electronics

MDPI

Article

Graph Attention Networks and Track Management for Multiple Object Tracking

Yajuan Zhang [1], Yongquan Liang [1], Ahmed Elazab [2], Zhihui Wang [1,*] and Changmiao Wang [3,*]

[1] College of Computer Science and Engineering, Shandong University of Science and Technology, Qingdao 266510, China
[2] School of Biomedical Engineering, Shenzhen University, Shenzhen 518060, China
[3] Shenzhen Research Institute of Big Data, Shenzhen 518172, China
* Correspondence: zh_wang@sdust.edu.cn (Z.W.); cmwangalbert@gmail.com (C.W.)

Abstract: Multiple object tracking (MOT) constitutes a critical research area within the field of computer vision. The creation of robust and efficient systems, which can approximate the mechanisms of human vision, is essential to enhance the efficacy of multiple object-tracking techniques. However, obstacles such as repetitive target appearances and frequent occlusions cause considerable inaccuracies or omissions in detection. Following the updating of these inaccurate observations into the tracklet, the effectiveness of the tracking model, employing appearance features, declines significantly. This paper introduces a novel method of multiple object tracking, employing graph attention networks and track management (GATM). Utilizing a graph attention network, an attention mechanism is employed to capture the relationships of nodes within the graph as well as node-to-node correlations across graphs. This mechanism allows selective focus on the features of advantageous nodes and enhances discriminability between node features, subsequently improving the performance and robustness of multiple object tracking. Simultaneously, we categorize distinct tracklet states and introduce an efficient track management method, which employs varying processing techniques for tracklets in diverse states. This method can manage occluded tracks in crowded scenes and improves tracking accuracy. Experiments conducted on three challenging public datasets (MOT16, MOT17, and MOT20) demonstrate that our method could deliver competitive performance.

Keywords: multiple object tracking; tracking-by-detection; graph attention networks; track management

Citation: Zhang, Y.; Liang, Y.; Elazab, A.; Wang, Z.; Wang, C. Graph Attention Networks and Track Management for Multiple Object Tracking. *Electronics* **2023**, *12*, 4079. https://doi.org/10.3390/electronics12194079

Academic Editor: Hamed Mozaffari

Received: 26 July 2023
Revised: 23 August 2023
Accepted: 25 August 2023
Published: 28 September 2023

1. Introduction

Over recent years, multiple object tracking (MOT) has garnered significant interest within the field of computer vision. Mainstream trackers typically employ the tracking-by-detection paradigm, correlating detected targets throughout a video sequence over multiple frames. The tracking process within this paradigm generally involves two stages. Initially, the target is detected and identified, which is followed by the performance of data association on the detected target, enabling the tracking of the target's movement within the frame. Challenges associated with the detection phase encompass issues such as target deformation, background clutter, and variations in light. The advent of deep learning has led to substantial performance enhancements in some target detectors [1,2]. However, the data association phase remains complex due to obstacles such as occlusion and target interaction. Pedestrians, being among the most common objects tracked in MOT, present unique challenges such as extensive pose variations, resemblant appearances, and frequent occlusions, resulting in considerable errors or omitted detection areas. Following the updating of these inaccurate observations into the tracklet, the effectiveness of the data association, utilizing appearance features, becomes considerably reduced. In recent years, models employed attention mechanisms to capture relationships among different words or

between various image parts and have gained widespread attention due to their superior performance over traditional convolutional neural network models. Inspired by this, we construct a detection graph and a tracklet graph using detection and tracklet as vertices. We then employ graph attention networks to capture the relationships within the graph nodes and the node-to-node relationships between the graphs. Selectively focusing on the features of beneficial nodes allows ignoring irrelevant nodes, as they contribute minimally to the data association and may even introduce noise that affects the data association. This selective attention mechanism enhances the discriminability between node features, thereby improving the performance and robustness of MOT.

Additionally, within the framework of online multi-target tracking, targets randomly entering or exiting the scene present a significant challenge. Therefore, efficient creation and termination strategies for tracklets are of utmost importance. Inaccurate track management algorithms may result in target identity switching, thereby undermining tracking effectiveness. In response to this, we propose an effective track management methodology in this paper. This methodology entails the establishment of four distinct tracklet states: temporary, confirmed, occluded, and deleted. These states facilitate various processes such as tracklet creation, confirmation, occluded tracklet processing, and tracklet deletion.

The work presented in this paper contributes to the field in the following ways:

- We utilize graph attention networks with attention mechanisms to capture and model the inherent graph structure and cross-graph information. By focusing on the more relevant node features, the discriminative power of the model is enhanced, thereby improving the distinguishability of node features.
- We propose a track management method that systematically manages tracklet states and performs tasks such as tracklet creation, confirmation, occluded tracklet processing, and tracklet deletion. This method can handle occlusions in crowded scenes and improve tracking accuracy.
- We combine graph attention networks and track management to design simple and versatile online trackers that exhibit advanced performance on the MOT16, MOT17, and MOT20 datasets.

The remainder of the paper is structured as follows: Section 2 presents the related work on multi-target tracking and attention modeling. Section 3 provides a detailed description of the proposed methodology. Section 4 conducts an in-depth study of the proposed method and compares it with other tracking methodologies. Finally, Section 5 summarizes the characteristics of the proposed tracker and provides directions for further exploration.

2. Related Work

2.1. Multiple Object Tracking

In recent years, MOT has emerged as a research hotspot with wide-ranging applications in fields such as intelligent surveillance, autonomous driving, and behavioral analysis [3]. To associate targets with stability and efficiency, certain studies [4,5] attempted to calculate the distance of the spatial position as a cost matrix by estimating the target's position in the subsequent frame and subsequently applying the Hungarian algorithm for association [6]. OC-SORT [4] predicts the target's state in the next frame using the Kalman filter [7] and calculates the Intersection over Union (IoU) for bipartite graph matching. While this tracking method is apt for short-term tracking, its effectiveness diminishes in scenes involving moving cameras or requiring long-term associations. Contrarily, some works represent detections or tracklets as single vertices, formulating the data association problem as a combinatorial optimization problem. For instance, GMTracker [8] constructs a detection graph and a tracklet graph, considers the matching between the detection graph and the tracklet graph as a convex optimization problem, and achieves end-to-end multi-target tracking through relaxation. Several studies have also utilized a convolutional neural network approach that fuses features through an aggregation mechanism to enhance the distinctiveness of appearance features [9–11]. GCT [9], for example, designed a context graph convolutional network to learn the adaptive features of a target using the

context of the current frame and integrated the spatio-temporal structure of historical target samples to model the structured representation of these samples. Brasó et al. [10] used a message-passing network with a time-aware update step to integrate deep features with higher-order information, accounting for global interactions between detections. Tracking methods using combinatorial optimization are generally more complex. Tracking methods using graph convolutional networks usually assign equal importance to neighbors during message passing. Unlike them, this paper pursues real-time tracking, and we use graph attention networks to assign different weights to neighboring nodes according to their characteristics. Later, we solve the problem by linear assignment to improve the tracking performance of the algorithm as a whole.

2.2. Attention Model

Originally proposed within the realm of natural language processing, attention mechanisms have since been introduced to the computer vision field. Recent studies have demonstrated the impressive performance of attention networks in computer vision, which is achieved by introducing trainable attention weights to aggregate neighborhood information. Several works leverage attention networks to focus on the most relevant regions of an image or video sequence in order to extract more discriminative features [12–16]. For example, STAM [15] employed a spatio-temporal attention mechanism for online multi-target tracking, mitigating the occlusion problem by learning the target's visibility map, which is then used to infer the spatial attention map. DMAN [16] introduced a dual matching attentional network with spatial and temporal attention mechanisms to handle noise detection and frequent target interactions. The spatial attention module centers on the matching patterns of the input image pairs, while the temporal attention module adaptively assigns varied attention levels to different samples in the tracklet. Certain models establish context dependencies through attention, attaching context information to different objects to effectively distinguish their identities. For instance, GCA-LSTM [17] applied global context-aware attention to capture the relevant locations of motion features, selectively focusing on informative joints in the action sequence. DSGA [18] proposed a distractor-suppressing graph attention network to effectively suppress the distractors of target localization, reducing their influence on learning attentional weighting features.

3. Methodology

In this section, we present an MOT methodology that utilizes graph attention networks and track management, as illustrated in Figure 1. Initially, features are extracted from the detection and subsequently enhanced using a graph attention network. Concurrently, the affinity between nodes is computed, as is the affinity of features that have not been enhanced. The affinities obtained from these two processes are then accumulated, and the Hungarian algorithm is employed to correlate the detections and tracklets. Subsequently, the detection and tracklet of previous unsuccessful matches are combined with motion and appearance cues to compute a cost matrix for secondary data association. Finally, the status of the tracklet is updated using the track management module.

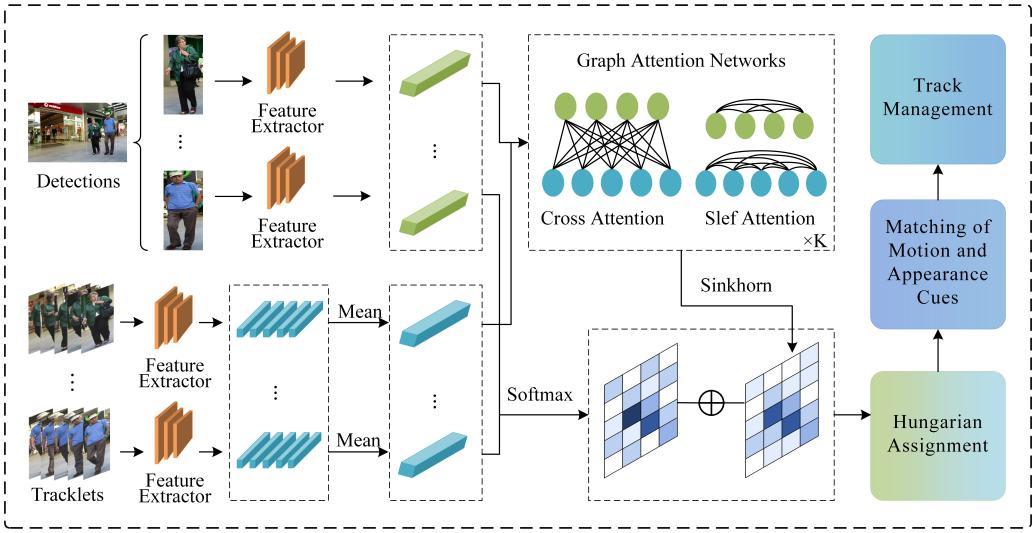


Figure 1. Overview of the proposed tracking method.

3.1. Graph Attention Networks

In this context, we leverage graph attention networks [19] to capture the intricate relationships of nodes within a graph as well as the relationships between nodes across different graphs. This approach enhances the distinctiveness of node features, thereby improving their discriminability.

Constructing detection and tracklet graphs. Let us define the set of detections at frame t as $\mathcal{D} = \{D_1, D_2, \ldots, D_{n_d}\}$ and the set of tracklets as $\mathcal{T} = \{T_1, T_2, \ldots, T_{n_t}\}$. Each tracklet comprises a series of detections sharing the same tracklet id, implying that a detection from a successful data association is added to the tracklets' collection. We construct the detection graph and the tracklet graph by using the detection and tracklet of frame t as vertices, respectively. We denote the detection graph as $G_D = (V_D, E_D)$ and the tracklet graph as $G_T = (V_T, E_T)$. Both G_D and G_T are complete graphs, with V_D and V_T representing sets of vertices, and E_D and E_T representing sets of edges. Vertex $i \in V_D$ represents detection D_i, and vertex $j \in V_T$ represents tracklet T_j.

Graph attention network enhanced node features. The graph attention network comprises a cross-attention layer and a self-attention layer. The information aggregation process of the detection graph within the cross-attention layer is given by:

$$\mathcal{F}_{ca}\left(\mathbf{h}_i^{(k)}, \{\mathbf{h}_j^{(k)}\}_{j\in V_T}\right) = \mathbf{h}_i^{(k)} \Big\| \sum_{j\in V_T} \mathbf{A}_{i,j}^{ca}\left(\mathbf{h}_i^{(k)}, \mathbf{h}_j^{(k)}\right), \tag{1}$$

where $\mathbf{h}_i^{(k)}$ denotes the vertex feature of the k-th aggregation in the detection graph, with the initial vertex feature being the appearance feature $\mathbf{a}_i$ obtained from the feature extraction network, i.e., $\mathbf{h}_i^{(0)} = \mathbf{a}_i$. Similarly, $\mathbf{h}_j^{(k)}$ denotes the vertex feature of the k-th aggregation in the tracklet graph. Its initial vertex feature is calculated as the mean of the appearance features of the detections sharing the same tracklet ID that appeared in the previous frames of the tracklet, i.e., $\mathbf{h}_j^{(0)} = \mathcal{F}_{mean}(\{\mathbf{a}_{(j)}\})$.

The symbol $||$ represents concatenation. The attention coefficient $\mathbf{A}_{i,j}^{ca}$ is calculated as follows:

$$\mathbf{A}_{i,j}^{ca} = \frac{\exp\left(\left\langle \mathbf{ff}_h \odot \mathbf{h}_i^{(k)}, \mathbf{ff}_h \odot \mathbf{h}_j^{(k)} \right\rangle\right)}{\sum_{\hat{j}\in V_T} \exp\left(\left\langle \mathbf{ff}_h \odot \mathbf{h}_i^{(k)}, \mathbf{ff}_h \odot \mathbf{h}_{\hat{j}}^{(k)} \right\rangle\right)}, \tag{2}$$

where $\langle \cdot, \cdot \rangle$ represents the inner product. The channel attention weight vector, $\mathbf{ff}_h$, as described in [20], is calculated as:

$$\mathbf{ff}_h = \sigma\left(\mathbf{W}_{ca} * \mathbf{H}_{avg} + \mathbf{W}_{ca} * \mathbf{H}_{max}\right), \tag{3}$$

where $\mathbf{H}_{avg}$ and $\mathbf{H}_{max}$ represent the average-pooled and max-pooled features, respectively. $\mathbf{W}_{ca}$ is a trainable weight matrix, and $\sigma(\cdot)$ denotes the hyperbolic tangent function.

For the self-attention layer, graph convolutional networks (GCNs) [21] typically assign equal importance to neighbors during the message passing from neighborhoods to the central node. Contrarily, we employ the attention mechanism to learn the relationship between neighborhood features. The message aggregation process for the self-attention layer is given by:

$$\mathcal{F}_{sa}\left(\mathbf{h}_i^{ca}, \{\mathbf{h}_{i'}^{ca}\}_{i' \in \mathcal{N}_i}\right) = \Big\Vert_{k=1}^{K} \sum_{i' \in \mathcal{N}_i} \mathbf{A}_{i,i'}^{k} * \mathbf{h}_{i'}^{ca} * \mathbf{W}_{sa}, \tag{4}$$

where $\mathbf{h}_i^{ca}$ and $\mathbf{h}_{i'}^{ca}$ are the node features after the cross-attention layer. $\mathcal{N}_i$ denotes the neighborhood of node i in the detection graph. $\mathbf{W}_{sa}$ is a trainable weight matrix. The attention coefficient $\mathbf{A}_{i,i'}^{k}$ is computed as:

$$\mathbf{A}_{i,i'}^{k} = \frac{\exp\left(\left\langle \mathbf{h}_i^{ca} * \mathbf{W}_{sa}, \mathbf{h}_{i'}^{ca} * \mathbf{W}_{sa} \right\rangle\right)}{\sum_{\hat{i} \in \mathcal{N}_i} \exp\left(\left\langle \mathbf{h}_i^{ca} * \mathbf{W}_{sa}, \mathbf{h}_{\hat{i}}^{ca} * \mathbf{W}_{sa} \right\rangle\right)}, \tag{5}$$

As shown in Figure 2, beneficial information is selected from the relationship between the detection graph and tracklet graph and the relationship between nodes in the graph through cross-attention and self-attention, and more attention is focused on relevant node features to enhance the distinguishability of node features. In the cross-attention layer, the features of objects are enhanced between detections and tracklets, while the enhancement in the self-attention layer is operated in detections and tracklets independently.

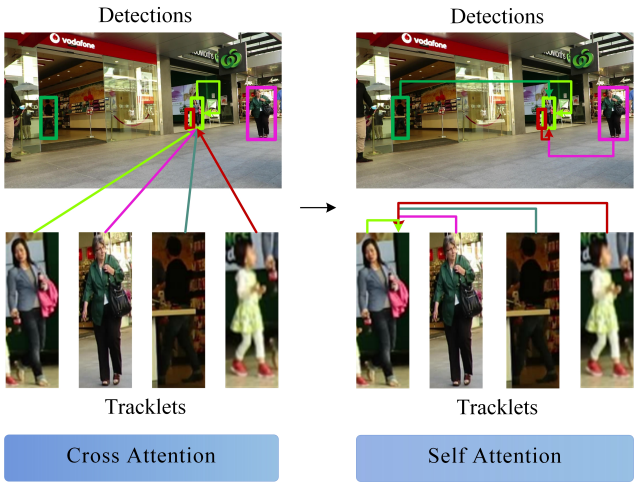


Figure 2. The diagram of feature enhancement based on graph attention network. For simplicity, we only draw the orientation relations for specific objects. Different objects are differentiated by bounding boxes of varying colors.

Data association: The affinity between the detection and the tracklet after feature augmentation by the graph attention network is calculated as:

$$\mathbf{S}_{i,j} = \text{Sinkhorn}(\mathbf{A}_{i,j}^{ca}) + \text{DS}\left(\cos(\mathbf{h}_i, \mathbf{h}_j) + \text{IoU}(\mathbf{g}_i, \mathbf{g}_j)\right). \tag{6}$$

Sinkhorn [22] consists of linear assignments given a predefined assignment cost and has been shown to be efficient for network-based permutation prediction. Sinkhorn$(\cdot)$ signifies the use of the Sinkhorn network, alternately normalizing in rows and columns until convergence. DS$(\cdot)$ denotes a softmax operation performed on the row and column dimensions, respectively. $\cos(\cdot,\cdot)$ signifies the cosine similarity between the computed features, IoU$(\cdot,\cdot)$ denotes the intersection over union between two bounding boxes, $\mathbf{g}_i$ is the bounding box of detection D_i, and $\mathbf{g}_j$ is the bounding box in the current frame estimated by the Kalman filter for tracklet T_j. Kalman filtering is used to estimate the future states of a specific target based on a series of its past variables and uncertain measurements. It is one of the most important and common estimation algorithms. The Kalman filter algorithm is used to predict the possible position of each tracklet in the new frame, which can be used to judge the correlation between detections and tracklets in the frame.

Subsequently, the matching matrix $\mathbf{M}$ is obtained by assignment using the Hungarian algorithm:

$$\mathbf{M} = \text{Hungarian}(\mathbf{S}). \tag{7}$$

Apply the Hungarian algorithm on the affinity matrix $\mathbf{S}$, then discretize it into a (0, 1)-matrix, and use this matrix as the matching matrix in the phase of target association. A value of 1 in the matching matrix indicates a success match between the detection and the tracklet with corresponding index, while 0 signifies no match.

3.2. Matching of Motion and Appearance Cues

Generally, superior detection and tracking accuracy can be achieved by leveraging motion cues, while longer association can be derived from appearance cues. For those detections and tracklets not successfully associated in the previous stage, we combine motion and appearance cues to compute the cost matrix for secondary data association. The cost matrix calculation process is as follows:

$$\mathbf{C} = \lambda \mathbf{d}_m + (1-\lambda)\mathbf{d}_a, \tag{8}$$

where λ represents the weight coefficient, and $\mathbf{d}_m$ and $\mathbf{d}_a$ denote the motion distance and appearance distance between the detection and tracklet, respectively. The motion distance implies the generalized Intersection over Union [23] between the bounding box of the detection and the predicted bounding box of the tracklet. The positional relationship between the detection and the tracklet is determined by introducing the smallest enclosing box encapsulating the two bounding boxes. $\mathbf{d}_m$ is computed as follows:

$$\mathbf{d}_m(i,j) = 1 - \left(\frac{|f_i \cap f_j|}{|f_i \cup f_j|} - \frac{|f_c \backslash f_i \cup f_j|}{|f_c|} + 1 \right) \Big/ 2, \tag{9}$$

where f_i represents the area of the bounding box of the detection, and f_j denotes the area of the predicted bounding box of the tracklet. f_c is the area of the smallest box that can encapsulate the bounding box of the detection and the predicted bounding box of the tracklet. The procedure for computing the appearance distance is:

$$\mathbf{d}_a(i,j) = \begin{cases} \frac{\mathbf{a}_i \cdot \mathbf{a}_j}{||\mathbf{a}_i|| \cdot ||\mathbf{a}_j||} & \mathbf{d}_m(i,j) < \delta, \\ 2(\mathbf{d}_m(i,j)+1) & otherwise, \end{cases} \tag{10}$$

where $\mathbf{a}_i$ and $\mathbf{a}_j$ are the appearance characteristics of the detection and tracklet, respectively. Finally, the matching matrix is derived based on the cost matrix $\mathbf{C}$ using the Hungarian algorithm.

As shown in Figure 3, the object in the yellow rectangle at frame 115 is gradually occluded by others. The extracted appearance features are disturbed and inaccurate when this pedestrian is occluded. When the object reappears (frame 156), the matching results

become less reliable if only these inaccurate appearance features are used. To deal with this issue, motion cues are used to compensate for the reduced discriminability of appearance cues, enabling a stronger correlation between detections and tracklets.

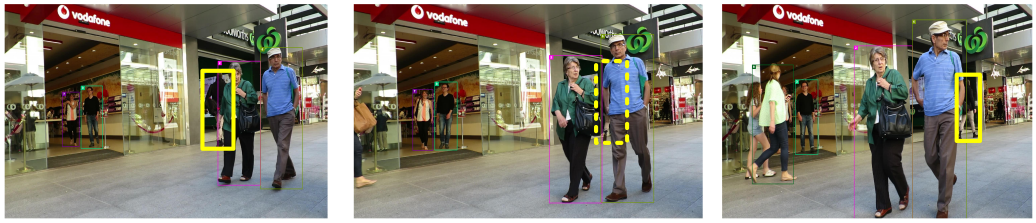


Figure 3. The process of being obscured and reappearing for the specific object in yellow rectangle. The dashed yellow box indicates the location of the occluded object, which the object detector failed to recognize. The solid yellow box indicates the bounding box of the object obtained using the object detector.

This straightforward yet effective method of combining motion and appearance cues enables a stronger correlation between the detection and tracklet by assigning suitable weights to motion and appearance distances in different scenarios. For instance, if the tracked target's appearance features are distinctly unique, a higher weight can be assigned to the appearance distance. Conversely, in crowded scenes with cluttered backgrounds, a higher weight can be assigned to the motion distance. Hence, even under circumstances where the discriminability of appearance features is degraded, this method can still correlate through motion cues, thereby preserving robust tracking performance.

3.3. Track Management

Online multi-target tracking frameworks face several challenges, including the creation and termination of tracklets based on distinguishing true detection results from false positives as well as handling occluded tracklets in crowded scenarios. Consequently, we propose an effective track management method, as illustrated in Figure 4. This method establishes four tracklet states: temporary, confirmed, occluded, and deleted, to facilitate tracklet creation, confirmation, processing of occluded tracklets, and tracklet deletion.

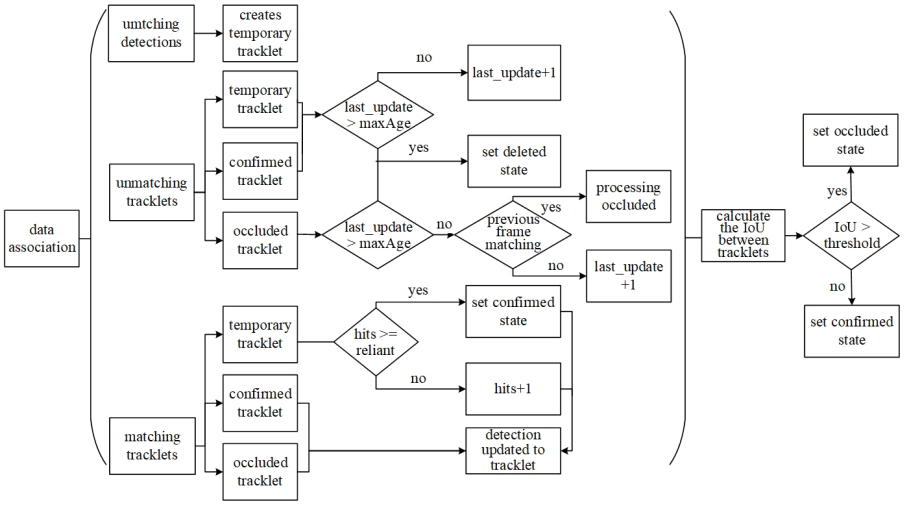


Figure 4. Update the status of the tracklet after data association.

Temporary State : In data association, detections that are not successfully matched are initialized as tracklets. However, the detector might misidentify a cluttered background as the target to be tracked, and initializing an incorrect detection as a new tracking target can

influence the subsequent tracking results. To distinguish true detections from false positive results, we assign the initialized tracklet to a temporary state.

Confirmed State: A tracklet initialized in a temporary state is considered a real target, and its state is switched from temporary to confirmed if it is successfully associated for three consecutive frames.

Occluded State: The Intersection over Union (IoU) is computed between tracklets. If the IoU exceeds 0.4, the tracklets are assigned an occluded state. When targets occlude each other, especially during complete occlusions, even the best detector cannot detect the target without context information. Thus, for a tracklet in an occluded state, if it was successfully associated with the previous frame but did not match successfully in the current frame, the Kalman filtered motion model is used to estimate the position of the missing target in the current frame.

Deleted State: A track is terminated if consecutive associations are missed. The number of frames since the track's most recent successful match is recorded, and if it exceeds a predefined threshold of 100 frames, the target is considered to have exited the scene and is removed from the tracking set. In certain cases, an object about to leave the scene may have its tracklet stop at the image boundary. Such a tracklet may later be inaccurately re-matched by a new object entering the scene. To effectively avoid such false matches, when the center of the object's bounding box exceeds the image boundary and the object's moving speed points toward the image boundary, the tracklet is no longer updated and is subsequently deleted from the tracking set.

As shown in Figure 5, a new target in the yellow dashed rectangle appears at the left boundary in frame 15, and it has been initialized to be a temporary tracklet. In frame 17, the status of the tracklet is changed to be the confirmed state after three consecutive frames with successful association. In frame 117, this target is partially occluded by a pedestrian wearing green clothes, and this tracklet has been set with an occluded state. Highly truncated objects appeared when they are just entering or leaving the camera's field of view. At this time, the extracted appearance features are limited, and it is impossible to distinguish different instances effectively. The second line shows the process of the tracked object gradually leaving the camera's field of view. In frame 77, the object pointed to by the yellow arrow is about to leave the scene, at which time the object's appearance features are incomplete, and the position of its bounding box stays at the image boundary after the object has completely left the scene (dashed box in frame 85). The tracklet for such cases is set to be the deleted state, and it is removed from the tracking collection in time to prevent false associations with newly entering objects.

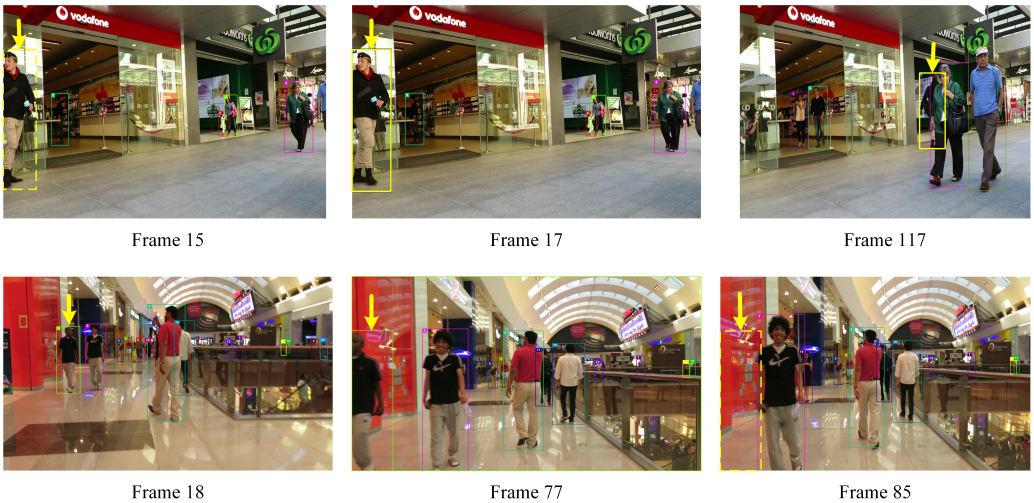


Figure 5. Changes in the tracklets' tracking states in different scenarios. Different objects are differentiated by bounding boxes of varying colors.

3.4. Loss Function

The proposed model is trained and supervised by leveraging the matching relationship between detection and tracklet. Due to the sparse correspondence between detection and tracklet, the final loss used to optimize the negative samples, accumulated from a high number of negative sample losses, is far greater than the loss used to optimize the positive samples. This discrepancy results in predictions that are more inclined toward the negative class. Although the application of focal loss [24] can address this imbalance through weight introduction, the weight is user-defined and highly reliant on practical experience. To overcome this issue, we utilize the Hungarian attention loss [25] to train the network to concentrate on "important" numbers. The loss is computed as follows:

$$\mathcal{L} = -\sum_{i=1}^{N_d}\sum_{j=1}^{N_t} \max\{\mathbf{M}_{i,j}, \mathbf{M}^G_{i,j}\}(\mathbf{M}^G_{i,j}\log \mathbf{S}_{i,j} + (1-\mathbf{M}^G_{i,j})\log(1-\mathbf{S}_{i,j})), \tag{11}$$

where $\mathbf{S}$ represents the affinity matrix between the detection and the tracklet. $\mathbf{M}$ is the matching matrix derived from the Hungarian algorithm, and $\mathbf{M}^G$ denotes whether the object belongs to the ground truth of the tracklet.

4. Experiments

4.1. Implementation Details

The experiments were executed using the PyTorch [26] deep learning framework and a two-layer graph attention network to achieve optimal performance. All experiments were conducted on a PC equipped with an NVIDIA GeForce MX450, 2GB RAM GPU, and an Intel(R) Core(TM) i7-11370H CPU. Following the approach in [8], appearance features were extracted utilizing the ReID network [27], with ResNet50 [28] serving as the backbone network. The model was trained on the MOT17 training set with an initial learning rate of 10^{-2}, and Adam [29] was employed as the optimizer. The value of λ was set to 0.7 for the MOT17 dataset and 0.3 for the MOT20 dataset. The value of δ was set to 0.5.

4.2. Datasets and Evaluation Metrics

4.2.1. Datasets

Our method has been evaluated on three widely used multi-target tracking datasets: MOT16 [30], MOT17 [30], and MOT20 [31]. These datasets provide public detection, but the annotations for the test sequences are not openly accessible. Thus, tracking results on the test sets need to be submitted to the official evaluation server (motchallenge.net), thereby facilitating a fair comparison between different tracking methods.

The MOT16 and MOT17 datasets contain identical video sequences but differ in their public detections. The DPM detector provides detection results for the MOT16 dataset, while the MOT17 dataset also includes detections from Faster R-CNN [32] and SDP [33]. The MOT20 dataset is a complex and dense dataset with crowded scenes, peaking at an average of 246 pedestrians per frame in extremely crowded scenes, which demands higher robustness for target association. The MOT20 dataset comprises eight videos, with public detection provided by the Faster R-CNN detector. In line with [34–38], we refine the bounding box using Tracktor [37] and CenterTrack [38] for the MOT16 and MOT17 datasets. For the MOT20 dataset, Tracktor is used for bounding box refinement.

4.2.2. Evaluation Metrics

To assess tracking performance, we utilize TrackEval [39] to compute various metrics, including Multiple Object Tracking Accuracy (MOTA) [40], Higher-Order Tracking Accuracy (HOTA) [41], IDF1 Score (IDF1) [42], Number of Identity Switches (IDSW) [43], Mostly Tracked Targets (MT), and Mostly Lost Targets (ML). MOTA, considered the most critical metric, combines false positives, missed targets, and identity switches to calculate the accuracy of multi-target tracking. MT and ML denote the ratio of Ground Truth (GT)

trajectories covered by a track hypothesis for at least 80% and at most 20% of their respective life span, respectively.

4.3. Ablation Study

To gain a more comprehensive understanding of our approach, we performed an ablation study on different components, including a matching module that employs graph attention networks for feature enhancement (GM), a matching module that incorporates motion and appearance cues (MA), and a track management module (TM). The impact of these components on MOT is illustrated in Table 1. Trackers that incorporate the GM module outperform trackers without the GM module by a large margin in HOTA and IDF1, even though they have the same MOTA. This is because using a graph attention network focuses more attention on relevant node features, and the affinity scores obtained through graph representation and matching are more reliable. This confirms that the tracker incorporating the GM module is more robust in target association than the tracker without the GM module, allowing the tracker to maintain the identity of the tracklet over a longer period of time. Another critical component is the track management module, which can effectively distinguish between true detections and false positives. Tracklets that manage occluded states can compensate for missed detections due to target occlusion, thereby enhancing MOTA and MT while reducing ML.

Table 1. Ablation study of the proposed GATM with different components, including a matching module that employs graph attention networks for feature enhancement (GM), a matching module that incorporates motion and appearance cues (MA), and a track management module (TM). ↑ indicates that a larger value is better and ↓ indicates that a smaller value is better.

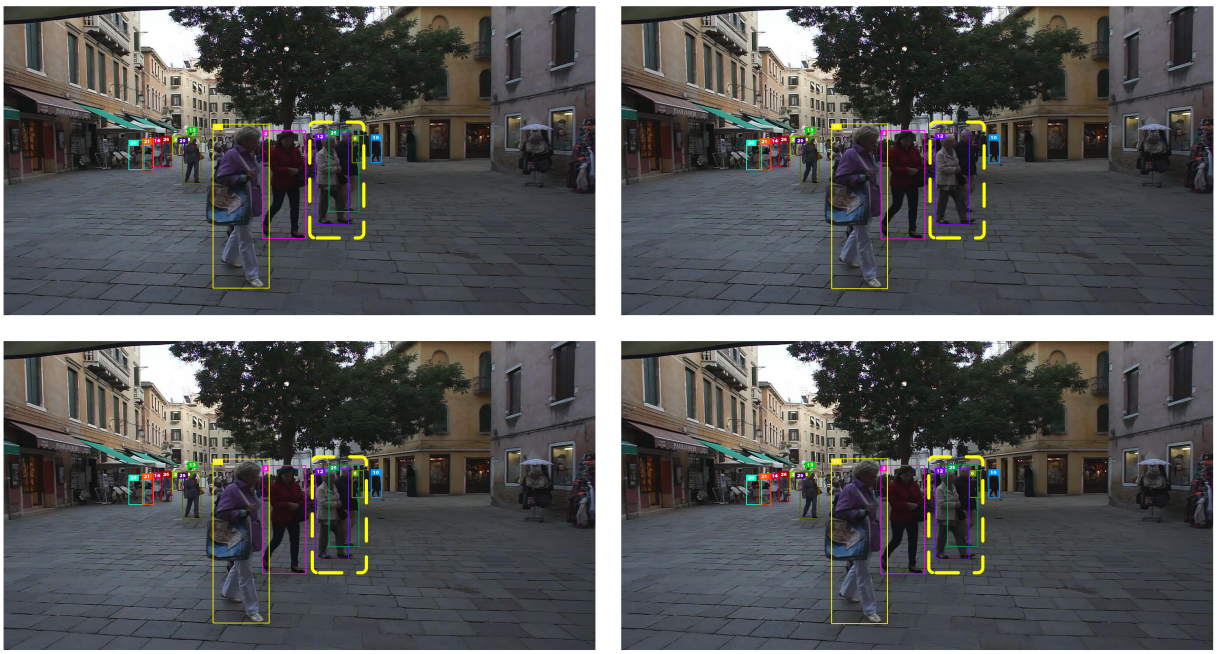
GM	MA	TM	MOTA↑	HOTA↑	IDF1↑	MT↑	ML↓	IDSW↓
√			62.3	59.7	68.4	548	373	949
	√		62.4	61.0	70.5	547	374	514
√	√		62.4	61.6	71.6	562	368	493
	√	√	63.4	61.5	71.0	602	352	512
√	√	√	63.4	62.2	72.4	605	352	517

When an object becomes progressively occluded, the appearance features of the targets can easily be conflated. Graph attention networks, however, allow for a selective focus on the most pertinent features, thereby enhancing target discriminability. As shown in Figure 6, within the yellow dashed box at frame 693, the target indicated by the arrow (referred to as the "new object") progressively eclipses the target behind it (referred to as the "17th object"). This results in an increasing similarity in appearance features between the two targets. By frame 701, the 17th target is completely obscured by the new object. Methods without the graph attention network incorrectly associate the new object with the 17th target. However, the graph attention network successfully differentiates between the 17th and new targets.

Addressing occluded targets using the track management module enhances the tracker's resistance to disruptions in crowded situations. This is illustrated in Figure 7, where there are two targets (identified as the "12th target" and the "25th target") enclosed in the yellow dashed box in frame 208. By frame 209, the 25th target gets occluded by the 12th target, leading to a missed detection. Without track management, the method fails to successfully associate with the 25th target. However, the method employing track management is capable of estimating the obscured target's location.

Frame 693 Frame 701

Figure 6. Qualitative analysis of tracking results on the MOT17-11 dataset using SDP detection. The top row presents tracking results achieved in the absence of the GM model. Conversely, the bottom row shows the tracking results obtained using the GM model. In these visuals, different objects are differentiated by bounding boxes of varying colors, and their respective IDs are indicated in the top-left corner of each box.



Frame 208 Frame 209

Figure 7. Qualitative analysis of tracking results on the MOT17-02 dataset using DPM detection. The top row illustrates tracking results obtained without the implementation of the TM model. In contrast, the bottom row demonstrates tracking results acquired using the TM model. In these representations, individual objects are differentiated by bounding boxes of diverse colors, with their corresponding IDs denoted at the top-left corner.

4.4. Tracking Efficiency

The experimental code has been executed using PyTorch version 1.9.1 on an 11th Gen Intel® Core™ i7-11370H CPU without the utilization of a GPU. To provide a visual representation of our approach's efficacy, the time consumed in the tracking process is computed. The tracker operates at a speed of 11 Hz on 21 video sequences extracted from the MOT17 training set.

4.5. MOT Challenge Evaluation Results

The proposed GATM tracker is bench-marked and contrasted with other tracking methodologies on the MOT16, MOT17, and MOT20 standards. In this section, we evaluate the GATM in two distinct settings. GATM_T and GATM_C amend the bounding box of public detection utilizing Tracktor and CenterTrack, respectively. The proposed method outperforms most existing methods, showing superior results across the majority of the evaluation metrics.

MOT16 dataset: Table 2 presents the results of the MOT16 test set. The proposed GATM_C yields superior outcomes concerning MOTA, IDF1, HOTA, MT, and ML, and it ranks second in IDSW. The highest MT obtained by the GATM_C method indicates that our method can produce stable and long-lasting tracklets, which is attributed to the proposed track management mechanism. In comparison to ArTIST-C [34], MOTA and IDF1 have improved by 1.2 and 6.9, respectively. When compared with trackers that employ Tracktor to refine public detection, GATM_T conspicuously outperforms them. For instance, when juxtaposed with GSM_Tracktor [36], MOTA is enhanced by 0.4 and IDF1 is enhanced by 4.6. Compared to Tracktor++v2, GATM_T has a higher IDF1 and a much lower IDSW, which shows that the proposed method can track one object with the same ID more robustly.

Table 2. Comparison with other trackers on the MOT16 test set. ↑ indicates that a larger value is better and ↓ indicates that a smaller value is better.

Methods	Refined Det	MOTA↑	IDF1↑	HOTA↑	MT↑	ML↓	IDSW↓
HISP_DAL [44]	-	37.4	30.5	25.7	7.6	50.9	2101
GMPHD_ReId [45]	-	40.4	50.1	36.0	11.5	43.1	789
BLSTM_MTP_O [35]	-	48.3	53.5	39.7	17.0	38.7	735
Tracktor++ [37]	Tracktor	54.4	52.5	42.3	19.0	36.9	682
Tracktor++v2 [37]	Tracktor	56.2	54.9	44.6	20.7	35.8	617
ArTIST-T [34]	Tracktor	56.6	57.8	-	22.4	37.5	519
GSM_Tracktor [36]	Tracktor	57.0	58.2	45.9	22.0	34.5	475
ArTIST-C [34]	CenterTrack	63.0	61.9	-	29.1	33.2	635
GATM _T (Ours)	Tracktor	57.4	62.8	48.7	23.2	33.1	347
GATM _C (Ours)	CenterTrack	64.2	68.8	53.4	32.0	28.7	458

The red represents the best results, blue represents the second best, and green represents the third best.

MOT17 dataset: Experimental results of the proposed GATM are reported and compared to other methods in Table 3. Despite the addition of more detectors in the MOT17 dataset compared to the MOT16 dataset, our method GATM_C continues to set the benchmark in most metrics among all competing works. Specifically, our proposed tracker, GATM_T, attains an MOTA of 57.6 and an IDF1 of 63.6 on the MOT17 dataset. In comparison with ArTIST-C, which uses similar detector enhancements, GATM_C improves MOTA and IDF1 by 0.9 and 7.9, respectively.

MOT20 dataset: As summarized in Table 4, the results of the proposed method, when applied to the challenging MOT20 dataset, demonstrate the method's efficacy. When compared with Tracktor++ v2, our method enhances MOTA and IDF1 by 0.1 and 2.0, respectively. Significant improvements are also observed in MT and ML, with MT increasing by 4.6 and ML decreasing by 2.1. In highly congested scenarios with frequent occlusions, GATM_T's improvement on several MOT metrics proves that using a graph attention network and performing track management yields better performance.

Table 3. Comparison with other trackers on the MOT17 test set. ↑ indicates that a larger value is better and ↓ indicates that a smaller value is better.

Methods	Refined Det	MOTA↑	IDF1↑	HOTA↑	MT↑	ML↓	IDSW↓
GMPHD_Re17 [45]	-	46.8	54.1	41.5	19.7	33.3	3865
DEEP_TAMA [46]	-	50.3	53.5	42.0	19.2	37.5	2192
TADN [47]	-	54.6	49.0	39.7	22.4	30.2	4869
Tracktor++ [37]	Tracktor	53.5	52.3	42.1	19.5	36.6	2072
BLSTM-MTP-T [35]	Tracktor	55.9	60.5	-	20.5	36.7	1188
Tracktor++v2 [37]	Tracktor	56.3	55.1	44.8	21.1	35.3	1987
GSM_Tracktor [36]	Tracktor	56.4	57.8	45.7	22.2	34.5	1485
ArTIST-T [34]	Tracktor	56.7	57.5	-	22.7	37.2	1756
CTTrackPub [38]	CenterTrack	61.5	59.6	48.2	26.4	31.9	2583
ArTIST-C [34]	CenterTrack	62.3	59.7	-	29.1	34.0	2062
GATM _T (Ours)	Tracktor	57.6	63.6	49.3	24.5	32.8	1163
GATM _C (Ours)	CenterTrack	63.2	67.6	52.7	31.4	29.8	1413

The red represents the best results, blue represents the second best, and green represents the third best.

Table 4. Comparison with other trackers on the MOT20 test set. ↑ indicates that a larger value is better and ↓ indicates that a smaller value is better.

Methods	Refined Det	MOTA↑	IDF1↑	HOTA↑	MT↑	ML↓	IDSW↓
SORT20 [7]	-	42.7	45.1	36.1	16.7	26.2	4470
GMPHD Rd20 [45]	-	44.7	43.5	35.6	23.6	22.1	7492
CT_v0 [48]	-	45.1	35.6	33.0	32.9	18.9	6492
IOU_KMM [49]	-	46.5	49.4	40.4	29.9	19.6	4509
Tracktor++v2 [37]	Tracktor	52.6	52.7	42.1	29.4	26.7	1648
ArTIST-T [34]	Tracktor	53.6	51.0	-	31.6	28.1	1531
GATM_T (Ours)	Tracktor	53.7	53.0	42.7	34.0	24.6	1956

The red represents the best results, blue represents the second best, and green represents the third best.

Overall, the proposed method attains competitive performance on the MOT16, MOT17, and MOT20 datasets. As the results show in Tables 2–4, this is evidenced by the higher IDF1 scores due to the ability of its graph attention and track management components to maintain the identity of tracklets for longer periods of time. This procedure allows GATM to keep more tracklets for more than 80% of their actual lifespan, resulting in very high MT and outperforming competing methods. High MOTA and IDF1 metrics across different scenarios demonstrate the robust association performance and generalization capability of our tracker. Moreover, our method, which primarily focuses on the association step, can be applied to any detector.

4.6. Visualization of the Results

As illustrated in Figures 8 and 9, our method effectively preserves the target identity across diverse scenarios. All scenes in Figure 8 are captured using a mobile camera, while a stationary camera records all scenes in Figure 9. The robust and reliable performance of the algorithm is demonstrated through successful tracking in video sequences filmed from various angles. For instance, the aspect ratios and sizes of the targets in groups c and d of Figure 8 display significant variation. Nevertheless, the tracker, with the assistance of the graph attention network, is capable of retaining its identity through appearance features. Group b and group c sequences in Figure 8 were shot with mobile cameras in busy commercial neighborhoods. Our method still had good tracking performance after experiencing chaotic background, varying light intensity, and a wide range of occlusions. In particular, the shooting time of group c is at night, and the appearance characteristics of pedestrians are fuzzy, so combining appearance information and movement information greatly improves the tracking effects. Tracking pedestrians on the road is particularly important in real-world autonomous driving environments in cities. Group e in Figure 8

shows a scenario in which pedestrians on both sides of the road are accurately tracked, which plays an essential role in the timely avoidance measures taken by subsequent vehicles to avoid traffic accidents. Considering that the motion of objects in a video is continuous, the higher the number of objects appearing in the same frame, the higher the likelihood of tracking failure. Especially in crowded scenes, it will be affected by similar objects around. In this paper, the generalization ability of the tracker is increased using graph attention network-augmented features and combined with a track management mechanism. The outcomes in group e of Figure 9 further reveal the tracker's suitability for crowded scenarios.

Figure 8. Qualitative tracking outcomes of our method are presented, derived from the MOT17 dataset, captured using a mobile camera. In these representations, individual objects are differentiated by bounding boxes of diverse colors with their corresponding IDs denoted at the top-left corner. Groups (**a**–**e**) represent different scenes from video sequences MOT17-06, MOT17-07, MOT17-10, MOT17-12, and MOT17-14, and four randomly selected frames from each scene are used to demonstrate the tracking effect.

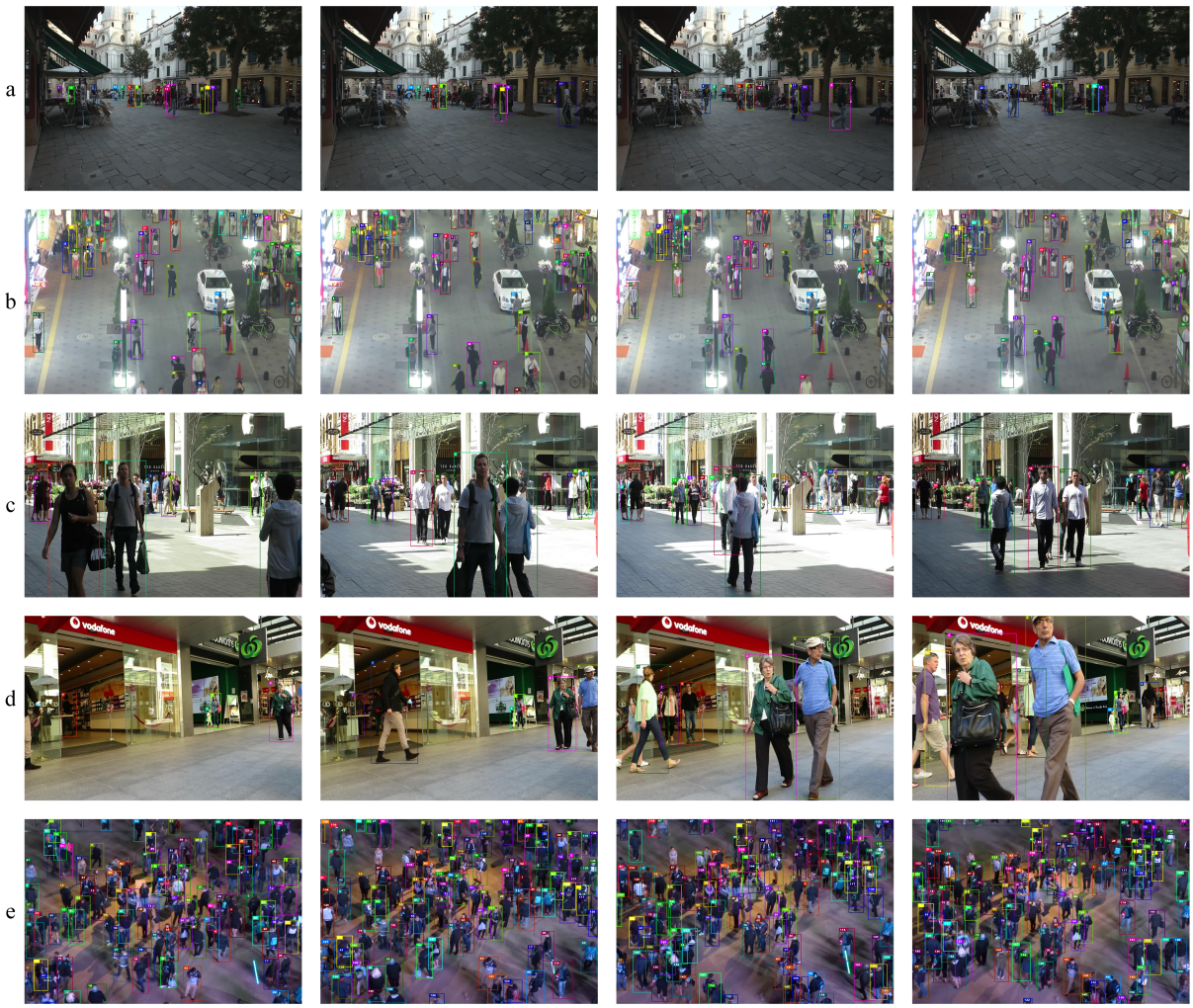


Figure 9. Qualitative tracking results of our method on MOT17 and MOT20 datasets by a static camera. In these representations, individual objects are differentiated by bounding boxes of diverse colors with their corresponding IDs denoted at the top-left corner. Groups (**a**–**e**) represent different scenes from video sequences MOT17-01, MOT17-03, MOT17-08, MOT17-09, and MOT20-04, and four randomly selected frames from each scene are used to demonstrate the tracking effect.

5. Conclusions

This paper presents a novel method for multiple object tracking that utilizes both graph attention networks and track management. The proposed method employs cross-attention and self-attention to selectively prioritize the features of beneficial nodes, thus enhancing the distinctness of node features and bolstering the discriminative capacity of the model. Simultaneously, we introduce a track management method that is designed to systematically control tracklet states, undertake tracklet creation, confirmation, termination, and manage occlusions. The online tracker is formulated by amalgamating the graph attention networks and track management, ensuring the maintenance of accuracy and robustness of the tracker across diverse scenarios. Comprehensive experiments on three MOT benchmark datasets (MOT16, MOT17, and MOT20) substantiate the precision and efficiency of the proposed method. In this study, a distinct appearance feature extractor network is used, which operates at a somewhat slower speed compared to joint trackers. Future work could incorporate the feature extractor network into the detector head, employing a joint detection and embedding approach.

Author Contributions: Methodology, Y.Z.; investigation, Y.L., A.E.; writing, review and editing, Z.W., C.W. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by GuangDong Basic and Applied Basic Research Foundation (No. 2022A1515110570), Elite Plan of Shandong University of Science and Technology (No. 0104060540508), and Innovation Teams of Youth Innovation in Science and Technology of High Education Institutions of Shandong province (No. 2021KJ088).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The publicly archived dataset is published at https://motchallenge.net/.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Wang, C.Y.; Bochkovskiy, A.; Liao, H.Y.M. YOLOv7: Trainable bag-of-freebies sets new state-of-the-art for real-time object detectors. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 7464–7475.
2. Jia, D.; Yuan, Y.; He, H.; Wu, X.; Yu, H.; Lin, W.; Sun, L.; Zhang, C.; Hu, H. Detrs with hybrid matching. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 19702–19712.
3. Wang, Z.; Li, M.; Lu, Y.; Bao, Y.; Li, Z.; Zhao, J. Effective multiple pedestrian tracking system in video surveillance with monocular stationary camera. *Expert Syst. Appl.* **2021**, *178*, 114992. [CrossRef]
4. Cao, J.; Pang, J.; Weng, X.; Khirodkar, R.; Kitani, K. Observation-centric sort: Rethinking sort for robust multi-object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Vancouver, BC, Canada, 17–24 June 2023; pp. 9686–9696.
5. Chen, J.; Wang, F.; Li, C.; Zhang, Y.; Ai, Y.; Zhang, W. Online multiple object tracking using a novel discriminative module for autonomous driving. *Electronics* **2021**, *10*, 2479. [CrossRef]
6. Kuhn, H.W. The Hungarian method for the assignment problem. *Nav. Res. Logist. Q.* **1955**, *2*, 83–97. [CrossRef]
7. Wojke, N.; Bewley, A.; Paulus, D. Simple online and realtime tracking with a deep association metric. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Beijing, China, 17–20 September 2017; pp. 3645–3649.
8. He, J.; Huang, Z.; Wang, N.; Zhang, Z. Learnable graph matching: Incorporating graph partitioning with deep feature learning for multiple object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 5299–5309.
9. Gao, J.; Zhang, T.; Xu, C. Graph convolutional tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 4649–4659.
10. Brasó, G.; Leal-Taixé, L. Learning a neural solver for multiple object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Seattle, WA, USA, 13–19 June 2020; pp. 6247–6257.
11. Shi, R.; Wang, C.; Zhao, G.; Xu, C. SCA-MMA: Spatial and Channel-Aware Multi-Modal Adaptation for Robust RGB-T Object Tracking. *Electronics* **2022**, *11*, 1820. [CrossRef]
12. Wang, Z.; Li, Z.; Leng, J.; Li, M.; Bai, L. Multiple Pedestrian Tracking With Graph Attention Map on Urban Road Scene. *IEEE Trans. Intell. Transp. Syst.* **2022**, *24*, 8567–8579. [CrossRef]
13. Wang, J.; Wei, Y.; Wu, X.; Huang, W.; Yu, L. Anti-Similar Visual Target Tracking Algorithm Based on Filter Peak Guidance and Fusion Network. *Electronics* **2023**, *12*, 2992. [CrossRef]
14. Gao, Y.; Gu, X.; Gao, Q.; Hou, R.; Hou, Y. TdmTracker: Multi-Object Tracker Guided by Trajectory Distribution Map. *Electronics* **2022**, *11*, 1010. [CrossRef]
15. Chu, Q.; Ouyang, W.; Li, H.; Wang, X.; Liu, B.; Yu, N. 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, Venice, Italy, 22–29 October 2017; pp. 4836–4845.
16. Zhu, J.; Yang, H.; Liu, N.; Kim, M.; Zhang, W.; Yang, M.H. Online multi-object tracking with dual matching attention networks. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 366–382.
17. Liu, J.; Wang, G.; Hu, P.; Duan, L.Y.; Kot, A.C. Global context-aware attention lstm networks for 3d action recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1647–1656.
18. Gao, G.; Gao, Y.; Xu, L.; Tan, H.; Tang, Y. DSGA: Distractor-Suppressing Graph Attention for Multi-object Tracking. In Proceedings of the 8th International Conference on Robotics and Artificial Intelligence, Singapore, 18–20 November 2022; pp. 69–76.
19. Jiang, Z.; Rahmani, H.; Angelov, P.; Black, S.; Williams, B.M. Graph-context attention networks for size-varied deep graph matching. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, New Orleans, LA, USA, 18–24 June 2022; pp. 2343–2352.

20. Woo, S.; Park, J.; Lee, J.Y.; Kweon, I.S. Cbam: Convolutional block attention module. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany, 8–14 September 2018; pp. 3–19.
21. Kipf, T.N.; Welling, M. Semi-supervised classification with graph convolutional networks. *arXiv* **2016**, arXiv:1609.02907.
22. Wang, R.; Yan, J.; Yang, X. Combinatorial learning of robust deep graph matching: An embedding based approach. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *45*, 6984–7000. [CrossRef] [PubMed]
23. Rezatofighi, H.; Tsoi, N.; Gwak, J.; Sadeghian, A.; Reid, I.; Savarese, S. Generalized intersection over union: A metric and a loss for bounding box regression. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 15–20 June 2019; pp. 658–666.
24. Lin, T.Y.; Goyal, P.; Girshick, R.; He, K.; Dollár, P. Focal loss for dense object detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 2980–2988.
25. Yu, T.; Wang, R.; Yan, J.; Li, B. Learning deep graph matching with channel-independent embedding and hungarian attention. In Proceedings of the International Conference on Learning Representations, Addis Ababa, Ethiopia, 26–30 April 2020.
26. Paszke, A.; Gross, S.; Massa, F.; Lerer, A.; Bradbury, J.; Chanan, G.; Killeen, T.; Lin, Z.; Gimelshein, N.; Antiga, L.; et al. Pytorch: An imperative style, high-performance deep learning library. In Proceedings of the Advances in Neural Information Processing Systems 32, Vancouver, BC, Canada, 8–14 December 2019.
27. Zhou, K.; Yang, Y.; Cavallaro, A.; Xiang, T. Omni-scale feature learning for person re-identification. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 3702–3712.
28. He, K.; Zhang, X.; Ren, S.; Sun, J. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27–30 June 2016; pp. 770–778.
29. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
30. Milan, A.; Leal-Taixé, L.; Reid, I.; Roth, S.; Schindler, K. MOT16: A benchmark for multi-object tracking. *arXiv* **2016**, arXiv:1603.00831.
31. Dendorfer, P.; Rezatofighi, H.; Milan, A.; Shi, J.; Cremers, D.; Reid, I.; Roth, S.; Schindler, K.; Leal-Taixé, L. Mot20: A benchmark for multi object tracking in crowded scenes. *arXiv* **2020**, arXiv:2003.09003
32. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. In Proceedings of the Advances in Neural Information Processing Systems 28, Montreal, QC, Canada, 7–12 December 2015.
33. Yang, F.; Choi, W.; Lin, Y. 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, Las Vegas, NV, USA, 27–30 June 2016; pp. 2129–2137.
34. Saleh, F.; Aliakbarian, S.; Rezatofighi, H.; Salzmann, M.; Gould, S. Probabilistic tracklet scoring and inpainting for multiple object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 14329–14339.
35. Kim, C.; Fuxin, L.; Alotaibi, M.; Rehg, J.M. Discriminative appearance modeling with multi-track pooling for real-time multi-object tracking. In Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, Nashville, TN, USA, 20–25 June 2021; pp. 9553–9562.
36. Liu, Q.; Chu, Q.; Liu, B.; Yu, N. GSM: Graph Similarity Model for Multi-Object Tracking. In Proceedings of the IJCAI, Yokohama, Japan, 7–15 January 2020; pp. 530–536.
37. Bergmann, P.; Meinhardt, T.; Leal-Taixe, L. Tracking without bells and whistles. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Seoul, Republic of Korea, 27 October–2 November 2019; pp. 941–951.
38. Zhou, X.; Koltun, V.; Krähenbühl, P. Tracking objects as points. In *Computer Vision—ECCV 2020, Proceedings of the 16th European Conference, Glasgow, UK, 23–28 August 2020*; Proceedings, Part IV; Springer: Berlin/Heidelberg, Germany, 2020; pp. 474–490.
39. Jonathon Luiten, A.H. TrackEval. 2020. Available online: https://github.com/JonathonLuiten/TrackEval (accessed on 7 July 2022).
40. Bernardin, K.; Stiefelhagen, R. Evaluating multiple object tracking performance: The clear mot metrics. *EURASIP J. Image Video Process.* **2008**, *2008*, 246309. [CrossRef]
41. Luiten, J.; Osep, A.; Dendorfer, P.; Torr, P.; Geiger, A.; Leal-Taixé, L.; Leibe, B. Hota: A higher order metric for evaluating multi-object tracking. *Int. J. Comput. Vis.* **2021**, *129*, 548–578. [CrossRef] [PubMed]
42. Ristani, E.; Solera, F.; Zou, R.; Cucchiara, R.; Tomasi, C. Performance measures and a data set for multi-target, multi-camera tracking. In Proceedings of the Computer Vision—ECCV 2016 Workshops, Amsterdam, The Netherlands, 8–10 and 15–16 October 2016; Proceedings, Part II; Springer: Berlin/Heidelberg, Germany, 2016; pp. 17–35.
43. Li, Y.; Huang, C.; Nevatia, R. Learning to associate: Hybridboosted multi-target tracker for crowded scene. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Miami, FL, USA, 20–25 June 2009; pp. 2953–2960.
44. Baisa, N.L. Robust online multi-target visual tracking using a HISP filter with discriminative deep appearance learning. *J. Vis. Commun. Image Represent.* **2021**, *77*, 102952. [CrossRef]
45. Baisa, N.L. Occlusion-robust online multi-object visual tracking using a GM-PHD filter with CNN-based re-identification. *J. Vis. Commun. Image Represent.* **2021**, *80*, 103279. [CrossRef]
46. Yoon, Y.C.; Kim, D.Y.; Song, Y.M.; Yoon, K.; Jeon, M. Online multiple pedestrians tracking using deep temporal appearance matching association. *Inf. Sci.* **2021**, *561*, 326–351. [CrossRef]

47. Psalta, A.; Tsironis, V.; Karantzalos, K. Transformer-based assignment decision network for multiple object tracking. *arXiv* **2022**, arXiv:2208.03571.
48. Lohn-Jaramillo, J.; Ray, L.; Granger, R.; Bowen, E. Clustertracker: An Efficiency-Focused Multiple Object Tracking Method. 2022. Available online: https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4102945 (accessed on 16 July 2023).
49. Urbann, O.; Bredtmann, O.; Otten, M.; Richter, J.P.; Bauer, T.; Zibriczky, D. Online and real-time tracking in a surveillance scenario. *arXiv* **2021**, arXiv:2106.01153.