UTM: A Unified Multiple Object Tracking Model with Identity-Aware Feature Enhancement

Sisi You[1] Hantao Yao[2] Bing-kun Bao[1] Changsheng Xu[2,3*]

[1]Nanjing University of Posts and Telecommunications

[2]State Key Laboratory of Multimodal Artificial Intelligence Systems, Institute of Automation, Chinese Academy of Sciences (CASIA)

[3]University of Chinese Academy of Sciences

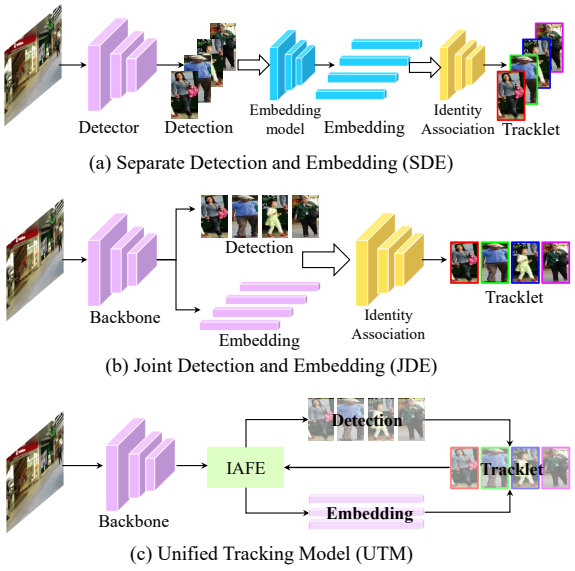
ssyou@njupt.edu.cn, hantao.yao@nlpr.ia.ac.cn, bingkunbao@njupt.edu.cn, csxu@nlpr.ia.ac.cn

Abstract

Recently, Multiple Object Tracking has achieved great success, which consists of object detection, feature embedding, and identity association. Existing methods apply the three-step or two-step paradigm to generate robust trajectories, where identity association is independent of other components. However, the independent identity association results in the identity-aware knowledge contained in the tracklet not be used to boost the detection and embedding modules. To overcome the limitations of existing methods, we introduce a novel Unified Tracking Model (UTM) to bridge those three components for generating a positive feedback loop with mutual benefits. The key insight of UTM is the Identity-Aware Feature Enhancement (IAFE), which is applied to bridge and benefit these three components by utilizing the identity-aware knowledge to boost detection and embedding. Formally, IAFE contains the Identity-Aware Boosting Attention (IABA) and the Identity-Aware Erasing Attention (IAEA), where IABA enhances the consistent regions between the current frame feature and identity-aware knowledge, and IAEA suppresses the distracted regions in the current frame feature. With better detections and embeddings, higher-quality tracklets can also be generated. Extensive experiments of public and private detections on three benchmarks demonstrate the robustness of UTM.

1. Introduction

Multiple Object Tracking (MOT) aims at locating and identifying all of the moving objects in the video, which has broad application prospects in visual surveillance, human-computer interaction, virtual reality, and unmanned vehicles. With the rapid development of object detection [12, 34, 35, 66], *Tracking-By-Detection* has became a favorite paradigm in MOT. Recently, a number of Tracking-By-Detection approaches have been proposed, which can be divided into two paradigms: *Separate Detection and Embedding* (SDE) [1, 2, 4, 5, 8, 10, 16, 25, 38, 43, 52, 54], and *Joint Detection and Embedding* (JDE) [15, 29, 45, 50, 64].



(a) Separate Detection and Embedding (SDE)

(b) Joint Detection and Embedding (JDE)

(c) Unified Tracking Model (UTM)

Figure 1. Comparison of different MOT frameworks in existing methods. (a) Separate Detection and Embedding (SDE) [2]. (b) Joint Detection and Embedding (JDE) [50]. (c) The proposed Unified Tracking Model (UTM) that constructed with Identity-Aware Feature Enhancement (IAFE) module.

As illustrated in Figure 1(a), SDE can be divided into three independent components: object detection, feature embedding, and identity association[1]. Candidate bounding boxes (bboxes) are obtained by standard detectors in each frame first, then identity embedding of each bbox is extracted by the re-identification algorithms, finally linked across frames through identity association to generate tra-

*indicates corresponding author: Changsheng Xu.

[1]In this paper, identity association includes affinity computation and data association.

jectories. Therefore, SDE mainly exploits refining detection [2, 16], enhancing identity embedding [1, 25, 38, 52], or designing robust association algorithms [5, 8, 54] to improve tracking performance. With the maturity of multi-task learning, some approaches [45, 50, 64] propose JDE framework to reduce the computation cost. Different from SDE, JDE applies the one-shot tracker to generate object detections and corresponding visual embeddings simultaneously, thus treating MOT as two independent components, shown in Figure 1(b). Since the identity association in the above two paradigms is independent of object detection and feature embedding, the significant clues contained in the tracklet cannot be applied to enhance the detection and embedding modules.

To address the above problem, a feasible idea is introducing an auxiliary module to associate and propagate identity-aware knowledge between the identity association and the other two components. Therefore, we design a novel Identity-Aware Feature Enhancement (IAFE) module to achieve information interaction between different components, shown in Figure 1(c). In detail, IAFE propagates the identity-aware knowledge generated by identity association module to enhance the backbone feature of object detection. Meanwhile, the enhanced backbone feature can be utilized by the feature embedding module to generate discriminative embeddings. With more accurate detections and robust embeddings, the identity association module can produce higher-quality tracklets. Therefore, object detection, feature embedding, and identity association are involved with each other, thus forming a positive feedback loop with mutual benefits.

As shown in Figure 2, the proposed Unified Tracking Model (UTM) is composed of IAFE, detection branch, embedding branch, identity association branch, and memory aggregation module. The detection and embedding branches are applied to locate and identify each object of the current frame, and the identity association branch applies graph matching to associate the candidate bboxes with history tracklets. To achieve information interaction, IAFE is proposed to bridge and benefit these three branches, which utilizes the identity-aware knowledge to boost the detection and embedding. Specifically, IAFE consists of two modules: Identity-Aware Boosting Attention (IABA) and Identity-Aware Erasing Attention (IAEA), where IABA boosts the consistent regions between the current frame feature and identity-aware knowledge, and IAEA erases the distracted regions in the current frame feature. With the designed IAFE, UTM constructs a positive feedback loop among all the three branches to improve the performance of MOT. Furthermore, we introduce a memory aggregation module to capture identity-aware knowledge through adaptively selecting features of history frames, which can alleviate the effect of identity switches.

The main contributions of the proposed method can be summarized as follows:

- We design an Identity-Aware Feature Enhancement (IAFE) module to bridge and benefit different components in MOT. Specifically, it utilizes the identity-aware knowledge to boost backbone feature, which is further used to enhance the detection and embedding.
- With the proposed IAFE, we construct a **U**nified **T**racking **M**odel (UTM) to form a positive feedback loop with mutual benefits.
- The evaluation of public and private detections on three benchmarks verifies the effectiveness and generalization ability of the proposed UTM.

2. Related Work

MOT has been received more and more attention from industry and academia in the past years. We review the most relevant work of MOT, *i.e.*, Separate Detection and Embedding, Joint Detection and Embedding.

2.1. Separate Detection and Embedding

Separate Detection and Embedding(SDE) is the most popular framework for MOT, which consists of three independent components: object detection, feature embedding, and identity association. Existing SDE methods exploit refining detection [2, 5, 15, 16, 59] or the provided public detections [11–13, 35] for MOT. Then, the appearance embedding model [1, 2, 25, 38, 52, 62] is used to extract discriminative feature for each candidate bbox, *e.g.,* Bae *et al.* [1] boost the representation learning by training the visual model on person re-identification datasets. Besides the appearance model, the motion model [10, 15, 39, 52, 58] is also applied to describe the motion information of each object, *e.g.,* the Kalman filter [22] is widely used in MOT. Finally, the matching algorithm is used to solve the problem of identity association, *e.g.,* Hungarian algorithm [24], multicut [19, 20], min-cost max-flow network [61], and conditional random field [55]. Except for the traditional matching algorithms, more and more methods [5, 8, 16, 36, 49, 51, 54] apply deep learning for identity association. Xu *et al.* [54] propose a Deep Hungarian Network module to substitute the Hungarian algorithm. Some approaches [5, 8, 16, 49, 51] formulate identity association as a graph optimization problem. The work most related to ours is MPNTrack [5], which treats each object as a graph node and applies the edge feature for classification. The major difference is that we construct the detection graph and tracklet graph for identity association according to high-order context information, which brings the better performance of our model.

2.2. Joint Detection and Embedding

To construct a real-time tracker, Joint Detection and Embedding (JDE) methods have begun to attract more atten-

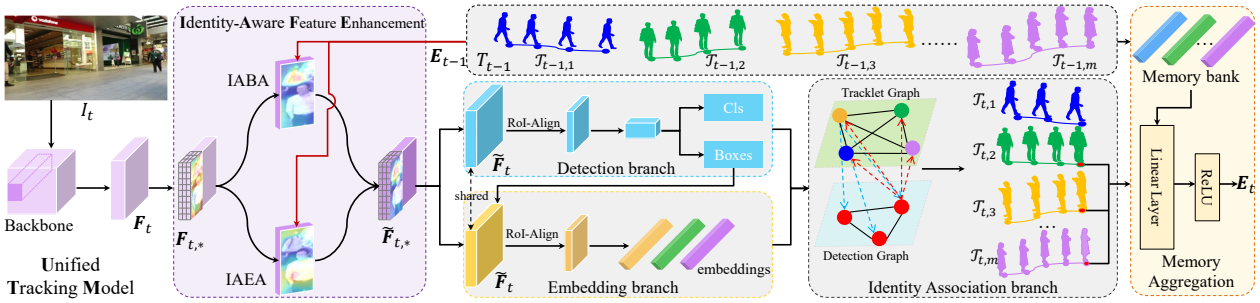


Figure 2. The overall framework of the proposed Unified Tracking Model (UTM). UTM is composed of identity-aware feature enhancement module, detection branch, embedding branch, identity association branch, and memory aggregation module. Specifically, UTM leverages the identity-aware knowledge to enhance detection and embedding, which in turn benefits identity association.

tion. JDE methods allow object detection and feature embedding to be learned in a shared model, which can reduce the computation cost. Specifically, JDE methods [29, 45, 50, 64] add an embedding head on top of different detectors to output bboxes and corresponding appearance embeddings simultaneously, *e.g.,* Track-RCNN [45] is designed on top of Mask-RCNN [17], JDE [50] is introduced on YOLOv3 [34] framework for real-time MOT, FairMOT [64] proposes a simple baseline on CenterNet [67]. Furthermore, Guo *et al.* [15] make position prediction and embedding association benefit each other based on the JDE framework. Specifically, they apply the target feature and distractor feature to generate the target attention and distractor attention for reliable embeddings, which can reduce the occlusion by pedestrians but cannot eliminate the background information. To eliminate the background information, we propose an identity-aware erasing attention to suppress the distracted regions in the current frame.

Besides the JDE methods, many other methods [6, 7, 32, 33, 46, 53, 57, 68] have been proposed. Pang *et al.* [32] introduce the bounding-tube to combine detection and association in a short video clip, and utilize the IoU-based greedy algorithm to link different bounding-tubes. Specifically, a bounding-tube can be treated as the combination of three bounding-boxes, *e.g.,* start box, middle box, and end box, from different video frames. Peng *et al.* [33] take adjacent frame pairs as input and regress the paired bboxes for the targets that appear in both adjacent frames, and also apply IoU matching between different adjacent pairs. Moreover, some online solutions [6, 7, 57, 68] apply Single Object Trackers (SOT) for MOT, *e.g.,* Yin *et al.* [57] simultaneously learn the SOT based object motion prediction and affinity-dependent ranking model.

However, all of these methods are multi-step methods, *e.g.*, SDE methods conduct detection, embedding and identity association separately, and JDE methods jointly learn object detection and feature embedding, where identity association is still independent with others. Since information cannot be propagated between independent modules, the valuable identity-aware knowledge cannot be utilized to enhance detection and embedding modules in the multi-step methods. To address these problems, we introduce the identity-aware feature enhancement module to form a unified tracking model, which can bridge and benefit the above three components to form a positive feedback loop.

3. Methodology

As shown in Figure 2, UTM is composed of the Identity-Aware Feature Enhancement (IAFE) module, detection branch, embedding branch, identity association branch, and memory aggregation module. Given the t-th frame, we first apply the backbone network to obtain the backbone feature $\mathbf{F_t}$. Then, IAFE takes the efficient tracklet features as the identity-aware knowledge to enhance the backbone feature of each tracked object, where tracklet feature set and enhanced backbone feature are denoted as $\mathbf{E_{t-1}}$ and $\tilde{\mathbf{F}}_t$, respectively. Next, the detection and embedding branches utilize $\tilde{\mathbf{F}}_t$ to obtain candidate bbox set B_t and object embedding set $\hat{\mathbf{F}}_t$. Then, the identity association branch utilizes the high-order contextual information to associate the candidate bboxes and history tracklets. Finally, the memory aggregation module is used to update the tracklet feature that is applied to enhance the backbone feature with IAFE in the next frame. In the following, we will describe the detail of each module.

3.1. Identity-Aware Feature Enhancement

To construct a Unified Tracking Model, we propose an Identity-Aware Feature Enhancement (IAFE) module to bridge the detection, embedding, and identity association branches to generate a positive feedback loop. Specifically, IAFE leverages the efficient tracklet features to boost the detection and embedding branches, thereby generating high-quality tracklets with the identity association branch. As shown in Figure 3, IAFE consists of the Identity-Aware Boosting Attention (IABA) and Identity-Aware Erasing Attention (IAEA), where IABA is utilized to boost the features of consistent regions between history tracklets and current frame, and IAEA is applied to suppress the features of dis-

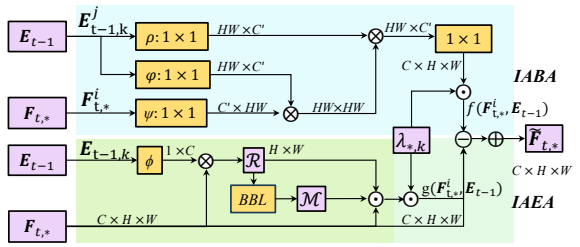


Figure 3. Illustration of the proposed IAFE module. BBL indicates the block binarization layer, $\mathcal{R}$ and $\mathcal{M}$ represent the correlation feature and binary mask.

tracted regions in the current frame.

Following Tracktor [2], we combine the corresponding public detections and tracked objects in the previous frame as the candidate proposals P_t of the current frame I_t. After that, IAFE applies the tracklet features $\mathbf{E}_{t-1}$ to enhance the backbone features of candidate proposals, where $\mathbf{E}_{t-1} = \{\mathbf{E}_{t-1,1}, ..., \mathbf{E}_{t-1,m}\}$, m is the number of history tracklets. For example, given the candidate proposal $P_{t,*} \in P_t$ along with the backbone feature $\mathbf{F}_{t,*}$, we can leverage $\mathbf{E}_{t-1}$ to generate the enhanced feature $\tilde{\mathbf{F}}_{t,*}$ as follows:

$$\tilde{\mathbf{F}}_{t,*} = \mathbf{F}_{t,*} \oplus [f(\mathbf{F}_{t,*}, \mathbf{E}_{t-1}) \ominus g(\mathbf{F}_{t,*}, \mathbf{E}_{t-1})], \quad (1)$$

where $\oplus$ and $\ominus$ indicate element-wise addition and subtraction, $f(\mathbf{F}_{t,*}, \mathbf{E}_{t-1})$ and $g(\mathbf{F}_{t,*}, \mathbf{E}_{t-1})$ represent IABA and IAEA, respectively. In the following, we give a detailed description about the boosting attention module $f(\cdot)$ and erasing attention module $g(\cdot)$.

The boosting attention module $f(\cdot)$ leverages $\mathbf{E}_{t-1}$ to enhance the consistent feature between $\mathbf{E}_{t-1}$ and $\mathbf{F}_{t,*}$. Given the proposal feature $\mathbf{F}_{t,*} \in \mathbb{R}^{C\times H\times W}$ and $\mathbf{E}_{t-1}$, $f(\mathbf{F}_{t,*}, \mathbf{E}_{t-1})$ can be formulated as follows:

$$f(\mathbf{F}^i_{t,*}, \mathbf{E}_{t-1}) = \sum_{k=1}^{m} \frac{\lambda_{*,k} \sum_{\forall j} h(\mathbf{F}^i_{t,*}, \mathbf{E}^j_{t-1,k}) \rho(\mathbf{E}^j_{t-1,k})}{\sum_{\forall j} h(\mathbf{F}^i_{t,*}, \mathbf{E}^j_{t-1,k})}, \quad (2)$$

where $\mathbf{E}_{t-1,k} \in \mathbf{E}_{t-1}$ is the tracklet feature of the k-th tracklet $\mathcal{T}_{t-1,k}$, and $\mathbf{F}^i_{t,*}, \mathbf{E}^j_{t-1,k} \in \mathbb{R}^C (i, j \in [1, HW])$ are the features sampled from $\mathbf{F}_{t,*}$ and $\mathbf{E}_{t-1,k}$. $h(\mathbf{x}_i, \mathbf{x}_j) = e^{\psi(\mathbf{x}_i)^T \varphi(\mathbf{x}_j)}$, where ψ, φ, ρ are convolution layers. $\lambda_{*,k}$ is an indicator and is defined as follows:

$$\lambda_{*,k} = \begin{cases} 1 & IoU(P_{t,*}, B_{t-1,k}) > \lambda_{iou}, \\ 0 & otherwise, \end{cases} \quad (3)$$

where $IoU(P_{t,*}, B_{t-1,k})$ is the geometric similarity between the candidate proposal $P_{t,*}$ and the last bbox $B_{t-1,k}$ of $\mathcal{T}_{t-1,k}$, and λ_{iou} is the geometric threshold.

To eliminate the background information, we apply the tracklet features to erase distracted regions in the backbone feature $\mathbf{F}_{t,*}$, which can be formulated as follows:

$$g(\mathbf{F}_{t,*}, \mathbf{E}_{t-1}) = \sum_{k=1}^{m} \lambda_{*,k} \mathbf{F}_{t,*} \odot \mathcal{R}(\mathbf{F}_{t,*}, \phi(\mathbf{E}_{t-1,k})) \odot \mathcal{M}(\mathbf{F}_{t,*}, \phi(\mathbf{E}_{t-1,k})), \quad (4)$$

where ϕ indicates the average pooling layer, $\odot$ is element-wise product operation, $\mathcal{R}(\mathbf{x}_i, \mathbf{x}_j)$ and $\mathcal{M}(\mathbf{x}_i, \mathbf{x}_j)$ represent the correlation feature and binary mask, respectively. The correlation feature denotes the dot-produce similarity between tracklet feature and all the local features in $\mathbf{F}_{t,*}$.

$$\mathcal{R}(\mathbf{F}_{t,*}[i,j], \phi(\mathbf{E}_{t-1,k})) = (\mathbf{F}_{t,*}[i,j])^T \phi(\mathbf{E}_{t-1,k}), \quad (5)$$

where $\mathbf{F}_{t,*}[i,j]$ denotes the local feature of spatial location (i,j) of $\mathbf{F}_{t,*}$.

After obtaining the correlation feature, we apply $\mathcal{R}(\cdot)$ to generate the binary mask $\mathcal{M}(\cdot)$ between backbone feature $\mathbf{F}_{t,*}$ and tracklet feature $\mathbf{E}_{t-1,k}$. Inspired by [21], we apply a block binarization layer to generate the mask for erasing a contiguous region in $\mathbf{F}_{t,*}$. We exploit a sliding block with the size of 3×3 and strides $= 1$ to search the most highlighted continuous area in the correlation feature, and define the correlation value of each block as the sum of the values in the block. Then, we select the candidate block with the highest correlation value as the block to be reversed, *i.e.,* the binary mask of correlation feature is obtained by setting the values of the selected block to 0 and others to 1. Next, we apply a softmax layer to generate a gate map for end-to-end training, where gate map is obtained by the element-wise produce operation between softmax-based correlation feature and binary mask. Finally, the erasing attention can be obtained based on the gate map and binary mask.

3.2. Unified Tracking Model

The proposed IAFE in Sec. 3.1 utilizes the identity-aware knowledge to achieve information interaction between different components in MOT, thus constructs a Unified Tracking Model (UTM) to form a positive feedback loop. Except for IAFE, UTM also contains the detection, embedding, identity association branches, and memory aggregation module, which will be described in the following.

Detection Branch. Inspired by Tracktor [2], we adopt Faster R-CNN [35] as the detection branch, where the regression head and classification head are utilized to refine bboxes and infer classes of objects inside boxes. Different from the candidate proposals generated by RPN in Faster R-CNN, we combine the public detections in the current frame and the tracked objects in the previous frame as the candidate proposals. Then the regression head is exploited to predict candidate bboxes $B_t = \{B_{t,1}, ...B_{t,n}\}$ based on the enhanced backbone feature $\tilde{\mathbf{F}}_t$ which obtained by IAFE, and the classification head gives the confidence score for

each bbox. Note that $B_{t,i}$ indicates a bbox in B_t and n is the number of candidate bboxes. It is worth noting that we merely apply the regression head to refine the public detections and tracked objects, while not generate new candidate bboxes. Simultaneously, the regression head is trained with L1 loss on displacements, and the classification head is learned with a cross-entropy loss.

Embedding Branch. After obtaining the candidate bboxes B_t, the embedding branch targets to generate their discriminative embeddings. Given the enhanced backbone feature $\tilde{\mathbf{F}}_t$ and a bbox $B_{t,i}$, the corresponding embedding $\hat{\mathbf{F}}_{t,i}$ can be formulated as follows:

$$\hat{\mathbf{F}}_{t,i} = \mathcal{E}(\tilde{\mathbf{F}}_t, B_{t,i}), \tag{6}$$

where $\mathcal{E}$ represents convolution layer on top of the ROI-Align layer. Then, the embeddings of B_t can be denoted as $\hat{\mathbf{F}}_t = \{\hat{\mathbf{F}}_{t,1}, ..., \hat{\mathbf{F}}_{t,i}, ..., \hat{\mathbf{F}}_{t,n}\}$. Furthermore, the embedding branch is trained with the combination of cross-entropy loss and triplet loss to learn the discriminative identity embeddings.

Identity Association Branch. In this paper, we formulate identity association as a graph matching problem between the candidate bboxes B_t and history tracklets T_{t-1}, where $B_t = \{B_{t,1}, ..., B_{t,n}\}$, $T_{t-1} = \{\mathcal{T}_{t-1,1}, ..., \mathcal{T}_{t-1,m}\}$, n and m represent the number of candidate bboxes and history tracklets, respectively. We first construct the detection graph and tracklet graph to describe the relationships of different objects in the candidate bboxes and history tracklets, respectively. Then, the cross-graph message passing between the detection graph and tracklet graph is applied to enhance the node features. Finally, the graph matching layer applies high-order context information to associate the detection graph with the tracklet graph.

The detection graph is defined as $\mathcal{G}_D = (V_D, E_D)$. $V_D = \{(B_{t,i}; \hat{\mathbf{F}}_{t,i})\}(i \in [1, n])$ represents the node set, where $\hat{\mathbf{F}}_{t,i}$ is the embedding of the i-th bbox $B_{t,i}$ for the t-th frame. $E_D = \{[\hat{\mathbf{F}}_{t,i_1}, \hat{\mathbf{F}}_{t,i_2}]\}$ indicates the edge set, where $[\cdot]$ indicates the concatenation operation. Similarly, the tracklet graph is defined as $\mathcal{G}_T = (V_T, E_T)$. $V_T = \{(B_{t-1,j}; \mathbf{E}_{t-1,j})\}$, $E_T = \{[\mathbf{E}_{t-1,j_1}, \mathbf{E}_{t-1,j_2}]\}(j, j_1, j_2 \in [1, m])$, where $B_{t-1,j}$ and $\mathbf{E}_{t-1,j}$ indicate the last bbox and tracklet feature of the j-th history tracklet $\mathcal{T}_{t-1,j}$.

To model the feature interaction between the detection graph $\mathcal{G}_D$ and the tracklet graph $\mathcal{G}_T$, we adopt the cross-graph message passing to propagate the information across these two graphs. Let $\hat{\mathbf{F}}^{(0)}_{t,i} = \hat{\mathbf{F}}_{t,i}$ and $\mathbf{E}^{(0)}_{t-1,j} = \mathbf{E}_{t-1,j}$ be the initial feature of each node in V_D and V_T, we analyze the effect of three node aggregation rules.

$$\begin{aligned}
&\text{(Type1)} \quad \hat{\mathbf{F}}^{(l+1)}_{t,i} = \hat{\mathbf{F}}^{(0)}_{t,i}, \\
&\text{(Type2)} \quad \hat{\mathbf{F}}^{(l+1)}_{t,i} = \mathcal{N}_v(\hat{\mathbf{F}}^{(l)}_{t,i} + \frac{1}{m}\textstyle\sum_{j=1}^{m} \mathbf{E}^{(l)}_{t-1,j}), \\
&\text{(Type3)} \quad \hat{\mathbf{F}}^{(l+1)}_{t,i} = \mathcal{N}_v(\hat{\mathbf{F}}^{(l)}_{t,i} + \textstyle\sum_{j=1}^{m} w^{(l)} \mathbf{E}^{(l)}_{t-1,j}),
\end{aligned} \tag{7}$$

where $\hat{\mathbf{F}}^{(l)}_{t,i}$ and $\mathbf{E}^{(l)}_{t-1,j}$ are the features of the l-th propagation, $w^{(l)} = cos(\hat{\mathbf{F}}^{(l)}_{t,i}, \mathbf{E}^{(l)}_{t-1,j}) + IoU(B_{t,i}, B_{t-1,j})$ is the combination of cosine similarity and geometric similarity obtained by the Intersection over Union (IoU) of two bboxes, and $\mathcal{N}_v$ represents learnable function, *e.g.,* MLP. The final feature of the node in V_D can be denoted as $\hat{\mathbf{F}}_{t,i} = \hat{\mathbf{F}}^{(L)}_{t,i}$, where L is the total steps of the message passing. Meanwhile, the similar operation is also applied to the tracklet graph $\mathcal{G}_T$.

After that, we utilize the first-order node-to-node similarity and the second-order edge-to-edge similarity to compute the affinity matrix $\mathbf{M}$, where both of the similarities are described with cosine similarity. Furthermore, two nodes between the detection graph and tracklet graph are matched if the corresponding similarity in $\mathbf{M}$ is higher than an affinity threshold γ. Finally, the optimal matching $\mathbf{Y}^*$ can be obtained with:

$$\mathbf{Y}^* = \arg\max_{\mathbf{Y}} \mathbf{Y}^T \mathbf{M} \mathbf{Y}, \tag{8}$$

where $\mathbf{Y} \in \{0, 1\}^{n \times m}$ is a permutation matrix that denotes the matching result between the detection and tracklet graphs. The more details of the optimization of $\mathbf{Y}^*$ can be referred to [60]. Moreover, the identity association branch is trained with weighted binary cross-entropy loss.

Memory Aggregation. The identity-aware feature enhancement module is described in Sec. 3.1, which applies the efficient tracklet features to enhance the detection and embedding branches. Although storing identity embeddings formed in the previous frames is an intuitive and accessible way to obtain tracklet feature, the identity embeddings may noisy due to occlusion and identity switches. To enhance the robustness of tracklet feature, we introduce a memory bank to aggregate effective tracklet feature for each object, so that all of the object detection, feature embedding, and identity association can be further boosted. Furthermore, the memory bank groups cached memories in different memory units for different objects, each memory unit can be denoted as $\mathbf{F}^m_{t-1,j} = \{\hat{\mathbf{F}}_{t-\eta,j}, ..., \hat{\mathbf{F}}_{t-1,j}\}$, where $\mathbf{F}^m_{t-1,j}$ and $\hat{\mathbf{F}}_{t-1,j}$ denote the memory and identity embedding of the j-th object in the (t-1)-th frame, and η is the memory length. Specifically, we adopt a learnable memory aggregation module to adaptively select valid identity embeddings, which consists of a linear layer with a ReLU activation function. Each tracklet feature $\mathbf{E}_{t,j}$ can be updated as follows:

$$\mathbf{E}_{t,j} = \theta(\mathbf{F}^m_{t-1,j}, \hat{\mathbf{F}}_{t,j}), \tag{9}$$

where θ represents the memory aggregation operation.

Optimization. To achieve faster convergence, we adopt a multi-step optimization strategy. Firstly, we jointly train the detection branch, embedding branch, and memory aggregation module. Specifically, the detection branch is

constrained with L1 loss and cross-entropy loss, the embedding branch and memory aggregation module are constrained with two discriminative identity losses that consists of cross-entropy loss and triplet loss. Secondly, we utilize the output of detection branch and embedding branch to train the identity association branch while the detection and embedding branches are fixed. Finally, we jointly fine-tune all the components in the full unified model.

4. Experiments

Datasets and evaluation metrics. We conduct all the experiments on three MOT benchmarks, *e.g.*, MOT16 [31], MOT17 [31], and MOT20 [9]. Following the CLEAR MOT Metrics [3], IDF1 Score [37], and HOTA [30], we apply some basic items for quantitative evaluation, *e.g.,* Multiple Object Tracking Accuracy (MOTA ↑), IDF1(↑), Higher Order Tracking Accuracy (HOTA ↑), Mostly Tracked (MT ↑), Mostly Lost (ML ↓), False Positives (FP ↓), False Negatives (FN ↓), and Identity Switches (IDS ↓).

Implementation details. The proposed method is implemented by Pytorch with RTX 3090. We adopt Faster R-CNN [35] with Feature Pyramid Network (FPN) [27] as the detection branch. For public detection, we pre-train the backbone of ResNet101 [18] on COCO dataset [28]. For private detection, we pre-train the detection branch on CrowdHuman [40] and MOT training datasets. Simultaneously, the embedding branch is pre-trained on Market1501 [65] and CUHK03 [26] datasets. Then we refine the whole model with the MOT training datasets. The initial learning rate is set to 0.002 with a decay factor 0.5 at every 3 epochs up to 30 epochs. Adam optimizer [23] is used with a mini-batch size of 2. We set the number of message passing steps $L = 3$, geometric threshold $\lambda_{iou} = 0.7$, affinity threshold $\gamma = 0.6$, and the memory length $\eta = 30$.

4.1. Benchmark Evaluation

We compare the proposed Unified Tracking Model (UTM) with several methods on three benchmarks, *e.g.,* MOT16, MOT17, and MOT20. The benchmark evaluation can be divided into public detection and private detection.

Public Detection. We compare the proposed method with traditional tracking-by-detection methods that apply Tracktor for refining detections on the public detection setting for a fair comparison, and the comparison with other refined detectors are provided in supplementary material. Furthermore, the compared methods can be categorized into online and offline tracking methods. As shown in Table 1, the proposed method achieves better performance than existing online methods on most of the evaluation metrics. Moreover, we compare UTM with two multi-step frameworks to demonstrate its superiority. Compared with the Separate Detection and Embedding (SDE) method Tracktor [2], UTM obtains a higher HOTA, *e.g.,* 8.5%, 7.7%, and 11.2% improvements on MOT16, MOT17, and MOT20. Meanwhile, we compare UTM with the Joint Detection and Embedding (JDE) method TADAM [15], the proposed UTM obtains a higher IDF1, *e.g.,* 7.6%, 6.4%, and 14.3% improvements on MOT16, MOT17, and MOT20. Among existing offline methods, the most related work to ours is GMTsI [16], which utilizes a similar graph matching method for identity association. The major difference and novelty is that UTM leverages the identity-aware knowledge to enhance the object detection and feature embedding modules. Compared with GMTsI [16], UTM achieves 4.5% improvement of the MOTA metric on MOT17 dataset. We attribute the performance improvement to the proposed UTM generates a positive feedback loop with identity-aware feature enhancement module.

Methods	Refined	MOTA	HOTA	IDF1	FP	FN	IDS
MOT16							
MPNT(O) [5]	Tracktor	58.6	48.9	61.7	4,949	70,252	354
LPC(O) [8]	Tracktor	58.8	51.7	**67.6**	6,167	68,432	435
GMTsI(O) [16]	Tracktor	61.1	51.2	66.6	3,891	66,550	503
DeepMOT [54]	Tracktor	54.8	42.2	53.4	2,955	78,765	645
GMT [16]	Tracktor	55.9	48.9	63.9	**2,371**	77,545	531
Tracktor [2]	Tracktor	56.2	44.6	54.9	2,394	76,844	617
ArTIST [39]	Tracktor	56.6	-	57.8	3,532	75,031	519
LifTsI [19]	Tracktor	57.5	49.6	64.1	4,249	72,868	**335**
TADAM [15]	Tracktor	59.1	-	59.5	2,540	71,542	529
TMOH* [41]	Tracktor	63.2	50.7	63.5	3,122	63,376	635
UTM	Tracktor	**63.8**	**53.1**	67.1	8,328	**57,269**	428
MOT17							
LifTsI(O) [19]	Tracktor	58.2	50.7	65.2	16,850	217,944	**1,022**
MPNT(O) [5]	Tracktor	58.8	49.0	61.7	17,413	213,594	1,185
LPC(O) [8]	Tracktor	59.0	51.7	**66.8**	23,102	206,947	1,122
GMTsI(O) [16]	Tracktor	59.0	51.1	65.9	20,395	209,553	1,105
GMT [16]	Tracktor	56.2	49.1	63.8	**8,719**	236,541	1,778
Tracktor [2]	Tracktor	56.3	44.8	55.1	8,866	235,449	1,987
ArTIST [39]	Tracktor	56.7	-	57.5	12,353	230,437	1,756
TADAM [15]	Tracktor	59.7	-	58.7	9,676	216,029	1,930
TMOH* [41]	Tracktor	62.1	50.4	62.8	10,951	201,195	1,897
UTM	Tracktor	**63.5**	**52.5**	65.1	33,683	**170,352**	1,686
MOT20							
LPC(O) [8]	Tracktor	56.3	49.0	62.5	11,726	213,056	1,562
MPNT(O) [5]	Tracktor	57.6	46.8	59.1	16,953	210,384	**1,210**
Tracktor [2]	Tracktor	52.6	42.1	52.7	**6,930**	236,680	1,648
ArTIST [39]	Tracktor	53.6	-	51.0	7,765	230,576	1,531
TADAM [15]	Tracktor	56.6	-	51.6	38,407	182,520	2,690
TMOH* [41]	Tracktor	60.1	48.9	61.2	38,043	165,899	2,342
UTM	Tracktor	**64.4**	**53.3**	**65.9**	82,726	**98,974**	2,592

Table 1. Comparison with modern methods on MOT16, MOT17, and MOT20 benchmarks with the provided public detections. Best results are marked in **BLOD**. "O" and * indicate the offline methods and post processing methods.

Private Detection. To further verify the effectiveness of the proposed UTM, we compare UTM with some algorithms on private detection setting and summarize the re-

Methods	Detector	MOTA	HOTA	IDF1	FP	FN	IDS
MOT16							
FairMOT [64]	CeneterNet	75.7	61.6	75.3	16,163	27,442	621
GRTU [48]	CeneterNet	76.5	62.6	75.9	11,438	30,866	584
TLR [47]	CeneterNet	76.6	61.0	74.3	**10,860**	30,756	979
UTM	FRCNN	**81.1**	**64.1**	**79.0**	11,722	**22,367**	**440**
MOT17							
FairMOT [64]	CeneterNet	73.7	59.3	72.3	27,507	117,477	3,303
PermaTrack [44]	CeneterNet	73.8	55.5	68.9	28,998	115,104	3,699
GRTU [48]	CeneterNet	74.9	62.0	75.0	32,007	107,616	1,812
TLR [47]	CeneterNet	76.5	60.7	73.6	29,808	99,510	3,369
MAA [42]	FRCNN	79.4	62.0	75.9	37,320	77,661	1,452
ByteTrack [63]	YOLOX	80.3	63.1	77.3	25,491	83,721	2,196
UTM	FRCNN	**81.8**	**64.0**	**78.7**	**25,077**	**76,298**	**1,431**
MOT20							
FairMOT [64]	CeneterNet	61.8	54.6	67.3	103,440	88,901	5,243
MAA [42]	FRCNN	73.9	57.3	71.2	**24,942**	108,744	1,331
ReMOT [56]	CeneterNet	77.4	61.2	73.1	28,351	86,659	1,789
ByteTrack [63]	YOLOX	77.8	61.3	75.2	26,249	87,594	**1,223**
UTM	FRCNN	**78.2**	**62.5**	**76.9**	29,964	**81,516**	1,228

Table 2. Comparison on private detection setting of MOTChallenge benchmarks. The best results are marked in **bold**.

lated results in Table 2. The compared methods based on three different detectors, *e.g.,* Faster R-CNN [35], CenterNet [66], and YOLOX [14]. As shown in Table 2, UTM achieves the better performance than existing methods on most of the evaluation metrics. In terms of the most important evaluation metric MOTA, the proposed method obtains an obvious improvement upon the current state-of-the-art performance, *e.g.,* 4.5% improvement on MOT16 dataset. Among the compared method, the most related work to ours is FairMOT [64], which designs a joint detection and embedding network based on CenterNet. The major difference and novelty of ours is that we generate a positive feedback loop with identity-aware feature enhancement module in UTM. Compared with FairMOT, the proposed method obtains an obvious improvement, *e.g.,* 16.4% improvements on MOTA of MOT20 dataset. Compared with MAA [42], which applies the similar detector Faster R-CNN, UTM achieves a noticeable improvements on MOT17 and MOT20 datasets. We attribute the performance improvement to that the proposed UTM leverages identity-aware knowledge to enhance the object detection and feature embedding modules.

4.2. Ablation Study

To prove the effectiveness of the proposed components in UTM, we conduct some ablation studies on the MOT16 validation dataset following [5].

Effect of UTM: To verify the benefits of the proposed Unified Tracking Model (UTM), we conduct several comparisons between UTM and existing multi-step paradigms, *e.g.,* SDE and JDE. To make a fair comparison, we take the SDE method Tracktor [2] as the baseline for these three frameworks. As shown in Table 3, UTM (c) obtains a significant improvement compared with SDE (a) and JDE (b),

	Detector	Embedding	Association	MOTA	IDF1	HOTA
(a)	Tracktor	ResNet101	Hungarian	62.5	67.4	59.4
(b)	JDE		Hungarian	62.6	64.0	58.5
(c)	UTM			**64.5**	**73.1**	**63.8**

Table 3. Effect of different paradigms for MOT.

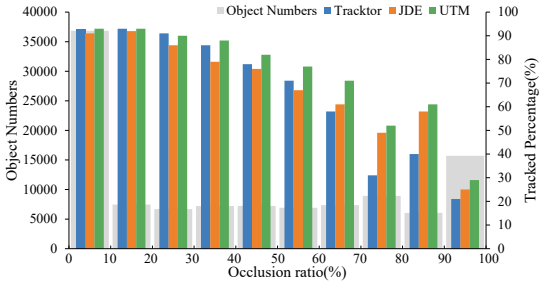
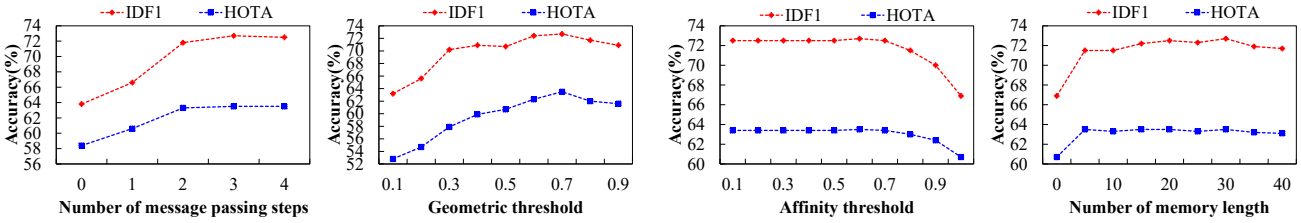


Figure 4. Illustration of the relation between the tracked objects number and occlusion ratio. Wider gray bars show the occurrence of ground truth object bboxes in each occlusion level interval, while narrower colored bars illustrate the percentage of objects tracked for their respective method. Note that occurrences and tracked percentages are not drawn in the same unit.

e.g., 2.0% and 1.9% improvements on MOTA. We attribute the performance gain to UTM which can generate a positive feedback loop with mutual benefits. To compare the effectiveness of different methods on occlusion, we analyze the ratio of successfully tracked objects with respect to their occlusion ratio. The occlusion ratio is defined as the ratio between the occluded area divided by its bbox area, and the higher object occlusion ratio denotes the heavier occlusion. From Figure 4, it can be observed that the proposed UTM performs sufficiently well on most settings. Specifically, the proposed method obtains a higher performance for the severely occluded object, *e.g.,* the object is occluded than 50%. The reason is that UTM can leverage identity-aware knowledge to enhance the description of the occluded objects.

Effect of each component in UTM: We evaluate the effectiveness of each component in the proposed UTM by removing it from UTM in Table 4, *i.e.,* IAFE, IABA, IAEA, and memory aggregation module. Without considering IAFE, the MOTA performance is dropped from 64.5% to 62.6%. The performance degradation is caused by the increasing of FN without IAFE. Meanwhile, higher IDF1 and HOTA indicate that UTM performs better in distinguishing identities with IAFE. Compared with the model w/o IAFE, introducing IABA without IAEA improves MOTA by 1.0%, while applying IAEA alone leads to 1.3% higher MOTA. The higher performance indicates that IABA and IAEA can improve tracking performance on their own, while putting them together obtains a higher improvement. Moreover, we also conduct an experiment to show the benefit of the memory aggregation module (w/o Memory). Instead of



(a)Accuracy *v.s.* munber of steps (b)Accuracy *v.s.* geometric threshold (c)Accuracy *v.s.* affinity threshold (d)Accuracy *v.s.* number of memory length

Figure 5. Effect of the message passing steps L, the geometric threshold λ_{iou}, the affinity threshold γ, and the memory length η.

Methods	MOTA	IDF1	HOTA	MT	ML	FP	FN	IDS
w/o IAFE	62.6	65.9	59.8	172	126	**629**	40,262	366
w/o IABA	63.9	70.6	62.2	**188**	123	680	38,565	518
w/o IAEA	63.6	67.7	61.5	188	128	746	39,170	**237**
w/o Memory	64.2	72.7	63.5	184	118	787	38,434	370
UTM	**64.5**	**73.1**	**63.8**	186	**117**	878	**38,002**	285

Table 4. Effect of each component in the proposed UTM.

Methods	MOTA	IDF1	HOTA	MT	ML	FP	FN	IDS
Hungarian	63.7	68.0	61.3	**187**	123	**689**	39,246	**184**
GM	**64.5**	**73.1**	**63.8**	186	**117**	878	**38,002**	285

Table 5. Effect of different matching algorithms.

using memory aggregation module, we simply utilize average pooling layer to generate the tracklet feature as the input of IAFE. Without the memory aggregation module, we observe the decrease on MOTA, IDF1, and HOTA. This indicates that the discriminative memory aggregation exactly obtain more robust identity-aware knowledge to benefit IAFE, so that further boost the detection and embedding.

Effect of graph matching: To verify the effectiveness of the graph matching layer, we compare it with the Hungarian algorithm [24]. As shown in Table 5, compared with the Hungarian algorithm, graph matching (GM) obtains the improvement of 5.1% in term of IDF1. The reason is that the Hungarian algorithm ignores the second-order edge-to-edge similarity that can model the group activity to generate more reliable tracklets. Furthermore, the obvious improvement on IDF1 demonstrates the robust association of the graph matching with high-order context information.

Effect of node aggregation rules: We further analyze the aggregation rules used for feature interaction. As shown in Table 6, the node aggregation rule "Type 3" obtains the best performance. The reason is that the network can focus on the node with a high affinity score between the detection graph and tracklet graph.

Effect of message passing steps L: As shown in Figure 5(a), we observe a clear upward tendency for both IDF1 and HOTA from 0 to 3 steps, and then they tend to be flat after $L = 3$. Hence, we use $L = 3$ in this work.

Effect of geometric threshold λ_{iou}: We summarize the results for the effect of the geometric threshold λ_{iou} in Figure 5(b), it can be observed that IDF1 and HOTA metrics increase significantly from 0.1 to 0.7, and then decrease after $\lambda_{iou} = 0.7$. Thus, we set $\lambda_{iou} = 0.7$ in IAFE.

Agg.	MOTA	IDF1	HOTA	MT	ML	FP	FN	IDS
Type 1	63.4	63.8	58.4	181	126	746	39,257	496
Type 2	63.6	66.7	60.3	185	124	**677**	39,345	**203**
Type 3	**64.5**	**73.1**	**63.8**	**186**	**117**	878	**38,002**	285

Table 6. Effect of different types of node aggregation rules.

Effect of affinity threshold γ: We also analyze the effect of the affinity threshold γ in graph matching and the related results are summarized in Figure 5(c), it can be observed that HOTA and IDF1 metrics first increase for $\gamma \in [0.1, 0.6]$ and then slowly decrease for $\gamma > 0.6$. Thus, the proposed method works best when $\gamma = 0.6$.

Effect of memory length η: We conduct an experiment to show the effect of memory length. As illustrated in Figure 5(d), the HOTA and IDF1 metrics reach the best performance for $\eta = 30$.

5. Conclusion

In this work, we propose a Unified Tracking Model (UTM) to generate a positive feedback loop with multi benefits, which introduces the Identity-Aware Feature Enhancement (IAFE) module to bridge and benefit object detection, feature embedding, and identity association. IAFE leverages the identity-aware knowledge to enhance the detection and embedding modules, thereby generating reliable tracklets by identity association. Specifically, IAFE consists of identity-aware boosting attention and identity-aware erasing attention, where the boosting attention and erasing attention are utilized to enhance and suppress regions of the current frame feature. The proposed method achieves the best performances on three benchmarks, which illustrates the effectiveness of UTM. In the future, we will optimize the embedding branch to reduce the influence caused by the small batch size for the end-to-end training.

Acknowledgements. This work was supported by National Key Research and Development Project (2020AAA0106200), National Natural Science Foundation of China (61936005, 62036012, 61721004, U21B2044), Beijing Natural Science Foundation (L201001,4222039), and Natural Science Research Start-up Foundation of Recruiting Talents of Nanjing University of Posts and Telecommunications (NY222116).

References

[1] Seung-Hwan Bae and Kuk-Jin Yoon. Confidence-based data association and discriminative deep appearance learning for robust online multi-object tracking. *PAMI*, 40(3):595–610, 2017. 1, 2

[2] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixe. Tracking without bells and whistles. In *ICCV*, pages 941–951, 2019. 1, 2, 4, 6, 7

[3] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008:1–10, 2008. 6

[4] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *ICIP*, pages 3464–3468, 2016. 1

[5] Guillem Brasó and Laura Leal-Taixé. Learning a neural solver for multiple object tracking. In *CVPR*, pages 6247–6257, 2020. 1, 2, 6, 7

[6] Peng Chu and Haibin Ling. Famnet: Joint learning of feature, affinity and multi-dimensional assignment for online multiple object tracking. In *ICCV*, pages 6172–6181, 2019. 3

[7] Qi Chu, Wanli Ouyang, Hongsheng Li, Xiaogang Wang, Bin Liu, and Nenghai Yu. Online multi-object tracking using cnn-based single object tracker with spatial-temporal attention mechanism. In *ICCV*, pages 4836–4845, 2017. 3

[8] Peng Dai, Renliang Weng, Wongun Choi, Changshui Zhang, Zhangping He, and Wei Ding. Learning a proposal classifier for multiple object tracking. In *CVPR*, pages 2443–2452, 2021. 1, 2, 6

[9] Patrick Dendorfer, Hamid Rezatofighi, Anton Milan, Javen Shi, Daniel Cremers, Ian Reid, Stefan Roth, Konrad Schindler, and Laura Leal-Taixé. Mot20: A benchmark for multi object tracking in crowded scenes. *arXiv preprint arXiv:2003.09003*, 2020. 6

[10] Caglayan Dicle, Octavia I Camps, and Mario Sznaier. The way they move: Tracking multiple targets with similar appearance. In *ICCV*, pages 2304–2311, 2013. 1, 2

[11] Piotr Dollár, Ron Appel, Serge Belongie, and Pietro Perona. Fast feature pyramids for object detection. *PAMI*, 36(8):1532–1545, 2014. 2

[12] Andreas Ess, Bastian Leibe, and Luc Van Gool. Depth and appearance for mobile scene analysis. In *ICCV*, pages 1–8, 2007. 1, 2

[13] Pedro F. Felzenszwalb, Ross B. Girshick, David A. McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *PAMI*, 32(9):1627–1645, 2010. 2

[14] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 7

[15] Song Guo, Jingya Wang, Xinchao Wang, and Dacheng Tao. Online multiple object tracking with cross-task synergy. In *CVPR*, pages 8136–8145, 2021. 1, 2, 3, 6

[16] Jiawei He, Zehao Huang, Naiyan Wang, and Zhaoxiang Zhang. Learnable graph matching: Incorporating graph partitioning with deep feature learning for multiple object tracking. In *CVPR*, pages 5299–5309, 2021. 1, 2, 6

[17] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *ICCV*, pages 2961–2969, 2017. 3

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 6

[19] Andrea Hornakova, Roberto Henschel, Bodo Rosenhahn, and Paul Swoboda. Lifted disjoint paths with application in multiple object tracking. In *ICML*, pages 4364–4375, 2020. 2, 6

[20] Andrea Hornakova, Timo Kaiser, Paul Swoboda, Michal Rolinek, Bodo Rosenhahn, and Roberto Henschel. Making higher order mot scalable: An efficient approximate solver for lifted disjoint paths. In *ICCV*, pages 6330–6340, 2021. 2

[21] Ruibing Hou, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen. Temporal complementary learning for video person re-identification. In *ECCV*, pages 388–405, 2020. 4

[22] Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. *ASME Journal of Basic Engineering*, 82(1):35–45, 1960. 2

[23] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[24] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 2, 8

[25] Laura Leal-Taixé, Cristian Canton-Ferrer, and Konrad Schindler. Learning by tracking: Siamese cnn for robust target association. In *CVPRW*, pages 33–40, 2016. 1, 2

[26] Wei Li, Rui Zhao, Tong Xiao, and Xiaogang Wang. Deepreid: Deep filter pairing neural network for person re-identification. In *CVPR*, pages 152–159, 2014. 6

[27] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, pages 2117–2125, 2017. 6

[28] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Proceedings of the ECCV*, pages 740–755, 2014. 6

[29] Zhichao Lu, Vivek Rathod, Ronny Votel, and Jonathan Huang. Retinatrack: Online single stage joint detection and tracking. In *CVPR*, pages 14668–14678, 2020. 1, 3

[30] Jonathon Luiten, Aljosa Osep, Patrick Dendorfer, Philip Torr, Andreas Geiger, Laura Leal-Taixé, and Bastian Leibe. Hota: A higher order metric for evaluating multi-object tracking. *IJCV*, pages 1–31, 2020. 6

[31] Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016. 6

[32] Bo Pang, Yizhuo Li, Yifan Zhang, Muchen Li, and Cewu Lu. Tubetk: Adopting tubes to track multi-object in a one-step training model. In *CVPR*, pages 6308–6318, 2020. 3

[33] Jinlong Peng, Changan Wang, Fangbin Wan, Yang Wu, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yanwei Fu. Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking. In *ECCV*, pages 145–161, 2020. 3

[34] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018. 1, 3

[35] Shaoqing Ren, Kaiming He, Ross B. Girshick, and Jian Sun. Faster R-CNN: towards real-time object detection with region proposal networks. *PAMI*, 39(6):1137–1149, 2017. 1, 2, 4, 6, 7

[36] Weihong Ren, Xinchao Wang, Jiandong Tian, Yandong Tang, and Antoni B Chan. Tracking-by-counting: Using network flows on crowd density maps for tracking multiple targets. *TIP*, 30:1439–1452, 2020. 2

[37] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *ECCVW*, pages 17–35, 2016. 6

[38] Ergys Ristani and Carlo Tomasi. Features for multi-target multi-camera tracking and re-identification. In *CVPR*, pages 6036–6046, 2018. 1, 2

[39] Fatemeh Saleh, Sadegh Aliakbarian, Hamid Rezatofighi, Mathieu Salzmann, and Stephen Gould. Probabilistic tracklet scoring and inpainting for multiple object tracking. In *CVPR*, pages 14329–14339, 2021. 2, 6

[40] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*, 2018. 6

[41] Daniel Stadler and Jurgen Beyerer. Improving multiple pedestrian tracking by track management and occlusion handling. In *CVPR*, pages 10958–10967, 2021. 6

[42] Daniel Stadler and Jürgen Beyerer. Modelling ambiguous assignments for multi-person tracking in crowds. In *WACV*, pages 133–142, 2022. 7

[43] Siyu Tang, Mykhaylo Andriluka, Bjoern Andres, and Bernt Schiele. Multiple people tracking by lifted multicut and person re-identification. In *CVPR*, pages 3539–3548, 2017. 1

[44] Pavel Tokmakov, Jie Li, Wolfram Burgard, and Adrien Gaidon. Learning to track with object permanence. In *ICCV*, pages 10860–10869, 2021. 7

[45] Paul Voigtlaender, Michael Krause, Aljosa Osep, Jonathon Luiten, Berin Balachandar Gnana Sekar, Andreas Geiger, and Bastian Leibe. Mots: Multi-object tracking and segmentation. In *CVPR*, pages 7942–7951, 2019. 1, 2, 3

[46] Xingyu Wan, Jiakai Cao, Sanping Zhou, Jinjun Wang, and Nanning Zheng. Tracking beyond detection: Learning a global response map for end-to-end multi-object tracking. *TIP*, 30:8222–8235, 2021. 3

[47] Qiang Wang, Yun Zheng, Pan Pan, and Yinghui Xu. Multiple object tracking with correlation learning. In *CVPR*, pages 3876–3886, 2021. 7

[48] Shuai Wang, Hao Sheng, Yang Zhang, Yubin Wu, and Zhang Xiong. A general recurrent tracking framework without real data. In *ICCV*, pages 13219–13228, 2021. 7

[49] Yongxin Wang, Kris Kitani, and Xinshuo Weng. Joint object detection and multi-object tracking with graph neural networks. In *ICRA*, pages 13708–13715. IEEE, 2021. 2

[50] Zhongdao Wang, Liang Zheng, Yixuan Liu, and Shengjin Wang. Towards real-time multi-object tracking. In *ECCV*, pages 107–122, 2020. 1, 2, 3

[51] Xinshuo Weng, Yongxin Wang, Yunze Man, and Kris M Kitani. Gnn3dmot: Graph neural network for 3d multi-object tracking with 2d-3d multi-feature learning. In *CVPR*, pages 6499–6508, 2020. 2

[52] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *ICIP*, pages 3645–3649, 2017. 1, 2

[53] Jialian Wu, Jiale Cao, Liangchen Song, Yu Wang, Ming Yang, and Junsong Yuan. Track to detect and segment: An online multi-object tracker. In *CVPR*, pages 12352–12361, 2021. 3

[54] Yihong Xu, Aljosa Osep, Yutong Ban, Radu Horaud, Laura Leal-Taixé, and Xavier Alameda-Pineda. How to train your deep multi-object tracker. In *CVPR*, pages 6787–6796, 2020. 1, 2, 6

[55] Bo Yang, Chang Huang, and Ram Nevatia. Learning affinities and dependencies for multi-target tracking using a crf model. In *CVPR*, pages 1233–1240, 2011. 2

[56] Fan Yang, Xin Chang, Sakriani Sakti, Yang Wu, and Satoshi Nakamura. Remot: A model-agnostic refinement for multiple object tracking. *Image and Vision Computing*, 106:104091, 2021. 7

[57] Junbo Yin, Wenguan Wang, Qinghao Meng, Ruigang Yang, and Jianbing Shen. A unified object motion and affinity model for online multi-object tracking. In *CVPR*, pages 6768–6777, 2020. 3

[58] Sisi You, Hantao Yao, and Changsheng Xu. Multi-target multi-camera tracking with optical-based pose association. *CSVT*, 31(8):3105–3117, 2021. 2

[59] Sisi You, Hantao Yao, and Changsheng Xu. Multi-object tracking with spatial-temporal topology-based detector. *CSVT*, 32(5):3023–3035, 2022. 2

[60] Andrei Zanfir and Cristian Sminchisescu. Deep learning of graph matching. In *CVPR*, pages 2684–2693, 2018. 5

[61] Li Zhang, Yuan Li, and Ramakant Nevatia. Global data association for multi-object tracking using network flows. In *CVPR*, pages 1–8, 2008. 2

[62] Yang Zhang, Hao Sheng, Yubin Wu, Shuai Wang, Weifeng Lyu, Wei Ke, and Zhang Xiong. Long-term tracking with deep tracklet association. *TIP*, 29:6694–6706, 2020. 2

[63] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *ECCV*, pages 1–21. Springer, 2022. 7

[64] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *IJCV*, pages 1–19, 2021. 1, 2, 3, 7

[65] Liang Zheng, Liyue Shen, Lu Tian, Shengjin Wang, Jingdong Wang, and Qi Tian. Scalable person re-identification: A benchmark. In *ICCV*, pages 1116–1124, 2015. 6

[66] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 1, 7

[67] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. *arXiv preprint arXiv:1904.07850*, 2019. 3

[68] Ji Zhu, Hua Yang, Nian Liu, Minyoung Kim, Wenjun Zhang, and Ming-Hsuan Yang. Online multi-object tracking with dual matching attention networks. In *ECCV*, pages 366–382, 2018. 3