 3.3) and the process of MOT inference (3.4). Note that GRNet provides the refined tracklets and corresponding appearance features for the IV module to learn the mapping between the detections and tracklets, i.e., the proposed GRNet and IV module are correlated rather than separate models, and they cooperate to realize the multi-object tracking.

3.1. Geometry refinement network

Directly applying regression on data association has been proved to be effective [3] in MOT. However, the performance is restricted by the object detection method that only involves the appearance features of objects. This leads to object drifts during interactions of objects and affects the performance of MOT. To handle the drifts, we propose a GRNet for data association, which encodes the optical flow into the geometry information including direction and velocity of moving objects to guide the process of refining the geometry location of tracklets.

The proposed GRNet aims to estimate tracklets of objects and refine the geometry locations of these objects from tracklets B^{t-1} at frame $t-1$ to B^t at frame t through the guidance of optical flow information and linear regression. As shown in the green dotted box of Fig. 2, the lightweight NetC encoder and NetE decoder [26] are used to estimate the optical flow of adjacent frames and tracklets B^{t-1} of the previous frame are roughly refined through a wrapping operation $\mathcal{G}_w$. Meanwhile, the combination of ResNet101 [27] and Feature Pyramid [28] is employed as the backbone to extract semantic features in our GRNet. After that, the ROIPooling and linear regression $\mathcal{G}_r$ are employed to obtain the refined locations of tracklets in the current frame.

$$b_i^t = \mathcal{G}_r(\mathcal{G}_w(b_i^{t-1}, \triangle p_i^t), \triangle b_i^t) \tag{1}$$

Specifically, in the proposed GRNet, the geometric locations b_i^t of an object i in the current frame are refined by $\mathcal{G}_w$ and $\mathcal{G}_r$, as described in Eq. (1) and Fig. 3. $\mathcal{G}_w$ and $\mathcal{G}_r$ are formulated in Eq. (2), where $\triangle p^t$ is the corresponding flow motion vectors of an object obtained by b^{t-1} and the optical flow map. $\triangle b^t$ represents the refined information learned from the linear regression.

$$\begin{aligned}
&\mathcal{G}_w(b^{t-1}, \triangle p^t) = b^{t-1} + \triangle p^t, \\
&\triangle p^t = \left[\Delta x(p^t), \Delta y(p^t), \Delta w(p^t), \Delta h(p^t)\right] \\
&\mathcal{G}_r(p^t, \triangle b^t) = p^t + \triangle b^t, \\
&\triangle b^t = \left[\Delta x(b^t), \Delta y(b^t), \Delta w(b^t), \Delta h(b^t)\right]
\end{aligned} \tag{2}$$

3.2. Identity verification module

In our work, in order to obtain tracklets of the current frame, we learn the relative motion of objects through the proposed GRNet, yet the complex environment with frequent interactions and

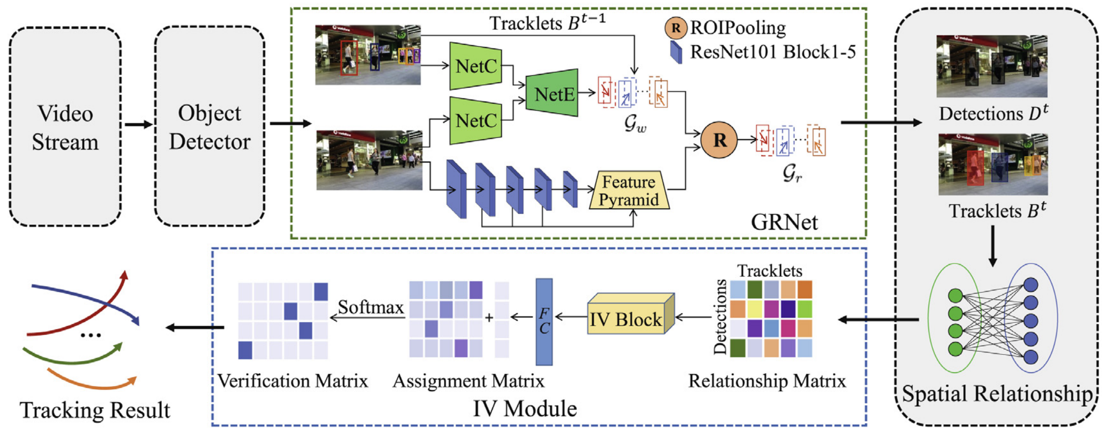

Fig. 2. The pipeline of the proposed IANet in the MOT inference. We formulate the multi-object tracking by the proposed GRNet and IV module, where GRNet performs the preliminary data association and the IV module learns the mapping between refined tracklets obtained by GRNet and detections to verify the identities of objects. (For interpretation of the references to color in the text, the reader is referred to the web version of this article.)

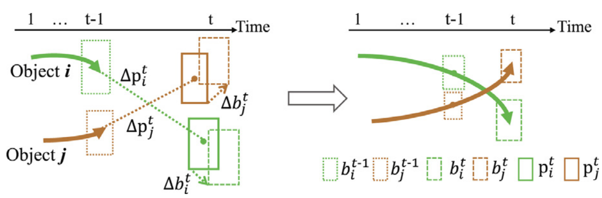

Fig. 3. The process of geometry refinement by the proposed GRNet, where the position b_i^{t-1}, b_j^{t-1} of objects i, j in the frame $t-1$ are utilized to refine the positions b_i^t, b_j^t of objects i and j in the frame t. The left panel shows the details of the refinement process and the right panel is the result after geometry refinement. Here, Δp^t and Δb^t are refined information learned from the optical flow map and the linear regression, respectively.

occlusions in MOT may lead to drifts or identity switches, which stimulates us to verify the identities of objects after obtaining tracklets. We design a more comprehensive spatial relationship matrix (3.2.1) and propose a lightweight identity verification module integrated into GRNet to learn the mapping between tracklets and detections adaptively (3.2.2).

3.2.1. Design of the relationship matrix

In order to improve the precision of the mapping between tracklets and detection, we designed a more comprehensive relationship matrix, which fully excavates the appearance and geometric clues of bounding boxes in the spatial domain. In the existing literature, Intersection-over-Union (IoU) is the most commonly used metric to measure the similarity between bounding boxes. However, due to the complexity of MOT scenes, the IoU between the bounding boxes of the same object obtained by the tracker and detector is slight, which easily affects the accuracy of match results. In addition, just employing IoU for the match such as [29] is easily leads to serious ID switches, which intensely violates our motivation. To this end, we present a new formulation to generate the relationship matrix between tracklets and detections by fusing multiple spatial clues. As shown in Fig. 4, the proposed fusion distance can accurately measure the diversity between tracklets and detection, yet the IoU (evaluated by Jaccard distance) easily causes the phenomenon that the distance between wrong pairs is smaller than the right ones.

In our work, the relationship matrix M_r is constructed by modeling the fusion distance between tracklets and detections, and s_{ij} is the fusion distance of b_j (the bounding box of jth tracklet) and d_i (the bounding boxe of ith detection) as described in Eq. (3).

$$\begin{aligned} s_{ij} = \mathcal{D}_1 & (b_j, d_i) + \mathcal{D}_2(b_j, d_i) \\ s.t. \quad & 0 < i \leqslant k, \\ & 0 < j \leqslant m. \end{aligned} \tag{3}$$

where $\mathcal{D}_1(,)$, $\mathcal{D}_2(,)$ represent the functions of appearance distance and geometric distance respectively, m and k represent the number of tracklets and detections in the current frame.

We model the appearance distance by extracting deep features of bounding boxes as shown in Eq. (4), where $\mathcal{F}(\cdot)$ represents the deep feature obtained by GRNet. In GRNet, a fully connected layer is added after the ROIPooling layer, and we use the output of this fully connected layer as the deep feature in the training stage. In the inference, to suppress the noise interference caused by the tracking failure of an object in a certain frame, the average of deep features of the tracklet in the last 10 frames is utilized to calculate the appearance distance. For the geometric distance, we consider the overlap as well as the center distance to alleviate the misjudgment caused by inaccurate bounding boxes. As presented in Eq. (5), the Euclidean distance of center points and Jaccard distance are introduced, where $c(\cdot)$ represents the center point of a bounding box, $\cup$ and $\cap$ denote the union and intersection of two bounding boxes.

$$\mathcal{D}_1(b_j, d_i) = avg\left(\left\|\mathcal{F}(b_j) - \mathcal{F}(d_i)\right\|^2\right) \tag{4}$$

$$\begin{aligned} \mathcal{D}_2(b_j, d_i) = & \frac{1}{2}\left(\frac{\left\|c(b_j) - c(d_i)\right\|^2}{\sqrt{H^2 + W^2}}\right) \\ & + \frac{1}{2}\left(\frac{\left|b_j \cup d_i\right| - \left|b_j \cap d_i\right|}{\left|b_j \cup d_i\right|}\right) \end{aligned} \tag{5}$$

3.2.2. Learning to verify identities

An outline of the IV module is described in the blue area of Fig. 2, the relationship matrix of the current frame is transported to this lightweight network to learn the mapping $\mathcal{M}_v$ between tracklets and detections. We handle this problem by an IV block which conducts the row-wise and column-wise flattening in turn and conveying them to two Bi-RNNs for global reasoning. The IV block encodes the row-wise intermediate assignments and column-wise definitive assignments, respectively. In particular, the final soft assignment matrix $\mathcal{M}_a$ is obtained using the IV block and

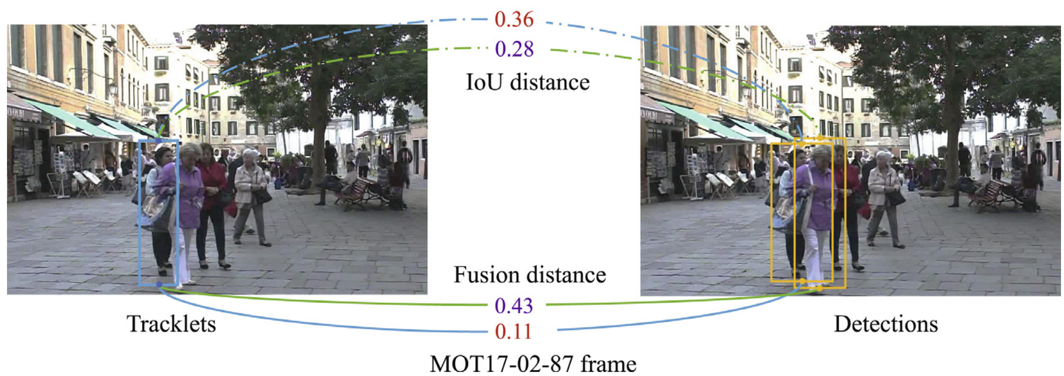

Fig. 4. The fusion distance between tracklets and detections. For the pedestrian with a blue bounding box, the IoU distance (evaluated by Jaccard distance) to its corresponding detection is 0.36, which is larger than another false detection (0.28). However, by the metric of the fusion distance, the distance between the correct tracklet and detection (0.11) is obviously smaller than the incorrect pair (0.43). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

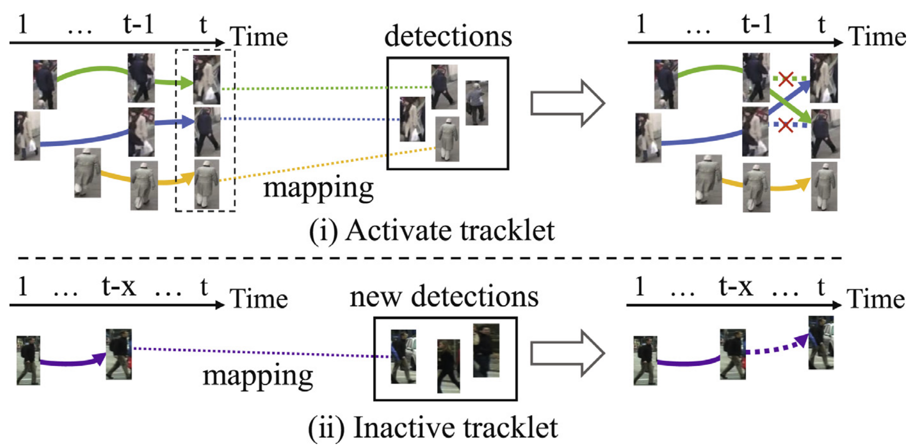

Fig. 5. The visualization on two situations of learning the identity verification in the tth frame. (i): The identity of each active tracklet (black dotted box) in the tth frame is verified by the unique mapped detection. When the identities of tracklets drift (green and blue tracklets), their identities can be rectified through the mapping so that the correct tracklets can be obtained as indicated by the gray arrow. (ii): If there is a unique mapping between a new detection and an inactive tracklet, the inactive tracklet is activated with this new detection. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

has the same size $k \times m$ as the input relationship matrix $\mathcal{M}_r$. Finally, in order to obtain the final unique mapping verification matrix $\mathcal{M}_v$, we append a column of the threshold 0.5 to the assignment matrix and then execute a row-wise softmax.

According to the mapping results, we verify the identities of tracklets in the current tth frame, which can be divided into two situations as shown in Fig. 5:

(i) We verify the identities of the active tracklets obtained by GRNet in the tth frame (black dotted box), and the identity of a tracklet that uniquely maps with a detection is refined by the corresponding detection. We can observe that when the identities of tracklets obtained by GRNet drift, the identities can be rectified through the IV module so that the correct tracklets can be obtained.

(ii) Then we use the new detections to verify the inactive tracklet. If there is a unique mapping between a new detection and an inactive tracklet that disappeared in the $(t-x)$th frame, we assign the detection to the corresponding inactive track and reactivate this tracklet. Here, x is a time interval and its maximum value is set to 20. That is, IANet can effectively address the re-identification of the occluded objects within 20 frames. If the object is occluded continuously for more than 20 frames, our model may fail to associate this occluded object.

Intuitively, these two situations of learning identity verification can not only further restrain the drifts caused by short-time interactions but also realize the re-identification of objects with long-term occlusions to reduce identity switches.

3.3. End-to-End training

3.3.1. GRNet loss

By integrating the geometry refinement network and identity verification module mentioned above, we construct the end-to-end trainable identity association network for multi-object tracking as shown in Fig. 6. In the training phase of this IANet, the input pairs of the network are I^t and $I^{t'}$, where I^t is the original image of frame t from the dataset sequence and $I^{t'}$ comes from the aggregate as described Eq. (6).

$$I^{t'} \in \left\{ \widetilde{I^t}, I^{t-1} \right\}, \quad \widetilde{I^t} = tran(I^t) \tag{6}$$

Here I^{t-1} is the original image of frame $t-1$. To improve the generalization and robustness of the proposed method, data augmentation is conducted by a transformation operation of randomly scaling and translating, and we use $tran(I^t)$ to represent the image obtained by this transformation operation.

As shown in Fig. 6, the training objective of the GRNet is defined as Eq. (7), which consists of the binary classification loss $\mathcal{L}_c(\hat{q})$ and the regression loss $\mathcal{L}_r(\hat{o})$.

$$\mathcal{L}_1(\hat{q}, \hat{o}, y, o) = \mathcal{L}_c(\hat{q}, y) + \mathcal{L}_r(\hat{o}, o) \tag{7}$$

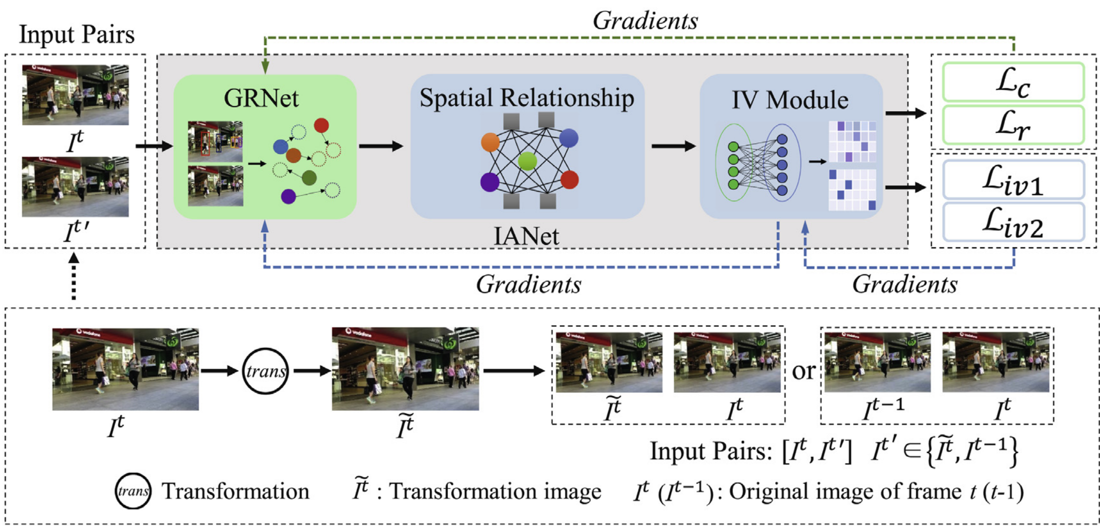

Fig. 6. The training process of the whole identity association network (IANet). The data augmentation (as shown in the bottom dotted box) is conducted by the transformation operation of randomly scaling and translating to increase the training samples, and four loss functions $\mathcal{L}_c$, $\mathcal{L}_r$, $\mathcal{L}_{iv1}$, $\mathcal{L}_{iv2}$ are employed to constrain the whole IANet.

where $\hat{q}$ is the presence confidence obtained by the classification branch of GRNet and $\hat{o}$ is the predicted parameterized coordinates by the regression branch of GRNet. y is the classification label of the ground truth and o represents the parameterized coordinates of the ground truth.

The GRNet learns to predict the object presence confidence by means of the cross-entropy loss (Eq. (8)) and regresses the bounding box of the object with smooth L1 loss (Eq. (9)).

$$\mathcal{L}_c(\hat{q}_i, \mathrm{y}_i) = -(\log\left[\mathrm{y}_i\hat{q}_i + (1-\mathrm{y}_i)(1-\hat{q}_i)\right]) \tag{8}$$

$$\mathcal{L}_r(\hat{o}_i, o_i) = \begin{cases} 0.5 * (\hat{o}_i - o_i)^2 & \left|\hat{o}_i - o_i\right| < 1 \\ \left|\hat{o}_i - o_i\right| - 0.5 & otherwise \end{cases} \tag{9}$$

Where y_i is the classification ground truth of the bounding box i. If i is an object, we have $\mathrm{y}_i = 1$; otherwise $\mathrm{y}_i = 0$.

In the regression branch of GRNet, we follow the parameterizations of the 4 coordinates [9] and adopt a strategy of the regression from a rough box to a nearby ground-truth box as shown in Eq. (10).

$$\begin{aligned} \hat{o}_x &= (\hat{x} - x_p)/w_p, & \hat{o}_y &= (\hat{y} - y_p)/h_p \\ \hat{o}_w &= \log(\hat{w}/w_p), & \hat{o}_h &= \log(\hat{h}/h_p) \\ o_x &= (x - x_p)/w_p, & o_y &= (y - y_p)/h_p \\ o_w &= \log(w/w_p), & o_h &= \log(h/h_p) \end{aligned} \tag{10}$$

where $x, y, w,$ and h denote the center coordinates, width, and height of the bounding box. $[\hat{x}, \hat{y}, \hat{w}, \hat{h}]$, $[x_p, y_p, w_p, h_p]$ and $[x, y, w, h]$ represent the predicted box obtained by GRNet, rough box obtained by the wrapping described in Section 3.1, and the ground truth, respectively.

3.3.2. IV module loss

In order to improve the precision of MOT, we design a lightweight IV module to learn the mapping between tracklets obtained by GRNet and detections. Based on the observation that the module can learn the unique mapping and output the soft mapping probability, we regard the training process of the IV module as a 2D binary classification task and the loss is described in Eq. (11), which consists of two focal losses applied to the appearance distance matrix Eq. (12) and geometric distance matrix Eq. (13) to measure the probability distributions.

$$\mathcal{L}_2(\hat{s1}, \hat{s2}, s^*) = \mathcal{L}_{iv1}(\hat{s1}, s^*) + \mathcal{L}_{iv2}(\hat{s2}, s^*) \tag{11}$$

where $\hat{s1}$ is the classification probability of the appearance distance matrix described in Eq. (4) obtained by the IV module. $\hat{s2}$ is the classification probability of the geometric distance matrix described in Eq. (5). $\hat{s}$ denotes the approximate ground truth matrix obtained by HA [30].

$$\mathcal{L}_{iv1}(\hat{s1}, s1^*) = \begin{cases} -\alpha(1-\hat{s1})^\gamma \log \hat{s1}, & s1^* = 1 \\ -(1-\alpha)(\hat{s1})^\gamma \log(1-\hat{s1}), & s1^* = 0 \end{cases} \tag{12}$$

$$\mathcal{L}_{iv2}(\hat{s2}, s2^*) = \begin{cases} -\alpha(1-\hat{s2})^\gamma \log \hat{s2}, & s2^* = 1 \\ -(1-\alpha)(\hat{s2})^\gamma \log(1-\hat{s2}), & s2^* = 0 \end{cases} \tag{13}$$

where α is an adaptive weighting factor obtained by the number of zeros n_0 and ones n_1 in the approximate ground truth matrix. $\gamma = 2$ represents the focusing parameter that smoothly adjusts the rate at which easy examples are down-weighted.

$$\alpha = n_0/(n_0 + n_1) \tag{14}$$

3.3.3. Overall loss

By integrating the GRNet and IV module mentioned above, we construct the end-to-end trainable IANet for MOT. The loss of the whole network is shown in Eq. (15) by combining Eqs. (7) and (11) together, where β is set to 1 to balance the GRNet loss and IV Module Loss.

$$\begin{aligned} \mathcal{L}_{total} &= \mathcal{L}_1(\hat{q}, \hat{o}, \mathrm{y}, o) + \beta\mathcal{L}_2(\hat{s1}, \hat{s2}, s^*) \\ &= \mathcal{L}_c(\hat{q}, \mathrm{y}) + \mathcal{L}_r(\hat{o}, o) + \beta\mathcal{L}_{iv1}(\hat{s1}, s^*) + \beta\mathcal{L}_{iv2}(\hat{s2}, s^*) \end{aligned} \tag{15}$$

3.4. MOT inference

Our proposed MOT inference algorithm runs IANet to produce the trajectories of all objects whose details are described in Algorithm 1. In summary, there are 6 steps in our inference algorithm:

Step 1. The detections in the first frame are refined by the GRNet of IANet and we obtain the detection set D^1 by comparing the classification scores and the thresh $\delta_{score} = 0.5$. Then the set D^1 is utilized to initialize the active tracklet set T_{active}. (Line 3-7)

Step 2. For the current frame t, the same strategy as the 1th frame is employed in the GRNet to obtain the refined detection set D^t and the bounding boxes B^t of active tracklets. (Line 8)

Algorithm 1 The proposed IANet tracking algorithm.

Input: ~~

Video sequence with $\mathcal{T}$ frames $I = \left\{I^t\right\}_{t=1}^{\mathcal{T}}$;

The detection set $D_{ori} = \left\{D^t_{ori}\right\}_{t=1}^{\mathcal{T}}$.

Output: ~~

Trajectories $T = \{T_i\}_{i=1}^{N}$ of all objects, where $T_i = \left\{b^t_i\right\}_{t=1}^{T}$, $b^t_i = \left[x^t_i, y^t_i, w^t_i, h^t_i\right]$.

1: $T, T_{active} \leftarrow \varnothing$
2: **for** I^t, D^t_{ori} in zip(I, D_{ori}) **do**
3:   **if** $t == 1$ **then**
4:     $(D^t, None) \leftarrow$ GRNet$(None, I^t, D^t_{ori}, None)$
5:     Initialize the active tracklets T_{active} with D^t
6:     **continue**
7:   **end if**
8:   $(D^t, B^t) \leftarrow$ GRNet$(I^{t-1}, I^t, D^t_{ori}, B^{t-1})$
9:   $\mathcal{M}_r \leftarrow$ excavate the fusion distance between tracklets and detections.
10:   $\mathcal{M}_v \leftarrow$ IV module$(\mathcal{M}_r)$
11:   $B^t \leftarrow$ update by assigning the uniquely matched detected boxes to the corresponding bounding boxes of B^t obtained by Line 9.
12:   $T \leftarrow$ update according to the tracklets which do not match any detections and the classification score smaller than δ_{score}.
13:   $T_{active} \leftarrow$ update according to T and B^t.
14:   $D^t_{unm} \leftarrow \mathcal{M}_v$.
15:   $T, T_{active}, D^t_{unm} \leftarrow$ update according to the verification matrix between D^t_{unm} and T obtained by IV module.
16:   $T_{active} \leftarrow$ initialize new trajectories with D^t_{unm}.
17: **end for**
18: $T \leftarrow T + T_{active}$.
19: **return** T;

Step 3. We excavate the fusion distance between tracklets and detections and construct the relationship matrix $\mathcal{M}_r$. Then, the IV module of IANet is used to learning the mapping between tracklets and detections and generate the verification matrix $\mathcal{M}_v$. We replace the corresponding bounding boxes of B^t with the uniquely matched detected boxes according to $\mathcal{M}_v$, and the updated B^t is used to update the T_{active}. Finally, we terminate the tracklet of an object which does not match any detection and the classification score smaller than δ_{score}. Meanwhile, we add the inactive tracklet to T, and remove it from T_{active}. (Line 9-13)

Step 4. We find the detections D^t_{unm} that do not match any tracklets, and use the IV module of IANet to learn the mapping between D^t_{unm} and T. If there is a detection in D^t_{unm} uniquely matched with an object in T, we add the detection to the corresponding tracklet of the object, remove the object from T and add it to T_{active} again. (Line 14-15)

Step 5. Detections for activating the inactive tracklets are removed from D^t_{unm}, and the final new detections in D^t_{unm} are initialized as new trajectories. (Line 16)

Step 6. After traversing the whole video sequence, the tracklets in T_{active} are migrated to T to produce the trajectories of all objects. (Line 18)

4. Experimental results

The performance of the proposed method is demonstrated on several widely used multi-object tracking datasets including 2D MOT2015, MOT16, MOT17, MOT20, and KITTI. The detailed information of these benchmarks are disclosed in Section 4.1, and the implementation details of our proposed method are described in Section 4.2. To prove the effectiveness of our various components, we conduct the ablation study in Section 4.3. Our proposed method is further compared with several state-of-the-art tracking methods in Section 4.4.

4.1. Datasets

In this paper, we evaluate the performance of our proposed method utilizing five MOT benchmarks 2D MOT2015, MOT16, MOT17, MOT20 and KITTI. They are challenging because of the large variety in scene, lighting, pedestrian pose, viewpoint, etc.

2D MOT2015: This dataset is composed of 11 training sequences and 11 testing sequences, and both of them are provided with ACF [31] detections.

MOT16: MOT16 includes 7 training sequences and 7 testing sequences with DPM [32] detections. Pedestrians in the MOT16 scenes are more crowded, so it is more challenging than 2D MOT2015 dataset.

MOT17: MOT17 has the same sequences as MOT16 but with three public detections including DPM [32] detections, Faster RCNN [9] detections and SDP [33] detections. By providing three detection results, different MOT methods can be compared more comprehensively.

MOT20: MOT20 contains 8 novel challenging sequences including 4 training video sequences and 4 testing video sequences. This benchmark with Faster RCNN [9] detections addresses the challenge of very crowded scenes in which the density can reach values of 246 pedestrians per frame.

KITTI: KITTI [34] is a novel challenging real-world computer vision benchmark, which can address multiple tasks including stereo, optical flow, visual odometry, object detection, and tracking. The multi-object tracking of KITTI consists of 21 training videos and 29 test videos and it uses 2D bounding box overlap to evaluate tracking results.

Evaluation Metric: In this paper, the standard metrics of MOT Benchmarks are adopted for evaluation, including Multiple Object Tracking Accuracy (MOTA) [30], ID F1 Score (IDF1) [35], the number of Identity Switch (IDS) [36], Mostly Tracked objects (MT, the ratio of ground-truth trajectories covered by an output trajectory for at least 80% of ground truth length), Mostly Lost objects (ML, the ratio of ground-truth trajectories covered by an output trajectory for at most 20% of ground truth length), False Positives (FP) and False Negatives (FN).

4.2. Implementation details

In the training process, this GRNet consists of a flow embedding component and a feature embedding component. The structure of flow embedding follows Liteflownet [26] which is pre-trained on MPI-Sintel [37] and is kept fixed during the training of our proposed network. We employ a structure that combines ResNet101 [27] + FPN [28] to extract features. The parameters of regression layer are initialized by using the Faster RCNN pre-trained on ImageNet and the learning rate is initialized as $1e-4$, divided by 10 every 5 epochs. The parameters of two Bi-RNNs in the IV block are initialized by the result pre-trained on the data set in [38] and the setting of learning rate is the same as the feature embedding. We use the Adam optimizer to train the whole end-to-end trainable IANet for 25 epochs and the batch size is set to 1 on the training data set. It takes about 16 min to train our model on about 6700 frames of MOT17.

4.3. Ablation study

The ablation study on the MOT17 benchmark is presented in this section. We split each training sequences into two parts, and use the first half (MOT17-02, MOT17-04, MOT17-09, MOT17-10,

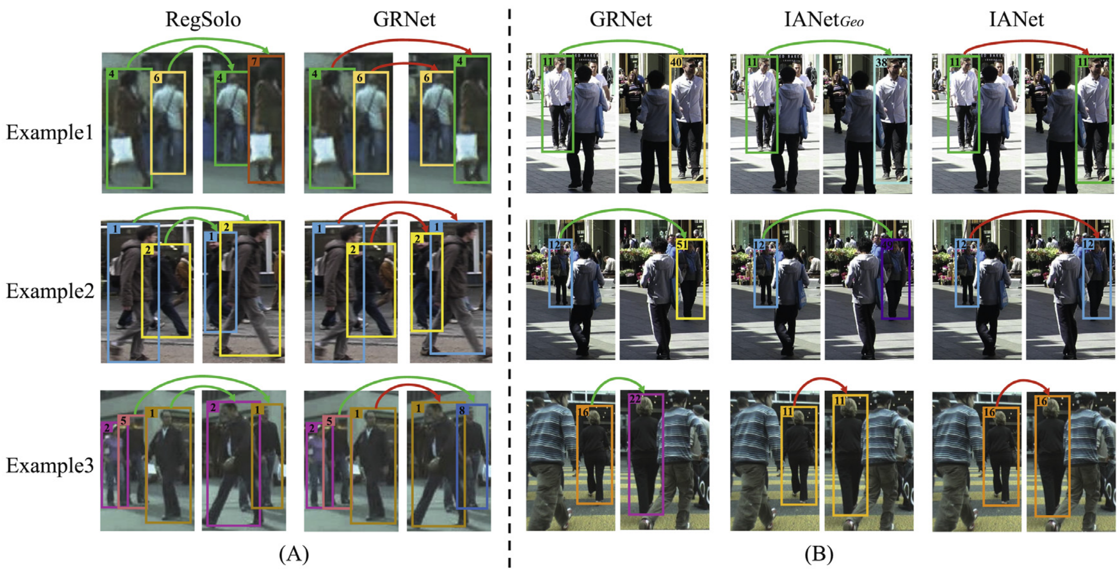

Fig. 7. (A): Tracking results performed by RegSolo and GRNet under the circumstances of interactions. (B): Tracking results performed by GRNet, IANet$_{Geo}$, and IANet under occlusions. Three scenes are presented, where the object disappears and reappears due to occlusions. The red arrow means that the identity does not change after the interactions or occlusions, while the green arrow indicates that the identity switch occurs after the interaction or occlusions. For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 1
The ablation study on the GRNet and the IV module.

Method	MOTA↑	IDF1↑	MT↑	ML↓	FP↓	FN↓	IDS↓
RegSolo	56.4	58.7	99	**32**	2,413	6,854	460
GRNet	60.4	63.6	**124**	33	2,996	**6,127**	382
IANet	**62.6**	**63.9**	102	33	**1,125**	6,230	**314**

Table 2
The ablation study on the fusion distance in IV module.

Method	MOTA↑	IDF1↑	MT↑	ML↓	FP↓	FN↓	IDS↓
IANet$_{Geo}$	57.4	53.4	95	48	**633**	8,499	469
IANet$_{App}$	61.4	61.9	**102**	35	1,222	6,917	385
IANet	**62.6**	**63.9**	**102**	**33**	1,125	**6,230**	**314**

and MOT17-11) for training, the rest ones (MOT17-05, and MOT17-13) for validation. We show the effectiveness of our GRNet and IV module in Table 1. We remove the GRNet and the IV module separately to prove the effectiveness of these two modules. In order to examine the significance of the flow-guided geometric information and the linear regression, we retrain the Faster RCNN network without optical flow and only accept the regression operation for data association which is named RegSolo.

For the IV module, to evaluate the effectiveness of the proposed fusion distance, four comparison models are estimated to evaluate the performance as shown in Table 2, where IANet$_{Geo}$ and IANet$_{App}$ indicate that the trackers only employ the geometry distance and appearance distance to construct the relationship matrix, respectively.

Impacts of GRNet. As shown in Table 1, compared with the RegSolo model, the GRNet further refines MOTA by 4.0% (from 56.4% to 60.4%) and boosts IDF1 by 4.9% (from 58.7% to 63.6%). other metrics such as MT, FN, and IDS are improved by the GRNet as well, especially the number of IDS, which is reduced by 78 (from 460 to 382). These results show that even with frequent interactions between objects, the flow-guided geometric information can still suppress the potential drifts of objects and ensure the integrity of each tracking trajectory as much as possible.

To present the impacts of interactions more vividly in MOT, we visualize the outputs of RegSolo and GRNet, and select three examples for analysis as shown in Fig. 7(A). The results of two frames before and after the interaction are presented, different pedestrians are represented by the bounding boxes with different colors. The IDs are marked in the upper left corner and the red arrow means that the example has the same ID before and after the interaction, while the green arrow indicates an ID switch occurs after the interaction. It can be observed that the original IDs cannot be maintained in the situation of interactions when using the RegSolo model. In contrast, the identities of these three examples are mostly maintained in the GRNet model, which indicates that this model can effectively suppress the drifts because of interactions.

Impacts of IV Module. Compared to the IANet with the GRNet, the contribution of IV module can be observed as a drastic reduction of IDS (from 382 to 314) and the improvements of the MOTA (from 60.4% to 62.6%), IDF1 (from 63.6% to 63.9%), FP (from 2,996 to 1,125) are achieved. In other words, the use of IV module to learn the mapping between tracklets obtained by GRNet and detections for identity verification can further improve the performance of the tracker.

Effectiveness of the fusion distance. As shown in Table 2, the model IANet, which utilizes the fusion distance including appearance and geometric clues for identity verification, obtains accuracies of 62.6% and 63.9% measured by MOTA and IDF1. It outperforms the model IANet$_{Geo}$ with geometry distance by a margin of 5.2% and 10.5% in MOTA and IDF1. Specifically, the IANet has a significant decline of IDS (from 469 to 314). Furthermore, although the overall performance of IANet$_{App}$ only using the appearance distance is higher than that of IANet$_{Geo}$, there is still a certain gap compared with IANet, which suggests that the fusion distance benefits the recognition of objects in the interaction and occlusion situation.

To illustrate the impacts of occlusions in MOT, we also provide a visualization on the outputs of GRNet, IANet$_{Geo}$, and IANet, and select three examples for analysis as shown in Fig. 7(B). The results of two frames before and after the occlusion are presented, the red arrow means that the example has the same ID before and after the occlusion, while the green arrow indicates an ID switch occurs after the occlusion. It can be observed from Fig. 7(B) that

Table 3
Comparisons of the tracking performance on the benchmarks of 2D MOT2015, MOT16 and MOT17.

	Method	MOTA↑	IDF1↑	MT(%)↑	ML(%)↓	FP↓	FN↓	IDS↓
	2D MOT2015							
Offline	QuadMOT [13]	33.8	40.4	12.9	36.9	7,898	32,061	703
	TPM [14]	36.2	43.6	15.4	42.6	<u>5,650</u>	33,102	420
	CRFTrack [15]	40.0	49.6	23.0	28.6	10,295	25,917	658
	LifTsimInt [16]	47.2	57.6	27.0	29.8	7,635	24,277	554
	MPNTrack [17]	<u>51.5</u>	<u>58.6</u>	<u>31.2</u>	<u>25.9</u>	7,620	<u>21,780</u>	<u>375</u>
Online	AMIR [1]	37.6	46.0	15.8	**25.8**	7,933	29,397	**1,026**
	STRN [12]	38.1	46.6	11.5	33.4	**5,451**	31,571	1,033
	TrctrD15 [38]	44.1	46.0	17.2	26.6	6,085	26,917	1,347
	Tracktor+ [3]	44.1	46.7	18.0	26.2	6,477	26,577	1,318
	Ours	**47.5**	**47.8**	**23.4**	26.4	5,531	**25,502**	1,240
	MOT16							
Offline	GCRA [39]	48.2	48.6	12.9	41.1	5,104	88,586	821
	CRFTrack [15]	50.3	54.4	18.3	35.7	7,148	82,746	702
	TPM [14]	51.3	47.9	18.7	40.8	<u>2,701</u>	85,504	569
	LifTsimInt [16]	57.5	<u>64.1</u>	25.4	<u>34.7</u>	4,249	72,868	<u>335</u>
	MPNTrack [17]	<u>58.6</u>	61.7	<u>27.3</u>	34.0	4,949	<u>70,252</u>	684
Online	DASOT16 [10]	46.1	49.4	14.6	41.6	8,222	89,204	802
	Deep-TAMA [11]	46.2	49.4	14.1	44.0	5,126	92,367	**598**
	AMIR [1]	47.2	46.3	14.0	41.6	**2,681**	92,856	774
	MOTDT [40]	47.6	50.9	15.2	38.3	9,253	85,431	792
	STRN [12]	48.5	53.9	17.0	**34.9**	9,038	84,178	747
	Tracktor+ [3]	54.4	52.5	19.0	36.9	3,280	79,149	682
	TrctrD16 [38]	54.8	53.4	19.1	37.0	2,955	78,765	645
	Ours	**56.5**	**55.4**	**20.6**	35.4	2,736	**75,922**	611
	MOT17							
Offline	jCC [41]	51.2	54.5	20.9	37.0	25,937	247,822	1,802
	CRFTrack [15]	53.1	53.7	24.2	<u>30.7</u>	27,194	234,991	2,518
	TPM [14]	54.2	52.6	22.8	37.5	<u>13,739</u>	242,130	1,824
	LifTsimInt [16]	58.2	<u>65.2</u>	28.6	33.6	16,850	217,944	<u>1,022</u>
	MPNTrack [17]	<u>58.8</u>	61.7	<u>28.8</u>	33.5	17,413	<u>213,594</u>	1,185
Online	MTDF17 [42]	49.6	45.2	18.9	**33.1**	37,124	241,768	5,567
	Deep-TAMA [11]	50.3	53.5	19.2	37.5	25,479	252,996	2,192
	STRN [12]	50.9	56.5	20.1	37.0	27,532	246,924	2,593
	DADM [43]	51.4	53.7	16.5	34.9	21,042	251,873	2,319
	Tracktor+ [3]	53.5	52.3	19.5	36.6	12,201	248,047	2,072
	TrctrD17 [38]	53.7	53.8	19.4	36.6	11,731	247,447	1,947
	Ours	**56.6**	**57.2**	**22.1**	34.3	**10,502**	**232,640**	**1,722**

the ID switches of the three examples occur when using the model without IV module (GRNet model). That is to say, the pedestrian that reappears after occlusions is regarded as a new object. For the IANet$_{Geo}$ model, only the identity of example3 is maintained, because this model only utilizes the geometry distance to construct the relationship matrix for identity verification. Different from the previous two models, the identities of these three examples are maintained in our proposed IANet model, which indicates that the fusion distance applied in the relationship matrix is effective to verify the identification of objects, thereby improves the performance of MOT.

4.4. Comparison with state-of-the-arts

In this subsection, we compare our method against numerous state-of-the-art MOT methods on the test sequences of 2D MOT2015, MOT16, MOT17, MOT20, and KITTI benchmarks. For the fairness of comparison, the compared state-of-the-art methods and ours employ the same public detections provided by 2D MOT2015, MOT16, MOT17, and MOT20 benchmarks in all experiments. The quantitative results are divided into two groups including online and offline tracking methods, the underline and bold values highlight the best results of offline and online methods, respectively. '↑' means the higher the better, and '↓' represents the lower the better.

Results on 2D MOT2015: The comparison results on 2D MOT2015 are presented in Table 3 where a total of seven metrics are employed to evaluate the performance. For offline methods, our proposed method is superior to most offline methods. Especially, ours achieves 47.5% in terms of MOTA, which exceeds the second-best offline tracker LifTsimInt by a gain of 0.3% and indicates that the proposed method possesses the ability to balance various factors. Although the best offline method MPNTrack exceeds ours, whose performance benefits from processing all frames in the whole video sequence, yet our method with no knowledge of future frames only involves the previous frames. Therefore our model satisfies the requirements of online tracking.

Among all the online methods, the proposed method ranks number one in terms of MOTA, IDF1, MT, and FN and exceeds the second best online trackers Tracktor++ by a gain of 3.4%, 1.1%, 5.4%, and 1,075, respectively. MOTA is the main metric to comprehensively reflect the performance of one tracker, and a better MOTA indicates that the proposed method possesses the ability to balance various factors including FP, MT, and IDS. The IDF1 metric is the ratio of correctly identified detections over the average of ground-truth and computed detections. The best performance evaluated by this measure indicates that the proposed method possesses the discriminative ability to accurately identify the detections and keep the identity unchanged. Our superior performance assessed by MT suggests that the fragments are largely reduced and the proposed tracker can track more complete trajectories of objects. FN represents the total number of false negatives and the excellent performance of ours on this metric fully indicates that the tracker has fewer misjudgments. Meanwhile, trackers that win the first place examined by the other three metrics (ML, FP, and IDS) are AMIR and STRN. The proposed method performs better than AMIR and SCEA in almost all the other metrics. For example, in terms of MOTA, the proposed method out-

Table 4
Comparisons of the tracking performance on the benchmark of MOT20.

	Method	MOTA↑	IDF1↑	MT(%)↑	ML(%)↓	FP↓	FN↓	IDS↓
Offline	MPNTrack [17]	57.6	59.1	38.2	22.5	16,953	201,384	1,210
Online	SORT20 [44]	42.7	45.1	16.7	26.2	**27,521**	264,694	4,470
	MLT [45]	48.9	**54.6**	30.9	22.1	45,660	216,803	2,187
	TBC [46]	54.5	**50.1**	33.4	19.7	37,937	195,242	2,449
	Ours	**55.4**	52.7	**40.9**	**17.8**	38,973	**189,715**	**2,159**

Table 5
Comparisons of the tracking performance on the KITTI multi-object tracking benchmark for cars.

	Method	MOTA↑	MT(%)↑	ML(%)↓	FP↓	FN↓	IDS↓
Online	JCSTD [47]	80.2	57.1	7.9	6,217	405	**173**
	mmMOT [48]	83.2	72.9	2.9	4,284	752	733
	SMAT [49]	83.6	62.8	6.0	5,254	**175**	198
	QuasiDense [50]	84.9	69.5	3.9	4,320	549	313
	Ours	**87.6**	**83.9**	**2.0**	**2,671**	1,314	283

Table 6
Comparisons of methods based on relative motion modeling on 2D MOT2015, MOT16, and MOT17 datasets.

	Method	MOTA↑	IDF1↑	MT(%)↑	ML(%) ↓	FP↓	FN↓	IDS↓	Hz↑
2D MOT2015	TrctrD15 [38]	44.1	46.0	17.2	26.6	6,085	26,917	1,347	2.9
	Tracktor+ [3]	44.1	46.7	18.0	**26.2**	6,477	26,577	1,318	2.8
	Ours	**47.5**	**47.5**	**23.4**	26.4	**5,531**	**25,502**	**1,240**	**3.2**
MOT16	Tracktor+ [3]	54.4	52.5	19.0	36.9	3,280	79,149	682	**1.6**
	TrctrD16 [38]	54.8	53.4	19.1	37.0	2,955	78,765	645	**1.6**
	Ours	**56.5**	**55.4**	**20.6**	**35.4**	**2,736**	**75,922**	**611**	1.5
MOT17	Tracktor+ [3]	53.5	52.3	19.5	36.6	12,201	248,047	2,072	**1.5**
	TrctrD17 [38]	53.7	53.8	19.4	36.6	11,731	247,447	1,947	**1.5**
	Ours	**56.6**	**57.2**	**22.1**	**34.3**	**10,502**	**232,640**	**1,722**	**1.5**

performs AMIR and SCEA by a significant gain of 9.9% and 9.4%, respectively.

Results on MOT16: The proposed method is further evaluated on MOT16 against 16 existing methods including both online and offline approaches, and experimental results are shown in Table 3. The proposed method obtains 56.5%, 55.4%, 20.6%, 35.4%, 2,736, 75,922, and 611 examined by the seven metrics, respectively and it exceeds most existing methods, which means that the proposed tracker is more robust than these multi-object tracking approaches. By comparing with online methods, our proposed method obtains the best performance in terms of the metrics MOTA, IDF1, MT, and FN. Although Deep-TAMA acquires the best performance for the metric IDS, yet there are only 13 switches less than our method. Moreover, ours is superior to Deep-TAMA in other metrics. In particular, Our superior performances assessed by IDF1 and MT adequately suggest the abilities to maintain the integrity of trajectories and restrain the identity switches.

Results on MOT17: We also examine the proposed method against 20 existing online and offline methods on MOT17 to demonstrate our performance. The results are reported in Table 3 where our method is superior to most offline methods. It is worth noting that even though the performance of LifTsimInt and MPNTrack is better than ours, it cannot be applied in online tracking. Our method utilizes fewer clues than these offline methods, while the scope of application is wider and can realize online tracking.

Furthermore, compared with the online methods, the proposed tracker obtains the best performance in the six measures MOTA (56.6%), IDF1 (57.2%), MT (22.1%), FP (10,502), FN (232,640), and IDS (1,722). In terms of ML, although ours is slightly lower than MTDF17, it occupies a leading position in other metrics compared with MTDF17. For instance, our method outperforms MTDF17 by a large margin of 7.0% in the MOTA metric, which suggests the overall performance of a tracker. Concretely speaking, the proposed method has achieved excellent performance on the measure MT for evaluating the integrity of trajectories and the measure IDS for evaluating identity consistency. We attribute the performance increase of MT to the ability of our GRNet to suppress the drifts. And at the meantime, taking into consideration identity verification when linking trajectories makes our tracker achieve more truthful predictions, which inevitably reduces the ID switches.

Results on MOT20: MOT20 is a newly released dataset with more crowded and intricate scenes. To prove the generalization ability of our method, we further evaluate the performance in the unconstrained environments of the MOT20 dataset, which are reported in Table 4. It is clear that our method still outperforms the existing online methods in the comprehensive metric MOTA. In terms of MT, our performance (40.9%) is superior to all online and offline methods, which proves that our method can still ensure the integrity of trajectories to the maximum extent in a crowded environment. Considering experimental results on 2D MOT2015, MOT16, MOT17, and MOT20 benchmarks, our proposed tracker maintains an excellent tracking performance even in complex and crowded scenarios.

Results on KITTI: To validate the generalization of the proposed IANet on different types of objects, we compare the tracking performance of ours with several state-of-the-arts on the KITTI multi-object tracking benchmark for cars. As reported in Table 5, we achieve competitive results even by using a public detector Faster RCNN [9]. It is clear that our method outperforms existing online methods measured by MOTA, MT, ML, and FP, which proves the effectiveness of the proposed method. The superiority of MT and ML demonstrates that the proposed method can maintain the integrity of trajectories for cars as much as possible. Note that other methods compared in Table 5 utilize a private detector, which can boost MOTA, FP, and FN.

Results on Methods for Modeling Relative Motion: In our work, we propose a novel new tracker based on relative motion. In order to verify the effectiveness of our proposed tracker, two existing state-of-the-arts also based on the relative motion are com-

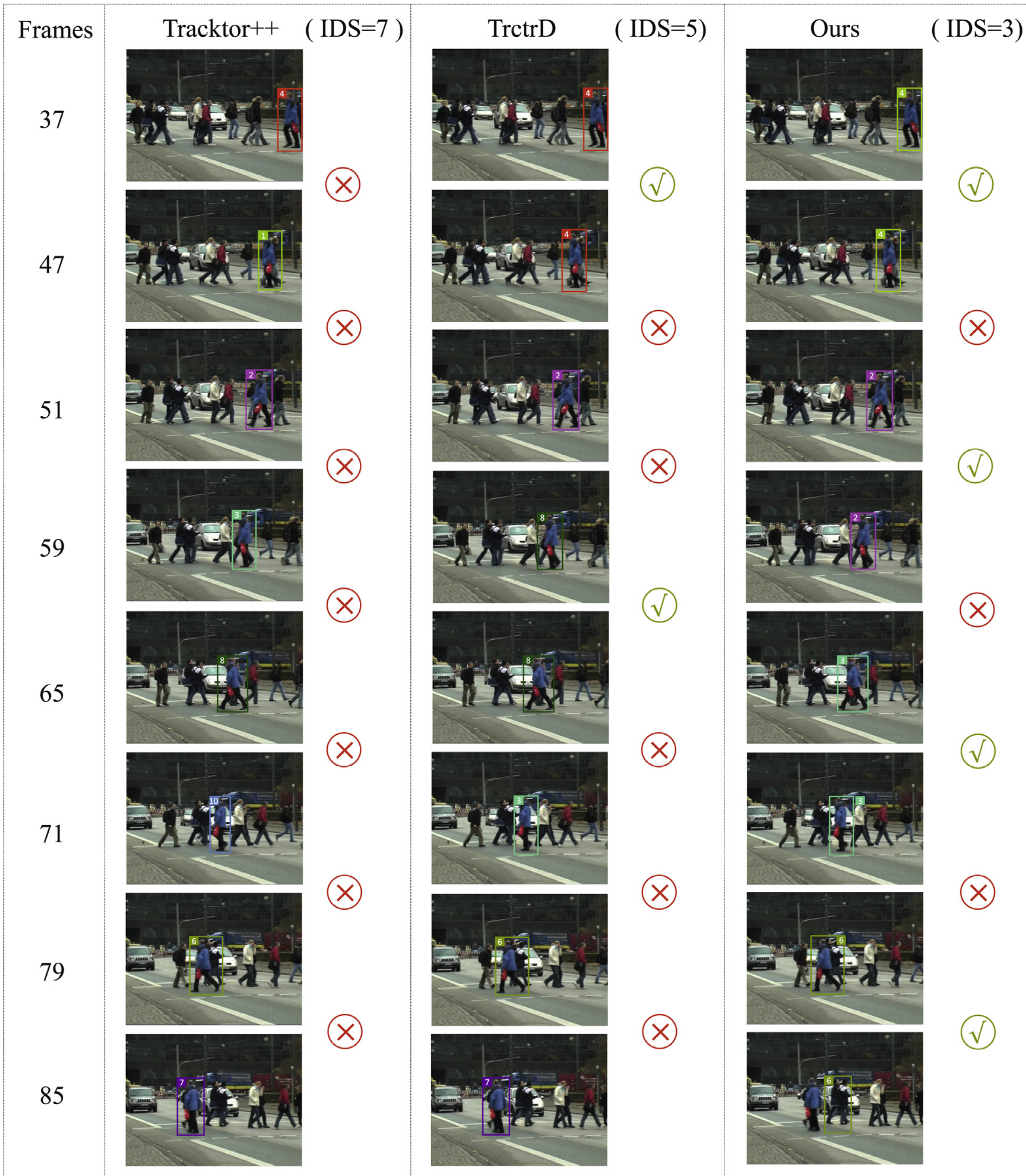

Fig. 8. An example of tracking results obtained by three different trackers based on the relative motion where the pedestrian has frequent interactions. A red circle indicates an identity switch. For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

pared in Table 6. From these quantitative results, it can be observed that our proposed method is superior to the other two approaches Tracktor++, TrctrD on 2D MOT2015, MOT16, and MOT17 data sets on the premise of ensuring the tracking speed. Evaluated on the three datasets, the average speed of our model is 2.07Hz, while the average speed of the other two methods is 1.97Hz, 2.0Hz. The superiority of IDF1, MT, and IDS in our method fully reflects the favourable influence of IANet on the tracking performance. In order to reveal the diversity among the three trackers more intuitively, we visualize three tracking results of an object with frequent interactions in Fig. 8. There are seven switches for this specific object in Tracktor++ and five switches in TrctrD from 37th frame to 85th frame. By contrast, in the same time interval, there are only three switches in our tracker, where the power of our proposed GRNet to suppress drifts is put to the proof.

5. Conclusion

In this paper, we analyze the multi-object tracking problem from the perspectives of interactions and occlusions and propose a novel multi-object tracking method IANet consisting of a GR-Net and an IV module. The GRNet is designed to suppress the drifts caused by the interactions between objects. It encodes the direction and velocity of objects to guide the localization. We also propose an IV module based on Bi-RNN integrated into the GR-Net to alleviate the misleading impacts from interactions and occlusions, where both the spatial appearance and geometric clues are exploited to reason the mapping between tracklets and detections. Our extensive experiments and comparative evaluations have demonstrated that our proposed method can significantly outperform numerous state-of-the-art methods on benchmarks of

2D MOT2015, MOT16, MOT17, MOT20, and KITTI by using public detection and online settings.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgment

This work was supported by the Natural Science Foundation of China (61972027) and the Beijing municipal Natural Science Foundation under Grant (Grant No. 4212041).

References

[1] A. Sadeghian, A. Alahi, S. Savarese, Tracking the untrackable: learning to track multiple cues with long-term dependencies, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 300–311.
[2] S. Tang, B. Andres, M. Andriluka, B. Schiele, Multi-person tracking by multicut and deep matching, in: European Conference on Computer Vision, 2016, pp. 100–111.
[3] P. Bergmann, T. Meinhardt, L. Leal-Taixe, Tracking without bells and whistles, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 941–951.
[4] Q. Chu, W. Ouyang, H. Li, X. Wang, B. Liu, N. Yu, Online multi-object tracking using CNN-based single object tracker with spatial-temporal attention mechanism, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 4836–4845.
[5] D. Reid, An algorithm for tracking multiple targets, IEEE Trans. Autom. Control 24 (6) (1979) 843–854.
[6] M.D. Breitenstein, F. Reichlin, B. Leibe, E. Koller-Meier, L. Van Gool, Robust tracking-by-detection using a detector confidence particle filter, in: Proceedings of the IEEE International Conference on Computer Vision, 2009, pp. 1515–1522.
[7] Y. Xiang, A. Alahi, S. Savarese, Learning to track: online multi-object tracking by decision making, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 4705–4713.
[8] J. Redmon, S. Divvala, R. Girshick, A. Farhadi, You only look once: unified, real-time object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 779–788.
[9] S. Ren, K. He, R. Girshick, J. Sun, Faster R-CNN: towards real-time object detection with region proposal networks, in: Advances in Neural Information Processing Systems, 2015, pp. 91–99.
[10] Q. Chu, W. Ouyang, B. Liu, F. Zhu, N. Yu, DASOT: a unified framework integrating data association and single object tracking for online multi-object tracking, in: Proceedings of the AAAI Conference on Artificial Intelligence, 2020, pp. 10672–10679.
[11] Y.-C. Yoon, D.Y. Kim, Y.-m. Song, K. Yoon, M. Jeon, Online multiple pedestrians tracking using deep temporal appearance matching association, Inf. Sci. 561 (2020) 326–351.
[12] J. Xu, Y. Cao, Z. Zhang, H. Hu, Spatial-temporal relation networks for multi--object tracking, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 3988–3998.
[13] J. Son, M. Baek, M. Cho, B. Han, Multi-object tracking with quadruplet convolutional neural networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 5620–5629.
[14] J. Peng, T. Wang, W. Lin, J. Wang, E. Ding, TPM: multiple object tracking with tracklet-plane matching, Pattern Recognit. (2020) 107480.
[15] J. Xiang, G. Xu, C. Ma, J. Hou, End-to-end learning deep CRF models for multi--object tracking, IEEE Trans. Circuits Syst. Video Technol. (2020) PP(99):1–1.
[16] A. Hornakova, R. Henschel, B. Rosenhahn, P. Swoboda, Lifted disjoint paths with application in multiple object tracking, in: International Conference on Machine Learning, 2020, pp. 4364–4375.
[17] G. Brasó, L. Leal-Taixé, Learning a neural solver for multiple object tracking, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2020, pp. 6247–6257.
[18] L. Leal-Taixé, C. Canton-Ferrer, K. Schindler, Learning by tracking: siamese CNN for robust target association, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2016, pp. 33–40.
[19] W. Hu, X. Li, W. Luo, X. Zhang, S. Maybank, Z. Zhang, Single and multiple object tracking using log-Euclidean Riemannian subspace and block-division appearance model, IEEE Trans. Pattern Anal. Mach. Intell. 34 (12) (2012) 2420–2440.
[20] G. Shu, A. Dehghan, O. Oreifej, E. Hand, M. Shah, Part-based multiple-person tracking with partial occlusion handling, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 1815–1821.
[21] Q. Chu, W. Ouyang, H. Li, X. Wang, B. Liu, N. Yu, Online multi-object tracking using CNN-based single object tracker with spatial-temporal attention mechanism, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 4836–4845.
[22] S. Wang, H. Lu, Z. Deng, Fast object detection in compressed video, in: Proceedings of the IEEE International Conference on Computer Vision, 2019, pp. 7104–7113.
[23] Y. Li, C. Lan, J. Xing, W. Zeng, C. Yuan, J. Liu, Online human action detection using joint classification-regression recurrent neural networks, in: European Conference on Computer Vision, Springer, 2016, pp. 203–220.
[24] C. Ventura, M. Bellver, A. Girbau, A. Salvador, F. Marques, X. Giro-i Nieto, RVOS: end-to-end recurrent network for video object segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 5277–5286.
[25] C. Kim, F. Li, J.M. Rehg, Multi-object tracking with neural gating using bilinear LSTM, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 200–215.
[26] T.-W. Hui, X. Tang, C. Change Loy, LiteflowNet: a lightweight convolutional neural network for optical flow estimation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 8981–8989.
[27] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.
[28] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, S. Belongie, Feature pyramid networks for object detection, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2117–2125.
[29] A. Bewley, Z. Ge, L. Ott, F. Ramos, B. Upcroft, Simple online and realtime tracking, in: 2016 IEEE International Conference on Image Processing (ICIP), 2016, pp. 3464–3468.
[30] K. Bernardin, R. Stiefelhagen, Evaluating multiple object tracking performance: the clear MOT metrics, EURASIP J. Image Video Process. 2008 (2008) 1–10.
[31] P. Dollár, R. Appel, S. Belongie, P. Perona, Fast feature pyramids for object detection, IEEE Trans. Pattern Anal. Mach.Intell. 36 (8) (2014) 1532–1545.
[32] P.F. Felzenszwalb, R.B. Girshick, D. McAllester, D. Ramanan, Object detection with discriminatively trained part-based models, IEEE Trans. Pattern Anal. Mach.Intell. 32 (9) (2009) 1627–1645.
[33] F. Yang, W. Choi, Y. Lin, Exploit all the layers: Fast and accurate CNN object detector with scale dependent pooling and cascaded rejection classifiers, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 2129–2137.
[34] A. Geiger, P. Lenz, R. Urtasun, Are we ready for autonomous driving? The KITTI vision benchmark suite, in: 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012, pp. 3354–3361.
[35] E. Ristani, F. Solera, R. Zou, R. Cucchiara, C. Tomasi, Performance measures and a data set for multi-target, multi-camera tracking, in: European Conference on Computer Vision, 2016, pp. 17–35.
[36] Y. Li, C. Huang, R. Nevatia, Learning to associate: hybridboosted multi-target tracker for crowded scene, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 2953–2960.
[37] D.J. Butler, J. Wulff, G.B. Stanley, M.J. Black, A naturalistic open source movie for optical flow evaluation, in: European Conference on Computer Vision, 2012, pp. 611–625.
[38] Y. Xu, A. Osep, Y. Ban, R. Horaud, L. Leal-Taixé, X. Alameda-Pineda, How to train your deep multi-object tracker, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2020, pp. 6787–6796.
[39] C. Ma, C. Yang, F. Yang, Y. Zhuang, Z. Zhang, H. Jia, X. Xie, Trajectory factory: tracklet cleaving and re-connection by deep siamese Bi-GRU for multiple object tracking, in: 2018 IEEE International Conference on Multimedia and Expo (ICME), 2018, pp. 1–6.
[40] L. Chen, H. Ai, Z. Zhuang, C. Shang, Real-time multiple people tracking with deeply learned candidate selection and person re-identification, in: 2018 IEEE International Conference on Multimedia and Expo (ICME), 2018, pp. 1–6.
[41] M. Keuper, S. Tang, B. Andres, T. Brox, B. Schiele, Motion segmentation & multiple object tracking by correlation co-clustering, IEEE Trans. Pattern Anal. Mach.Intell. 42 (1) (2018) 140–153.
[42] Z. Fu, F. Angelini, J. Chambers, S.M. Naqvi, Multi-level cooperative fusion of GM-PHD filters for online multiple human tracking, IEEE Trans. Multimedia 21 (9) (2019) 2277–2291.
[43] Z. Zhou, W. Luo, Q. Wang, J. Xing, W. Hu, Distractor-aware discrimination learning for online multiple object tracking, Pattern Recognit. (2020) 107512.
[44] A. Bewley, Z. Ge, L. Ott, F. Ramos, B. Upcroft, Simple online and realtime tracking, in: IEEE International Conference on Image Processing, 2016, pp. 3464–3468.
[45] Y. Zhang, H. Sheng, Y. Wu, S. Wang, W. Ke, Z. Xiong, Multiplex labeling graph for near-online tracking in crowded scenes, IEEE Internet Things J. 7 (9) (2020) 7892–7902.
[46] W. Ren, X. Wang, J. Tian, Y. Tang, A.B. Chan, Tracking-by-counting: using network flows on crowd density maps for tracking multiple targets, IEEE Trans. Image Process. 30 (2020) 1439–1452.
[47] W. Tian, M. Lauer, L. Chen, Online multi-object tracking using joint domain information in traffic scenarios, IEEE Trans. Intell. Transp. Syst. (2019) 1–11.
[48] W. Zhang, H. Zhou, S. Sun, Z. Wang, J. Shi, C.C. Loy, Robust multi-modality multi-object tracking, in: Proceedings of the IEEE/CVF International Conference on Computer Vision, 2019, pp. 2365–2374.
[49] N.F. Gonzalez, A. Ospina, P. Calvez, SMAT: smart multiple affinity metrics for multiple object tracking, in: International Conference on Image Analysis and Recognition, 2020, pp. 48–62.
[50] J. Pang, L. Qiu, X. Li, H. Chen, Q. Li, T. Darrell, F. Yu, Quasi-dense similarity learning for multiple object tracking, in: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 164–173.

Rui Li received the BS degree and MS degree in computer sciense from Beijing Technology and Business University and Beijing Jiaotong University, respectively, in 2017 and 2019, Beijing, China. Now, she is working towards the PhD degree from the School of Computer and Information Technology, Beijing Jiaotong University. Her research interests are both theory and applications of computer vision.

Baopeng Zhang received his PhD degree in computer science from Tsinghua University in 2008. He is currently an associate professor in the School of Computer and Information Technology, Beijing Jiaotong University, China. His research interests include semantic image/video classification and retrieval, statistical machine learning, large-scale semantic data management and analysis, and image privacy protection.

Zhu Teng received her BS degree in automation from Central South University, Changsha, China, in 2006, and the PhD degree in intelligent control and automation from Pusan National University, South Korea, 2013. From 2013 to 2015, she was a postdoctoral researcher with the school of Computer and Information Technology, Beijing Jiaotong University. From 2015 to 2016, she was an assistant professor at Beijing Jiaotong University. She is now an associate professor in the School of Computer and Information Technology, Beijing Jiaotong University. Her current research interests are computer vision and machine learning

Jianping Fan was a professor at UNC-Charlotte. He received his MS degree in theory physics from Northwestern University, Xian, China in 1994 and his PhD degree in optical storage and computer science from Shanghai Institute of Optics and Fine Mechanics, Chinese Academy of Sciences, Shanghai, China, in 1997. He was a Postdoc Researcher at Fudan University, Shanghai, China, during 1997-1998. From 1998 to 1999, he was a Researcher with Japan Society of Promotion of Science (JSPS), Department of Information System Engineering, Osaka University, Osaka, Japan. From 1999 to 2001, he was a Postdoc Researcher in the Department of Computer Science, Purdue University, West Lafayette, IN. His research interests include image/video privacy protection, automatic image/video understanding, and large-scale deep learning.