Machine Vision and Applications (2024) 35:54
https://doi.org/10.1007/s00138-024-01520-8

ORIGINAL PAPER

PTDS CenterTrack: pedestrian tracking in dense scenes with re-identification and feature enhancement

Jiazheng Wen[1] · Huanyu Liu[1] · Junbao Li[1]

Received: 26 March 2023 / Revised: 23 December 2023 / Accepted: 6 February 2024 / Published online: 15 April 2024

Abstract

Multi-object tracking in dense scenes has always been a major difficulty in this field. Although some existing algorithms achieve excellent results in multi-object tracking, they fail to achieve good generalization when the application background is transferred to more challenging dense scenarios. In this work, we propose PTDS(Pedestrian Tracking in Dense Scene) CenterTrack based on the CenterTrack for object center point detection and tracking. It utilizes dense inter-frame similarity to perform object appearance feature comparisons to predict the inter-frame position changes of objects, extending CenterTrack by using only motion features. We propose a feature enhancement method based on a hybrid attention mechanism, which adds information on the temporal dimension between frames to the features required for object detection, and connects the two tasks of detection and tracking. Under the MOT20 benchmark, PTDS CenterTrack has achieved 55.6%MOTA, 55.1%IDF1, 45.1%HOTA, which is an increase of 10.1 percentage points, 4.0 percentage points, and 4.8 percentage points respectively compared to CenterTrack.

Keywords Multi-object tracking · Joint detection tracking paradigm · Feature fusion · Dense pedestrian detection · Anchor-free detection

1 Introduction

Multiple Object Tracking (MOT), as a mid-level computer vision task, is the basis for many high-level tasks such as pose estimation [1], action recognition [18], and behavior analysis [28]. It aims to locate the trajectories of multiple objects consecutively in video frames. Meanwhile, compared with single-object tracking, multi-object tracking not only faces challenges in the number of objects, but also makes original identity preservation more difficult due to frequent occlusions, similar appearances, and interactions among multiple objects.

In multi-object tracking tasks, pedestrians are usually the center of attention in video scenes, which makes detecting and tracking them a fundamental problem that needs to be studied in computer vision. Furthermore, compared with other visual objects, pedestrians, as typical non-rigid objects, are ideal samples to study multi-object tracking problems [14]. However, the complexity of this task depends on occlusion, erratic motion, and visual similarity of the pedestrians to be tracked, and remains an open area of research [32]. As the situation of large-scale dense pedestrians becomes more and more common, due to the sudden increase of object density, trackers not only face challenges in object detection but also the occurrence of identity transitions in the trajectory generation process that is becoming more and more frequent, as shown in Fig. 1. The vast majority of existing methods [46, 48, 50, 59] do not specifically focus on the pedestrian tracking problem in dense scenarios, so when these methods are transferred to such scenarios, they do not achieve good generalization.

CenterTrack [59], based on the anchor-free keypoint-based object detection network CenterNet [57], exhibits an excellent balance in tracking performance and training/inference cost, making it suitable for a wide range of application scenarios. Specifically, CenterNet, as a detection network, models the category and location of an object

✉ Huanyu Liu
liuhuanyu@hit.edu.cn

Jiazheng Wen
22b903087@stu.hit.edu.cn

Junbao Li
lijunbao@hit.edu.cn

[1] Faculty of Computing, Harbin Institute of Technology, No.2, Yikuang Street, Harbin 150080, Heilongjiang, People's Republic of China

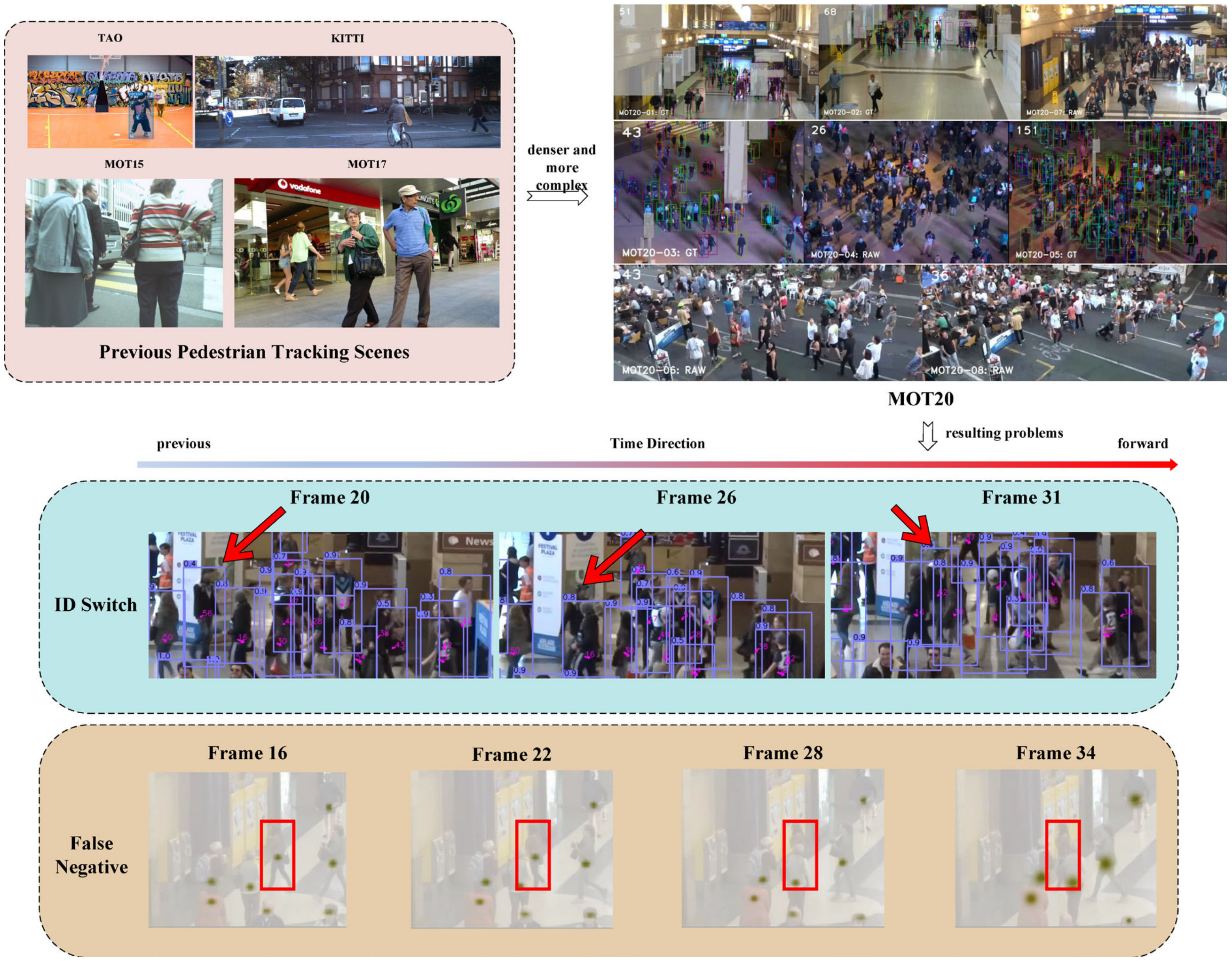


Fig. 1 We compare human tracking in different density scenarios, including TAO [13], KITTI [16], MOT15 [23], MOT17 [29]. Moreover, after the scene becomes more complex, more identity switches and missed detection problems will arise

by capturing the object's heat map, size, and center point position error. To associate objects across frames, CenterTrack incorporates an inter-frame displacement regression branch into the regression head of CenterNet to predict object position changes in the time dimension. This structural simplification streamlines two key steps of detection-based tracking schemes. First, in object detection, historical information of each object from the previous frame is included in the corresponding heat map, enabling the tracking model to directly access relevant information within the cluster. Second, the data association method establishes connections between the same objects in the previous and current frames through predicted displacement vectors. However, in dense scenes, locating an object's center point becomes more complex, and the heat map rendering method inherited from CornerNet [22] may lack robustness. Additionally, simple adjacent frame displacement prediction struggles to handle frequent intersections and occlusions, leading to identity switching and object trajectory mismatch problems. Despite being a multi-object tracking algorithm under the joint detection and tracking paradigm, CenterTrack's tracking effectiveness still heavily relies on the detection model [10].

In this paper, to solve the above-mentioned specific problems, we propose a network for pedestrian tracking in dense scenes using re-identification features and feature enhancement methods, named PTDS(Pedestrian Tracking in Dense Scene) CenterTrack, which is designed based on CenterTrack. We first improved the detection method of key points in heatmaps and redesigned the center point rendering method originally inherited from CornerNet to obtain more accurate positioning for object detection in dense scenes. Then, we design a simple re-identification feature extraction network, which extends the original tracking method in Centertrack that only uses the object displacement between frames for data association. After that, we design a deformable convolution-based feature enhancement module inspired by the pose estimation network PoseWarper [5] and combine it with the attention mechanism of spatiotemporal-level fusion. This module converts the feature differences of adjacent frames into the offset of each element for feeding the deformable convolution kernel through a designed hybrid attention mechanism network and then uses deformable convolution to extract features from adjacent frames. Finally, we superimpose the extracted features on the current frame features for feature enhancement.

We summarize the contributions as below:

1. We propose a novel heatmap acquisition method to improve object center point detection, enhancing the model's accuracy in locating object positions during training.
2. We develop a low-computing power re-identification feature extraction network that utilizes a method similar to TraDeS [48] to obtain pixel-level information between frames and inter-frame displacement of objects.
3. We introduce a feature enhancement method based on a hybrid attention mechanism, integrating temporal dimension information between video frames into object detection features.

2 Related works

Most of the excellent multi-object tracking methods follow a detection-based tracking paradigm, which first detects objects in each frame and then associates objects with identities. We classify existing methods into two categories by whether the network models both detection and tracking tasks simultaneously. We discuss the methods under these two different paradigms and compare them with our proposed network. In addition, since we also focus on object detection in dense scenes and the application of the attention mechanism in feature enhancement, the object detection and attention mechanism used in multi-object tracking are also discussed.

2.1 Keypoint-based object detection

From the perspective of encoding object category and location, the object detection algorithms based on deep convolutional neural networks can be roughly divided into two categories: anchor-based [11, 15, 17, 26, 34, 36, 51] and anchor-free [21, 22, 41, 57, 58] methods. The detection network used in this paper is mainly the object detection algorithm based on key points in an anchor-free style. Thus, the algorithm is transformed into a standard key point estimation problem. According to the spatial layout of the key points, two categories emerge for this type of algorithm: edge-point-based and center-point-based object detection models. It is worth noting that some detectors also apply both contour and center points to the same detection model. Among them, CenterNet used a simple but effective method to model an object as the center point of a bounding box. Through the object size and the center-point offset obtained by regression, CenterNet has become more widely applicable to a variety of other tasks, such as pose estimation and 3D object detection.

2.2 Separate model for detection and tracking

In recent years, detection-based tracking [6, 7, 38, 45, 47, 52] has been the mainstream method in MOT. The detection-based tracking method advocates that: firstly, the existing detection model is used to generate the detection results under each frame; after that, an additional re-identification model is used to extract the appearance features of each detection or a motion model is used to directly predict the inter-frame motion state of objects; finally, the correlation matching algorithm is used to complete the data association step, and the complete trajectory result is obtained. In the detection-based tracking paradigm, there are a large number of multi-object tracking algorithms that apply probabilistic inference models. In the case of linear systems and Gaussian distributed object states, the Kalman filter proved to be the best estimator [35]. SORT [6] uses a Kalman filter model to track the bounding box of each object and associates each bounding box with the largest overlap in the current frame through a binary matching algorithm. DeepSORT [47] utilizes deep convolutional neural networks to extract the appearance features of each tracked object to enhance coverage-based association methods in SORT. In addition, in the detection-based tracking paradigm, more and more algorithms have started to focus on the robustness of data association. Schulter et al. [38] pointed out that by representing the matching optimization problem as a differentiable function for backpropagation, the model can learn features related to data associations. There are also methods [7, 45, 52] that treat each detection result as a node in a directed graph and describe the basic data association problem as a graph optimization problem with a cost vector.

2.3 Joint model for detection and tracking

The methods of joint detection and tracking [3, 30, 33, 46, 54, 56, 59] are rapidly emerging in this field due to their advanced structure, and its re-examination of the relationship between detection and tracking models which have extremely high academic value for solving joint optimization. They integrate the originally completely separated detection and tracking tasks into the same framework by transforming the structure of the existing detection model or inserting it into the tracking model. Zhang et al. [56] leveraged a bounding box-based binary matching algorithm for data association, using the tracked object bounding boxes as additional region proposals to enhance the model's detection ability. Tracktor [3], as the foundation of the joint detection and tracking method, is a representative algorithm linking the two methods. Tracktor uses bounding box regression to directly deliver region proposals containing object identity information, thereby removing bounding box associations in previous methods, greatly promoting the development of joint detection and tracking methods. CTracker [33] concatenates pairwise bounding box regression results to predict object trajectories. In online tracking, researchers have also made a lot of effort. JDE [46] added the re-identification branch to the single-step detection model YOLOv3. By combining the detection model and the tracking model, the computational cost of the re-identification model was greatly reduced. CenterTrack [59] adds an inter-frame object displacement regression branch to the keypoint detection model CenterNet and directly obtains the correlation information of inter-frame objects through simple and effective means. FairMOT [54] demonstrated the importance of detection and recognition tasks for tracking, using an anchor-free method to disambiguate anchor boxes in extracting region proposals.

Although the detection-based tracking method has always been the mainstream method in multi-object tracking, it has two main drawbacks: 1) The separation of the two parts: detection and tracking is not conducive to the joint optimization of the model, and the optimization directions of the two parts of the model are inconsistent, and eventually, the overall model cannot obtain the optimal result globally; 2) To provide an optimization basis for the data association steps, the re-identification models used in such methods are independent and require high computational costs, which greatly limits the real-time performance of multi-object tracking algorithms. Compared with the detection-based multi-object tracking paradigm, the multi-object tracking algorithm based on the joint detection and tracking paradigm has better prospects in both theoretical research and practical application due to its advanced structure and tracking speed.

2.4 Attention mechanism

The hybrid attention mechanism method we adopted is different from the very popular Transformer [42] and self-attention mechanism [40, 44, 49, 50]. However, it is undeniable that the Transformer structure has achieved great success in the field of natural language processing and has been applied to computer vision-related tasks to capture longer-range dependencies. TransCenter [50] uses dense pixel-level multi-scale queries in the Transformer dual-decoder network to globally and robustly infer heatmaps of object centers and correlate them temporally. TransTrack [40] takes an object query as input to provide common object detection results and leverages feature from previously detected objects to form another "track query" to discover associated objects on the following frames. Different from these methods, we mainly encode differential features through local correlation weights as well as channel-adaptive weights and use these cues to improve the robustness of the feature enhancement module.

3 PTDS CenterTrack

3.1 Overview

In the domain of multi-object tracking, the key processes involve detection and re-identification. Detection is utilized to ascertain the location of objects, while re-identification is employed to link the same identity object across a sequence. Consequently, multi-object tracking networks encompass the acquisition and integration of temporal and spatial features, while simultaneously engaging in multi-task learning of detection and re-identification. Given the direct impact of detection on tracking, these two processes are typically conducted sequentially. However, while studying more efficient and accurate tracking, this paper also explores feeding back the part of the tracking information that is beneficial to detection to the network.

The overall network structure we designed is shown in Fig. 2, which can be divided into four main parts, namely the basic feature extraction part, the inter-frame displacement prediction part, the feature enhancement part, and the data association part. Firstly, in the basic feature extraction part, we use the backbone network part of CenterNet to obtain the basic features of video frames. We elaborate on the proposed method of obtaining heat maps in Sect. 3.2 and apply it in the basic feature extraction part. Secondly, we removed the original inter-frame displacement prediction part in CenterTrack and used our proposed re-identification feature extraction module for basic feature transformation. Then, the cost matrix is constructed based on different similarity measurement methods, and the displacement prediction in the horizontal and vertical directions is carried out through the designed displacement template. We describe the inter-frame displacement prediction part in detail in Sect. 3.3. Thirdly, the feature enhancement part exists as a way to transform the inter-frame displacement information in tracking into features beneficial for the detection task. We adopt a method based on a hybrid attention mechanism for implementation, which is detailed in Sect. 3.4. Finally, we discuss different data association methods in Sect. 4.4.4.

3.2 Object center point detection in dense scenes

In CenterNet [57], the method for representing the detection as the object center point can be summarized as an input

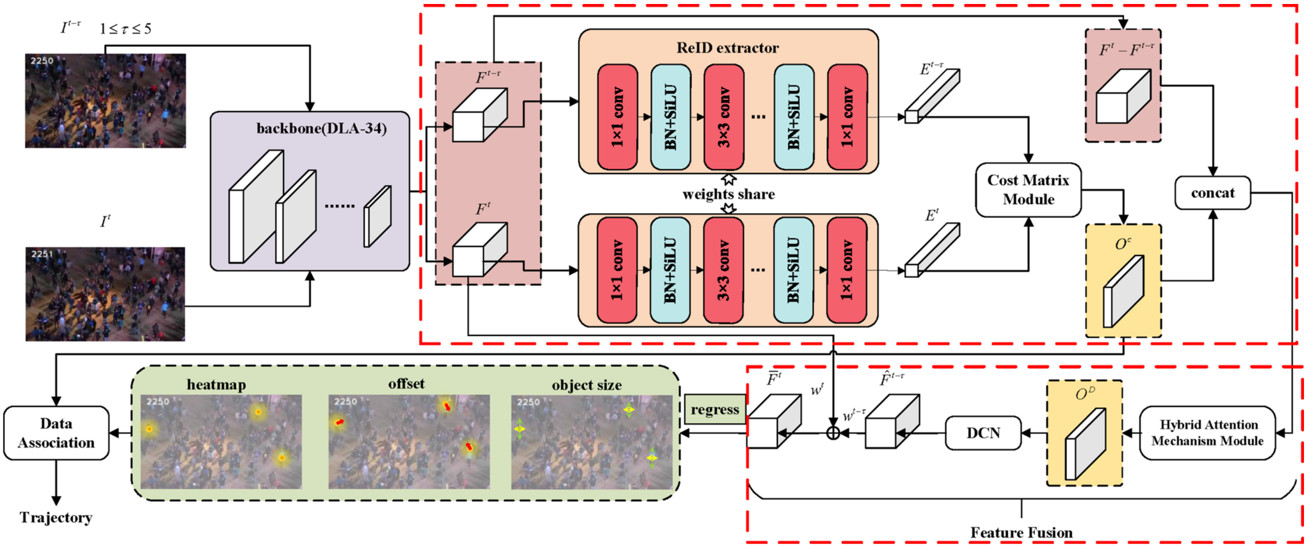


Fig. 2 The overall network structure of PTDS CenterTrack. It can be divided into 4 parts: a) basic feature extraction; b) inter-frame displacement prediction; c) feature enhancement; d) data association. The location marked by the red box is the main improvement with CenterTrack, which is explained in detail later

image $\mathbf{I} \in \mathbb{R}^{W \times H \times 3}$ mapped to key points on a heat map $\hat{\mathbf{Y}} \in [0, 1]^{(W/R)\times(H/R)\times C}$, where W is the input width, H is the input height, the number of channels is 3, R is the output size scaling ratio, and C is the number of key point types. During the training process, the ground-truth key points are transformed into a probability distribution through a binary Gaussian kernel function:

$$\mathbf{Y}_{xyc} = exp\left[-\frac{(x-k_x)^2 + (y-k_y)^2}{2\sigma_k^2}\right] \tag{1}$$

and scattered on the heat map $\mathbf{Y} \in [0, 1]^{(W/R)\times(H/R)\times C}$, where $(x, y) \in \mathbb{R}^2$ is the coordinate value of the key point on the heat map, $(k_x, k_y) \in \mathbb{R}^2$ is the ground truth coordinate value of the key point, and σ_k is the standard deviation of the Gaussian kernel function.

To ensure that the difference between the ground truth of the heat map and the annotated ground truth remains moderate, it is necessary to add a constraint on the IoU threshold between them. According to the nature of the Gaussian distribution, we set three times the standard deviation $3\sigma_k$ under the Gaussian distribution to describe the effective radius of the circular area occupied by this probability distribution on the heat map. According to the analysis of this constraint relationship for the situations described by [1] and [57], we considered three corner situations. On this basis, we propose a new enhanced method for generating the effective radius.

$$\begin{cases} r_1 = \left[(H+W) + \sqrt{(H+W)^2 - 4HW\frac{1-\mathbf{IoU}_{th}}{1+\mathbf{IoU}_{th}}}\right]/2 \\ r_2 = (H+W) + \sqrt{(H+W)^2 - 4HW(1-\mathbf{IoU}_{th})} \\ r_3 = -(H+W)\mathbf{IoU}_{th} + \sqrt{(H-W)^2\mathbf{IoU}_{th}^2 + 4HW\mathbf{IoU}_{th}} \end{cases} \tag{2}$$

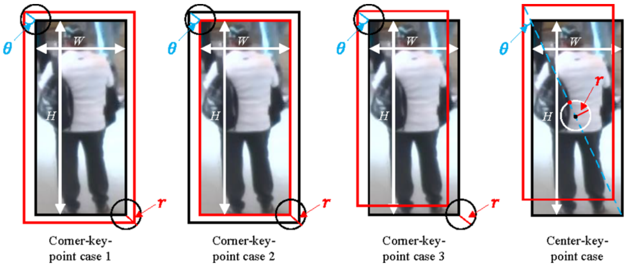


Fig. 3 Four cases where the ground-truth bounding boxes of the heat map overlap with the ones of the annotation. Corner-key-point case 1: the ground-truth bounding box of the heat map(red, the same below) covers one of the annotations (black, the same below). Corner-key-point case 2: the ground-truth bounding box of the heat map is contained in the annotated one. Corner-key-point case 3: the ground-truth bounding box of the heat map overlaps with the annotated one. Center-key-point case: including the above three cases. For the purpose of derivation, only the case where the constraint is equal to the threshold is shown

In Fig. 3, the black bounding boxes denote the annotated bounding boxes and the key points on the heat map are denoted by a circular probability distribution generated with the effective radius r. The ground-truth bounding boxes of the heat map are denoted by red bounding boxes. The red and black bounding boxes are satisfied with the constraint on the IoU threshold. Overall, the following formula can be obtained:

$$\frac{S_{inter}}{S_{union}} \le \mathbf{IoU}_{th} \tag{3}$$

where S_{inter} is the area occupied by the intersection of the red and black bounding boxes, S_{union} is the area occupied by the union of them, and $\mathbf{IoU}_{th}$ is the constrained IoU thresh-

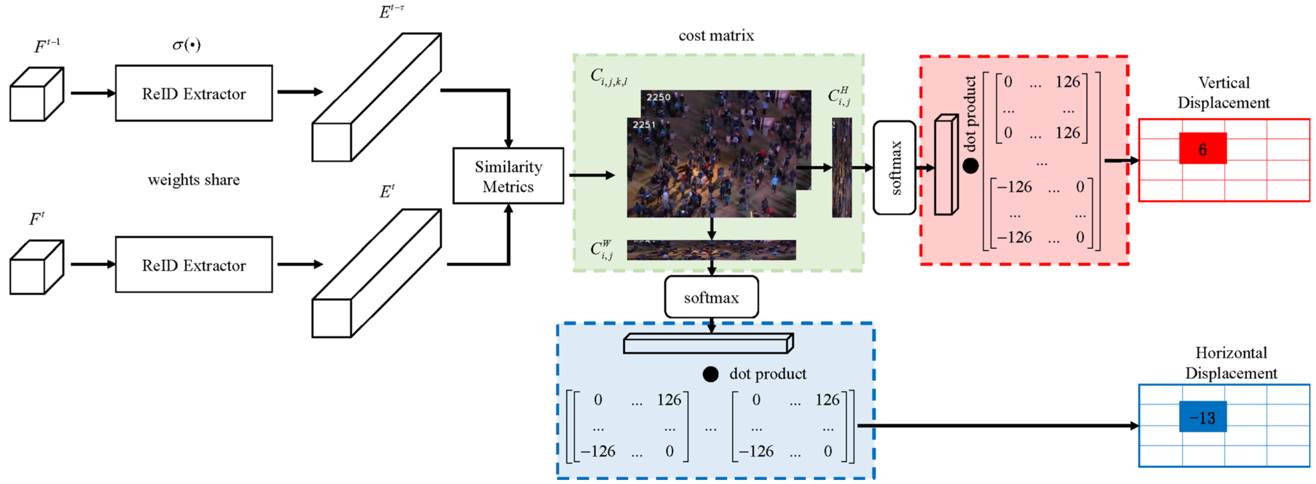


Fig. 4 The inter-frame displacement prediction module. We adopt a similar approach to TraDeS [48] and replace the inter-frame displacement regression branch in CenterTrack by re-identifying the embedding model to obtain inter-frame displacement prediction. However, we use improved Re-ID feature extractors and different cost matrix calculation methods

old. Based on this constraint and the description of [1, 57], the three cases of effective-radius formulas are described in Eq. (2).

However, we adopted a different generation method to further simplify this problem from three cases to one. In Fig. 3, we indicate a case in which the effective radius is generated when the key points are reduced to only one point, and the following formula can be obtained:

$$S_1 = (W - r\sin\theta)\cdot(H - r\cos\theta) \tag{4}$$

$$S_2 = W\cdot H - S_1 \tag{5}$$

$$\frac{S_1}{S_1 + 2S_2} \le \mathbf{IoU}_{th} \tag{6}$$

where S_1 is the area occupied by the intersection of the red and black bounding boxes, S_2 is the area occupied by their union, and θ is their offset angle. When $\sin\theta = W/(\sqrt{W^2+H^2})$ and $\cos\theta = H/(\sqrt{W^2+H^2})$, Eq. (6) becomes an identity, and the effective radius r_{center} is obtained from Eq. (7). The advantages of the proposed generation method are experimentally explained in Sec. 4.

$$r_{center} = \left(1 - \sqrt{\frac{2\mathbf{IoU}_{th}}{1+\mathbf{IoU}_{th}}}\right)\cdot\sqrt{W^2+H^2} \tag{7}$$

3.3 Object inter-frame displacement prediction based on cost matrix

During tracking, objects often undergo identity changes due to occlusion or significant alterations in appearance. Directly initiating a new trajectory from these objects can result in a plethora of fragments and identity switches. The utilization of re-identification embedding not only aids in distinguishing similar objects but also establishes a feature repository for them, laying the groundwork for the continuation of trajectories when occluded objects re-emerge.

The re-identification feature extraction approach proposed in this paper diverges from the traditional method of acquiring local features for analysis and comparison. Instead, we construct a high-dimensional embedding model to delineate distinctions between different individuals within the same class. Initially, we persist with using the fundamental appearance features obtained by the DLA-34 backbone network layer in CenterTrack as input for our network layer. Subsequently, we devise a lightweight re-identification feature extraction module that interfaces with the backbone network layer to transform appearance features into high-dimensional intra-class distinguishing features. The specific network structure is depicted in Fig. 4, and the corresponding formula for expressing this content is as follows:

$$\mathbf{E}^t = \sigma(\mathbf{F}^t) \tag{8}$$

where $\mathbf{F}^t$ represents the feature obtained after the t-th frame image is extracted by the backbone network layer, $\mathbf{E}^t$ represents the re-identification embedded feature of the t-th frame image, $\sigma(\cdot)$ represents the mapping corresponding to the re-identification embedded model extraction network in Fig. 5. In Fig. 5, compared with the general re-identification feature extraction network, the proposed network removes the maximum pooling layer, keeps the size of the input features unchanged, and only increases the number of input channels. This is because the output of the re-identification feature extractor on the left is upsampled to the size of the input for subsequent processing. The method proposed in this article

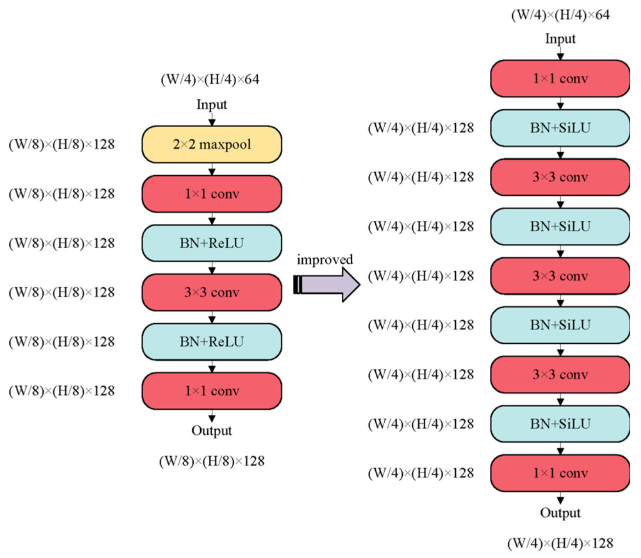


Fig. 5 The proposed re-identification feature extraction network. The network on the left of the figure is a general re-identification feature extractor, and the right is the improved version proposed in this paper

does not require upsampling operations and can be directly used to obtain the cost matrix, which weakens the transmission of errors in the network. In addition, the original ReLU non-linear activation unit is replaced by SiLU, and additional convolutional layers are added, to extract more accurate re-identification information in dense object scenes.

Further, we adopt a similar approach to TraDeS [48], by re-identifying the embedding model to obtain the inter-frame displacement prediction to replace the inter-frame displacement regression branch in CenterTrack. In scenes with dense pedestrians, the dense pixel displacement prediction method can more accurately capture the running direction and displacement distance of objects. We perform a correlation operation between the current frame part $\mathbf{E}^t$ and the previous frame part $\mathbf{E}^{t-\tau}$ of the extracted re-identification embedding model. Unlike TraDeS, we use three different calculation methods, which are used to judge the correlation of each corresponding pixel point. These calculation methods are based on vector inner product distance, vector cosine distance normalized by feature channel, and vector Euclidean distance, respectively, and their specific mathematical expressions are as follows:

$$\mathbf{C}_{i,j,k,l} = \mathbf{E}^t_{i,j} \cdot {\mathbf{E}^{t-\tau}_{k,l}}^T \tag{9}$$

$$\mathbf{C}_{i,j,k,l} = \frac{\mathbf{E}^t_{i,j} \cdot {\mathbf{E}^{t-\tau}_{k,l}}^T}{Norm_{L2}(\mathbf{E}^t_{i,j}) \cdot Norm_{L2}(\mathbf{E}^{t-\tau}_{k,l})^T} \tag{10}$$

$$\mathbf{C}_{i,j,k,l} = \sqrt{(\mathbf{E}^t_{i,j})^2 + (\mathbf{E}^{t-\tau}_{k,l})^2 - 2\mathbf{E}^t_{i,j} \cdot {\mathbf{E}^{t-\tau}_{k,l}}^T} \tag{11}$$

where $\mathbf{C}_{i,j,k,l}$ is the cost matrix, $\mathbf{E}^t_{i,j}$ is the embedding multi-dimensional matrix of the frame t, (i, j) is the abscissa index and ordinate index of $\mathbf{E}_{i,j}$ respectively, ${\mathbf{E}^{t-\tau}_{k,l}}$ is the embedded multi-dimensional matrix of the frame $t - \tau$, (k, l) is the abscissa index and ordinate index of $\mathbf{E}_{k,l}$, respectively, $(\cdot)^T$ is the matrix transpose operation.

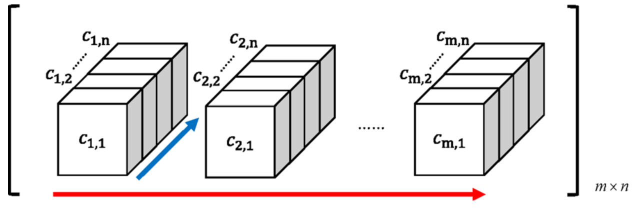


Fig. 6 The cost matrix. It can be understood as a multi-layer nested tensor, consisting of $m \times n$ tensors of size mn. This tensor is calculated from inter-frame re-identification features and used for displacement prediction

Equation (9) corresponds to the method of calculating the cost matrix by using the vector inner product; Eq. (10) corresponds to the method of the vector cosine distance normalized by the feature channel, where $Norm_{L2}(\cdot)$ is the characteristic L2 norm calculation in the channel direction; Eq. (11) corresponds to the method of the vector Euclidean distance, where $(\cdot)^2$ is the square operation at the matrix element level, not the multiplication operation of the matrix itself.

Based on the above analysis, a cost matrix (tensor form) of size $((H/8), (W/8), (H/8), (W/8))$ can be obtained. As shown in Fig. 6, it can be understood that each element in the $m \times n$-dimensional matrix is also an $m \times n$-dimensional matrix, where $m = H/8$ and $n = W/8$. Each matrix unit in Fig. 6, such as $c_{1,1}$, represents the similarity between the pixel at the $(1, 1)$ coordinate position of the current frame and all the pixels at the previous frame, so it has a 2-dimensional attribute. The other two dimensions are represented by blue arrows and red arrows, respectively. The blue arrow is the third dimension of the cost matrix, which represents the traversal of the current frame to the previous frame in the horizontal direction. The red arrow is the fourth dimension of the quantity matrix, which represents the traversal of the current frame to find the similarity of the previous frame in the vertical direction.

At last, we follow the processing method of the cost matrix in TraDeS, first convert it into a horizontal difference maximum vector and a vertical difference maximum vector, to predict the horizontal displacement and vertical displacement of each pixel between frames, and then convert these vectors into probability representations through the softmax function, and finally, the actual horizontal and vertical displacements of each corresponding pixel between frames are obtained through the designed displacement template. Taking the input image size of 512 × 512 as an example, the size of the feature map obtained after the feature extraction of the backbone network is 64 × 64.

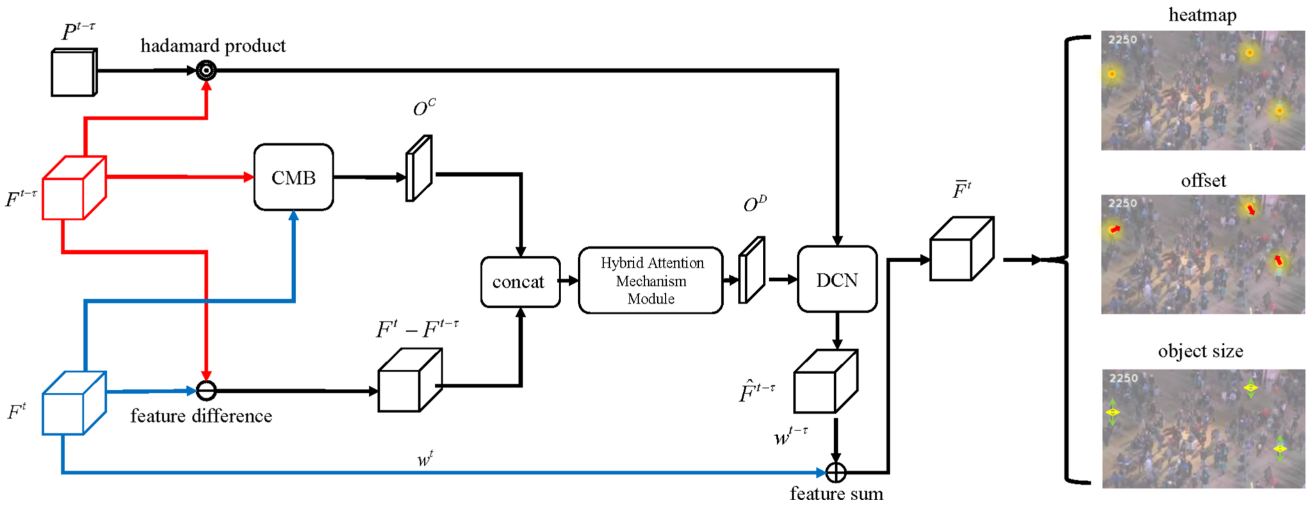


Fig. 7 The feature fusion module. The figure mainly shows the process of obtaining enhanced features through the feature fusion network based on the hybrid attention mechanism after combining the inter-frame displacement information and the feature frame difference. The detailed process can be referred to as Algorithm 1

3.4 Deformable convolutional feature fusion based on hybrid attention mechanism

The foundation of multi-object tracking lies in multi-object detection, yet detection is typically conducted in isolation without considering tracking cues. As previously mentioned, we contend that stable and consistent trajectories hinge on robust detection, and conversely, tracking cues significantly benefit detection, particularly in scenarios characterized by frequent occlusions stemming from an upsurge in object density. The conventional re-ID tracking loss does not align with the detection loss incurred from jointly training a single backbone network, and in some cases, it may even impede detection performance to a certain extent [8]. This is because re-ID prioritizes intra-class variance, whereas detection is geared towards accentuating inter-class disparities and minimizing intra-class variance.

To better balance the two tasks in one network, and use the predicted displacement as motion information to guide the detection module, inspired by the pose estimation network PoseWarper [5], based on the deformable convolution theory [12], we apply the object displacement between frames to the feature maps of the previous frames and perform convolution operations on them. The obtained new feature map based on the previous frame and the current frame feature map are adaptively weighted and summed as the input feature of regression branches. The overall structure is shown in Fig. 7.

The composition of deformable convolution includes two elements: first, the element value of the original convolution kernel; second, the offset of each pixel in the input feature map, which includes the horizontal direction and the vertical direction. According to the principle of deformable convolution, it is necessary to generate $2k^2$ offset values for each pixel position of the input feature map, where k is the size of the convolution kernel. Therefore, it is necessary to complete the mapping process $\gamma(\cdot)$ of shifting the frame to $2k^2$ offset values, and we can directly perform the difference operation on the equal-sized feature maps between the previous frame and the current frame to obtain the residual feature maps. Taking them together with the frame-to-frame displacement of objects as mapping input allows the model to utilize more spatio-temporal motion information.

We use the hybrid attention mechanism module to design this mapping network, the specific network structure is shown in Fig. 8. We add a channel attention module and a spatial attention module to the input downsampling residual block and the output residual block, respectively. Through the channel attention mechanism module, adaptive learning is performed on the input residual features between frames and the importance of each channel of the object displacement between frames to obtain more effective spatio-temporal information for detection enhancement. Through the spatial attention mechanism module, the attention points of the local motion information of the inter-frame residual feature maps are adaptively learned to obtain more effective inter-frame difference features.

We add the integrated features of the previous frame and the basic features of the current frame through the adaptive weight matrix, the specific form as follows:

$$\hat{\mathbf{F}}^{t-\tau} = \mathbf{F}^{t-\tau} \odot \mathbf{P}^{t-\tau} \tag{12}$$

where $\mathbf{F}^{t-\tau}$ is the base feature extracted by backbone layers for frame $t - \tau$, $\mathbf{P}^{t-\tau} \in \mathbb{R}^{(W/4)\times(H/4)\times 1}$ is the heatmap

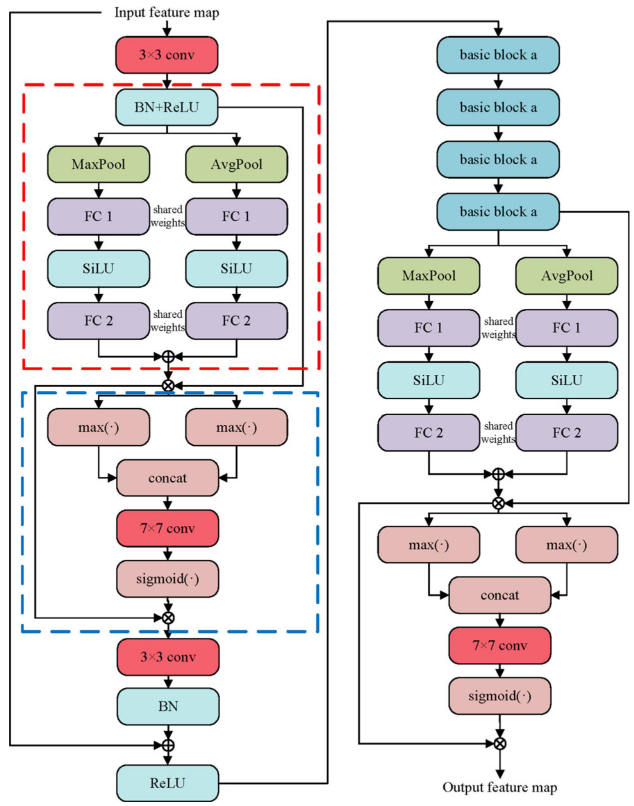


Fig. 8 Feature fusion network based on a hybrid attention mechanism. The red dotted box is the spatial attention mechanism, and the blue dotted box is the channel attention mechanism

Algorithm 1 Deformable Conventional Feature Fusion

```
Require: $\mathbf{F}^{t-\tau}$ is the base feature for frame $t-\tau$; $\mathbf{F}^{t}$ is the base feature for frame $t$; $\mathbf{P}^{t-\tau}$ is the heatmap for frame $t-\tau$; $CMB(\mathbf{A},\mathbf{B})$ is the cost matrix calculation function of feature $\mathbf{A}$ and feature $\mathbf{B}$; $concat(\mathbf{A},\mathbf{B})$ is the feature splicing function; $HAM(\cdot)$ is the feature fusion network based on a hybrid attention mechanism; $DCN(\cdot)$ is the deformable convolutional network.
Ensure: $\bar{\mathbf{F}}^{t}$ is the enhanced features.
for $\tau = 1 : T$ do
    $\mathbf{O}^{C} \Leftarrow CMB(\mathbf{F}^{t-\tau}, \mathbf{F}^{t})$
    $\bar{\mathbf{O}}^{D} \Leftarrow concat(\mathbf{O}^{C}, \mathbf{F}^{t} - \mathbf{F}^{t-\tau})$
    $\mathbf{O}^{D} \Leftarrow HAM(\bar{\mathbf{O}}^{D})$
    $\hat{\mathbf{F}}^{t-\tau} \Leftarrow DCN(\mathbf{P}^{t-\tau} \odot \mathbf{F}^{t-\tau}, \mathbf{O}^{D})$
end for
$\bar{\mathbf{F}}^{t} \Leftarrow \mathbf{w}^{t} \odot \mathbf{F}^{t} + \sum_{\tau=1}^{T} \mathbf{w}^{t-\tau} \odot \hat{\mathbf{F}}^{t-\tau}$
```

obtained by the detection model for frame $t-\tau$, under the problem of this paper, it is only for the classification of pedestrians, $\hat{\mathbf{F}}^{t-\tau} \in \mathbb{R}^{(W/4)\times(H/4)\times 64}$ is the result of the channel-by-channel and pixel-by-pixel overlay of $\mathbf{F}^{t-\tau}$ and $\mathbf{P}^{t-\tau}$. In addition, $\odot$ is the Hadamard product of matrices. The feature enhancement part is to add the integrated features of the previous frame and the basic features of the current frame through the adaptive weight matrix, the specific form is as follows:

$$\bar{\mathbf{F}}^{t} = \mathbf{w}^{t} \odot \mathbf{F}^{t} + \sum_{\tau=1}^{T} \mathbf{w}^{t-\tau} \odot \hat{\mathbf{F}}^{t-\tau} \tag{13}$$

where, $\mathbf{w}^{t} \in \mathbb{R}^{(W/4)\times(H/4)\times 1}$ is the adaptive weight matrix for the current frame, $\mathbf{w}^{t-\tau}$ is the adaptive weight matrix for the previous frame($\sum_{\tau=0}^{T} \mathbf{w}^{t-\tau} = 1$), T is the number of previous frames used. In addition, the adaptive weight matrix is obtained by two sets of convolutional layers and the softmax function. To express the steps of feature enhancement more clearly, Algorithm 1 is used here to show the related process.

3.5 Loss function

The overall loss of the PTDS CenterTrack can be defined as follows:

$$L = L_{det} + L_{idp} + L_{ff} \tag{14}$$

where L_{det} is the 2D detection loss in CenterNet [57], L_{idp} is the inter-frame displacement prediction loss, and L_{ff} is the feature fusion loss based on the hybrid attention mechanism.

In terms of displacement prediction, the re-ID feature extraction network is the only learnable part, so its training objective is to learn an effective re-ID embedding. The loss of this part is calculated by the logistic regression in the form of the focal loss [25] as:

$$L_{idp} = \frac{-1}{\sum_{ijkl} Y_{ijkl}} \sum_{ijkl} \begin{cases} \alpha log(C_{ijkl}) & , Y_{ijkl} = 1 \\ 0 & , otherwise \end{cases} \tag{15}$$

where $\alpha = (1 - C_{ijkl})^{\beta}$. β is the focal loss hyperparameter. If $Y_{ijkl} = 1$, an object at location (i, j) at current time t appears at location (k, l) at previous time $t - \tau$; otherwise $Y_{ijkl} = 0$. It is worth mentioning that C_{ijkl} is calculated by softmax from $\mathbf{C}_{i,j,k,l}$, which represents the similarity between the current frame object and itself in the previous frame, and the difference from background and objects at other locations. In terms of feature fusion, the learnable part is mainly reflected in the feature fusion network $\gamma(\cdot)$ based on the hybrid attention mechanism. L_{ff} is superimposed on the original detection loss.

4 Experiments and results

4.1 Benchmarks and evaluation metrics

We organize our experiments on the MOT20 dataset [14] and use the private detection results branch instead of the public detection results provided by the MOT20 dataset. It is worth mentioning that MOT17 [29] is one of the most widely used datasets in multi-object tracking. Compared

Table 1 Evaluation on MOT20 test sets

Method	Det	MOTA↑	IDF1↑	HOTA↑	MT↑	ML↓	FP↓	FN↓	IDs↓
OVBT [2]	Pub	40.0	37.8	30.5	11.4	30.1	23,368	282,949	4210
DeepSORT [47]	Pub	42.7	45.1	36.1	16.7	26.2	27,521	264,694	4470
CenterTrack [59]	Pri	45.5	51.1	40.3	30.0	21.3	**5450**	535,153	15,784
MLT [53]	Pri	48.9	54.6	43.2	30.9	22.1	45,660	216,803	2,187
TraDeS [48]	Pri	52.4	55.0	*44.9*	33.4	20.1	75,427	*162,675*	8320
Tracktor++ [46]	Pub	52.6	52.7	42.1	29.4	26.7	6930	236,680	**1648**
UnsupTrack [19]	Pub	53.6	50.6	41.7	30.3	25.0	*6439*	231,298	2178
TBC [37]	Pub	54.5	50.1	–	33.4	19.7	37,937	195,242	2449
GNNMatch [31]	Pub	54.5	49.0	40.2	32.8	25.5	9522	223,611	2038
SP-CON [43]	Pub	54.6	*53.4*	42.5	32.8	25.5	9486	223,607	*1674*
TransCenter [50]	Pri	**58.5**	49.6	43.5	**48.6**	**14.9**	64,217	**140,019**	4695
PTDS CenterTrack	Pri	*55.6*	**55.1**	**45.1**	*36.9*	*17.4*	50,589	170,933	8242

Bold marks indicate optimal results, italic indicate suboptimal results

with MOT20, there are two main differences: first, compared with the MOT17 scene, the pedestrian density in MOT20 is greatly increased, and the occurrence of occlusion and overlap between objects is higher; second, the scenes involved in all video sequences in the test set of MOT17 are the same as the training set, while the two video sequences MOT20-06 and MOT20-08 in the test set of MOT20 do not have the same or similar scenes in the training set, that is, the scenes of MOT17 in the training set and the test set belong to the same source, while the MOT20 is heterogeneous, so the detection and tracking of the MOT20 test set is more difficult. In addition, this paper uses the annotated samples of the whole body and visual part in the CrowdHuman dataset [39] as the pre-training dataset for the detection model, re-identification feature extraction layer, and feature fusion layer, and the label content only contains the bounding box annotation of the objects.

We evaluate our method on the test set of MOT20. We use mAP [9], MODA, and MODP [20] to evaluate the detection results of the model, which is mainly reflected in the ablation experiments. At the same time, in the ablation experiment, we also give the FP and FN values of different model combinations, these two indicators can more intuitively understand the detection effect of the model. In terms of tracking evaluation, we selected three most popular multi-objective algorithm evaluation indicators: CLEAR MOT [4], ID Scores [24], HOTA [27]. Among them, CLEAR MOT and ID Scores are currently the most widely used evaluation indicators in academia, and HOTA is a new multi-object tracking evaluation indicator proposed based on certain defects of the above two indicators. Measures the accuracy of multi-object tracking.

4.2 Implementation details

We use the same variant of DLA-34 as in CenterNet as the backbone of the overall network. The model of this network is pretrained on the COCO dataset [9] to initialize our model. We trained our network using the Adam optimizer for 70 epochs, starting with a learning rate of 3.25e−5. The learning rate decays to 3.25e−6 at the 60th epoch. We set the batch size to 8. We used some standard data augmentation strategies, including flipping, scaling, and color transforming. The input image size is reshaped to 960 × 544, and the feature map resolution at the regression branch location is 240 × 136. We spent about 12 h in the training phase, on two RTX3090s.

4.3 Results

We finally validate the performance of our proposed network under the MOT20 benchmark. We compared similar algorithms using public and private detection under the MOT20 test set, and the results are shown in Table 1.

PTDS CenterTrack achieves comparable results with similar methods, especially in terms of IDF1 and HOTA. The excellent IDF1 demonstrates PTDS CenterTrack can effectively reduce the occurrence of identity transition and capture more complete trajectories. The good HOTA represents PTDS CenterTrack can achieve a better balance in detection and tracking tasks. It proves the feature enhancement approach we adopt can act on both the detection and tracking parts. Furthermore, PTDS CenterTrack uses simpler feature extraction networks to achieve comparable tracking results compared to tracking frameworks using Transformer (e.g. TransCenter [50]). Our method is slightly weaker under detection metrics such as FN and FP, which we believe is reasonable. This is because the detector we use is very simple

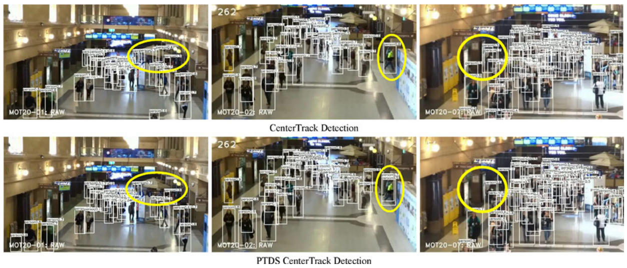


Fig. 9 Comparison of detection before and after improvement in subway scenes

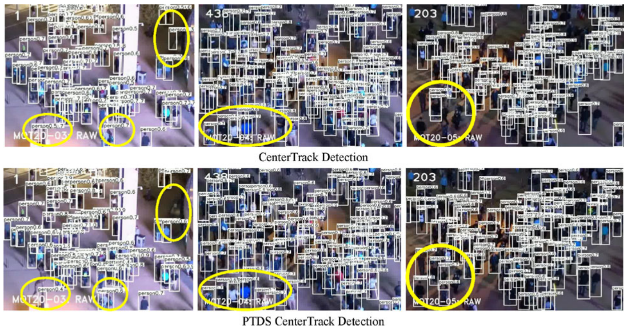


Fig. 10 Comparison of detection before and after improvement in cross scenes

and does not incorporate a relatively more complex object detection network like the Transformer-based method, or directly apply the SOTA detector. Therefore, we believe that PTDS CenterTrack can also enhance detection and tracking performance by utilizing tracking clues when using more naive detectors.

4.4 Discussion

In this subsection, we rigorously investigate three key improvements in PTDS CenterTrack through carefully designed multiple baseline methods. These include improved detection performance, displacement prediction based on re-identified embedded features, inter-frame feature fusion based on a hybrid attention mechanism, and experiments on data association strategies. Furthermore, we discuss the efficiency of the algorithm and give a quantitative analysis in the summary section of the discussion.

4.4.1 Detection level

We first compared from the detection level, before and after the improvement of the Gaussian effective radius generation method, the prediction results of the network's detection module in the MOT20 dataset for 8 video sequences, shown in Figs. 9, 10, 11, 12.

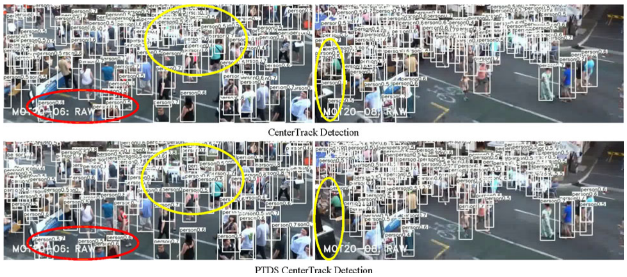
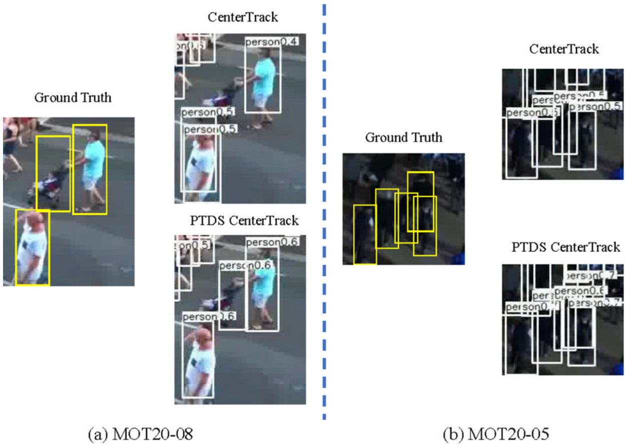


Fig. 11 Comparison of detection before and after improvement in street scenes



(a) MOT20-08

(b) MOT20-05

Fig. 12 Comparison of detection before and after improvement in specific locations captured

We group them according to the different scenes involved in video sequences, where the yellow circles indicate the parts where the improved model detection is better than the baseline, and the red circles indicate the parts where the baseline is better than the improved.

Furthermore, we also extract the heatmap of prediction and center point offset from the regression branches of the detection model and the combined center point detection results from both. Here we select a frame in the MOT20-01 video sequence as an example for illustration, in Fig. 13. It can be clearly seen that the application of the improved generation method can obtain a clearer center position in the regression layer, and the detection results can effectively reduce the false detection of objects.

We propose a feature fusion module based on the hybrid attention mechanism, which uses the detection results of the previous frame to enhance the detection of the current frame, so we also verify the improvement of this module on the detection results. Finally, we compare the results of related improvements with the baseline model we used, in Table2.

4.4.2 Displacement prediction based on Re-ID features

Different from CenterTrack, our designed tracking model applies Re-ID embedding for intra-class discrimination and applies this feature to the part of inter-frame displacement

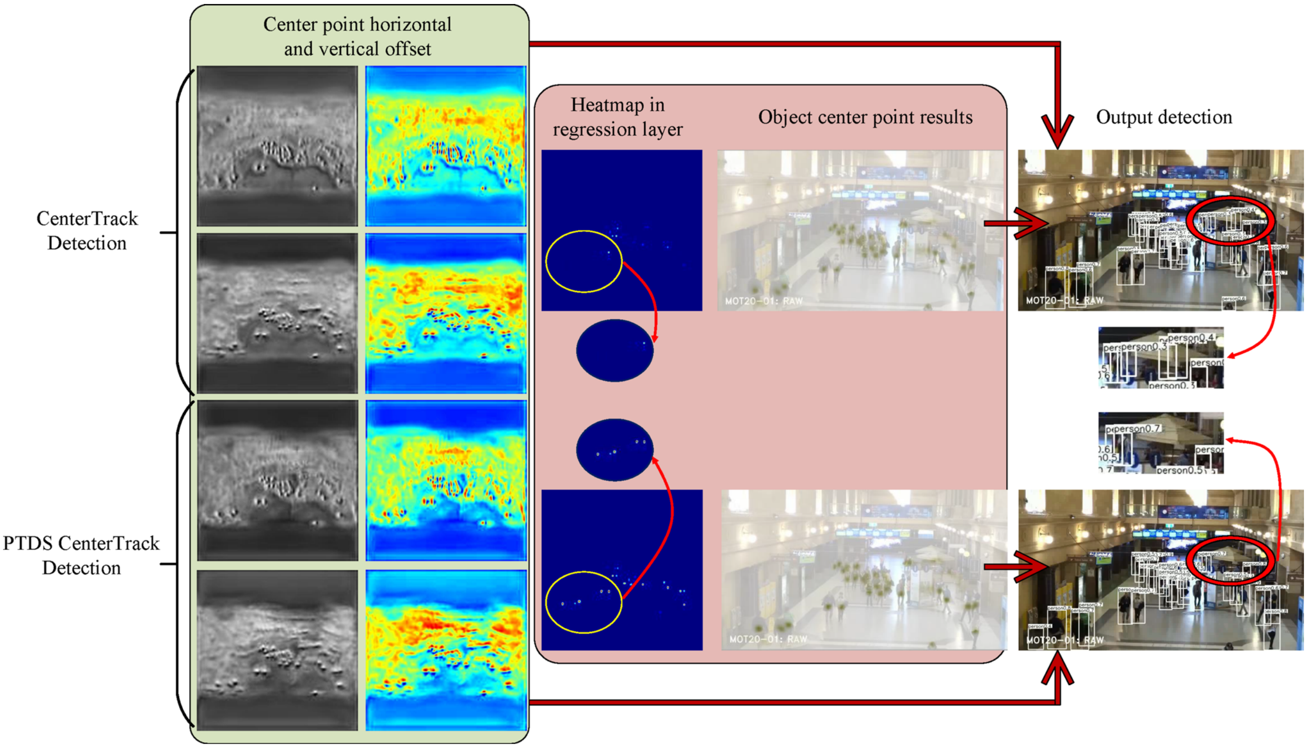

Center point horizontal and vertical offset
Heatmap in regression layer
Object center point results
Output detection
CenterTrack Detection
PTDS CenterTrack Detection

Fig. 13 Comparison of results in regression layers before and after improvement. We extracted the intermediate and final results of CenterTrack and PTDS CenterTrack within the detection part of the network respectively, showing that the improved heat map acquisition method can improve the performance of the detection part. The yellow circled part indicates that a clearer object center point can be obtained in the heat map layer, which facilitates the understanding of the object center features during the training process. The part circled in red is the location in the final detection result where false detections are significantly reduced

Table 2 Object detection performance comparison summary

Method	$AP_{0.5}$	MODA	MODP	Recall	Precision	F1
Base	64.6	64.6	**83.3**	65.7	**98.4**	78.8
Base+GR	71.7	73.8	79.2	80.1	90.3	84.9
Base+GR+ReID+FF	77.6	76.7	81.2	86.5	89.8	88.2
Base+GR+ReID+FFA	**78.2**	**77.3**	81.3	**86.9**	90.5	**88.7**

*’GR’ is the Gaussian effective radius method.
*’FF’ is the feature fusion method.
*’A’ is the hybrid attention mechanism method
Bold marks indicate optimal results

prediction. Therefore, we also conduct a comparative evaluation against tracking models in this improvement. We obtain the tracking results of the basic model under the original network structure of CenterTrack and name the model ’base’. After that, the object inter-frame displacement prediction of the regression layer in CenterTrack is deleted, and the re-identification unit is used to re-identify and extract the basic embedding. The displacement is obtained by the method described in Sec. 3.3. Here we use two different re-identification network structures in Fig. 5, namely the re-identification feature extraction network structure used in the conventional MOT network, and the improved version for it in this paper, and the tracking model formed by these two networks is named ’ReID’ and ’ReID+’. The tracking results of the above three models under the semi-validation set of MOT20 are shown in Table 3.

Table 3 Tracking results of model ’base’, ’base+ReID’, and ’base+(ReID+)’ under MOT20 validation set

Method	MOTA↑	IDF1↑	IDs↓
Base	56.3	36.7	15,326
Base+ReID	**67.9**	**69.6**	**2,472**
Base+(ReID+)	**68.5**	**69.6**	**2,390**

Bold marks indicate optimal results

Table 4 Tracking results of model 'ReID+cos', 'ReID+Euclidean', and 'ReID+dot' under MOT20 validation set

Method	MOTA↑	IDF1↑	IDs↓
Base+ReID+cos	61.5	39.8	50,696
Base+ReID+Euclidean	63.9	39.6	30,701
Base+ReID+dot	**67.9**	**69.6**	**2,472**

Bold values indicate optimal results

From the analysis of the results in the tables, it can be seen that compared with the inter-frame displacement of the direct regression, the use of re-identification embedding can obtain higher MOTA and IDF1, which are increased by 20.6% and 21.7%, respectively. The number of identity switches(IDs) dropped by 83.9% and 83.9%, respectively. Compared with using shallow CNN and ReLU activation function (ReID), using deeper CNN and SiLU activation function (ReID+) can obtain higher MOTA, which is improved by 0.6 percentage points. Among them, the number of identity switches decreased by 3.31%. Compared with the ReLU activation function, the SiLU activation function used in the improved version of the ReID module has certain smooth and non-monotonic characteristics. The smooth transition of its derivative near 0 is more suitable for deep CNN than the ReLU function. Therefore, using the improved re-identification network can obtain better re-identification feature extraction.

Further, we also explore the influence of the cost matrix calculation method on the similarity measure between frames. We have selected 3 different distance calculation methods, corresponding to Eqs. (9), 10, and 11, and named them 'ReID+dot', 'ReID+cos', 'ReID+euclidean'. The tracking results of 'ReID+dot' is the same as Table 3. The tracking results of the latter two under the MOT20 validation set are shown in Table 4.

Through the above results, the tracking results using cosine and Euclidean distance for similarity measure are inferior to the form of matrix multiplication, in which MOTA drops by 10.6% and 7.1%, and IDF1 drops by 43.2% and 43.5%, respectively. We believe that the reason for this phenomenon is the process of obtaining the cost matrix. In the experimental model of this paper, both cosine distance and Euclidean distance will introduce cross-channel features in the calculation process, which will weaken the differences between frames by the norm of the respective feature matrices. In the cosine form of the cost matrix calculation process, the feature matrix norm of the two frames is located in the denominator position, which normalizes the original difference, so that the similarity of the pixel points is lowered. In the calculation process of Euclidean distance, the similarity of pixel points is neutralized by the square term in the formula, and obtaining the similarity representation is also not as effective as matrix multiplication.

Table 5 Tracking results of model '(ReID+)+FF' and '(ReID+)+FFA' under MOT20 validation set

Method	MOTA↑	IDF1↑	IDs↓
Base+(ReID+)+FF	68.8	70.1	2425
Base+(ReID+)+FFA	**68.9**	**70.7**	**2304**

Bold marks indicate optimal results

Table 6 Tracking results of different data association methods under MOT20 validation set

Method	MOTA↑	IDF1↑	IDs↓
(ReID+)+FFA+(H+L)	68.6	70.1	2,481
(ReID+)+FFA+FB	**68.9**	**70.7**	**2,304**

Bold marks indicate optimal results

4.4.3 Feature fusion

After completing the prediction of the displacement between frames, we use this difference information to guide the object detection of the current frame and use two feature fusion networks to perform ablation experiments. One of them uses a time-series feature fusion structure similar to PoseWarper, and the other uses a feature fusion network based on a hybrid attention mechanism. We name them FF and FFA respectively, and their tracking results on the MOT20 semi-validation set are shown in Table 5.

From the above results, it can be seen that the feature fusion layer with the addition of an attention mechanism can effectively reduce the ID switches by 5.0%, and can effectively improve IDF1 by 0.6 percentage points. We believe that this is because the feature fusion unit takes the feature frame difference result as the main input of the network, and the multi-channel frame difference down-mix attention module can learn the global correlation lines of each channel, which is beneficial to utilize the features more effectively.

4.4.4 Data association

This section conducts comparative experiments on data association methods, mainly comparing the "secondary association algorithm" mentioned in ByteTrack [55] with the feature bank method constructed based on re-identification embedding in this paper. Here, the two are named as the 'H+L' and the 'FB' association method, respectively. The tracking results obtained by using two different data association methods are shown in Table 6.

ByteTrack considers the importance of low-confidence results to trajectories, so the 'secondary association' method

Table 7 Evaluation comparison and efficiency of different improved models under MOT20 validation set

Model	MOTA↑	Para	Size(MB)	FPS
Base	56.3	**19.81M**	**239.5**	**17.0**
+ReID+FF	68.3	21.15M	253.9	15.3
+(ReID+)+FF	68.6	21.45M	257.6	14.6
+(ReID+)+FFA	**68.9**	21.74M	261.2	13.2

* The last three rows of methods in the table are suitable for Gaussian effective radius improvement(GR)
Bold marks indicate optimal results

Table 8 Evaluation comparison of ReID modules with different depths under MOT20 validation set

Method	MOTA↑	IDF1↑	IDs↓
ReID(4)	68.0	70.1	2,425
ReID+(5)	**68.6**	**70.7**	**2,304**
ReID+(7)	68.5	70.3	2,321
ReID+(10)	68.3	70.0	2,431

* The number in parentheses indicates the number of convolutional layers. The above methods all apply the 'GR' and 'FF' improvements
Bold marks indicate optimal results

Table 9 Evaluation comparison of FF modules with different access location under MOT20 validation set

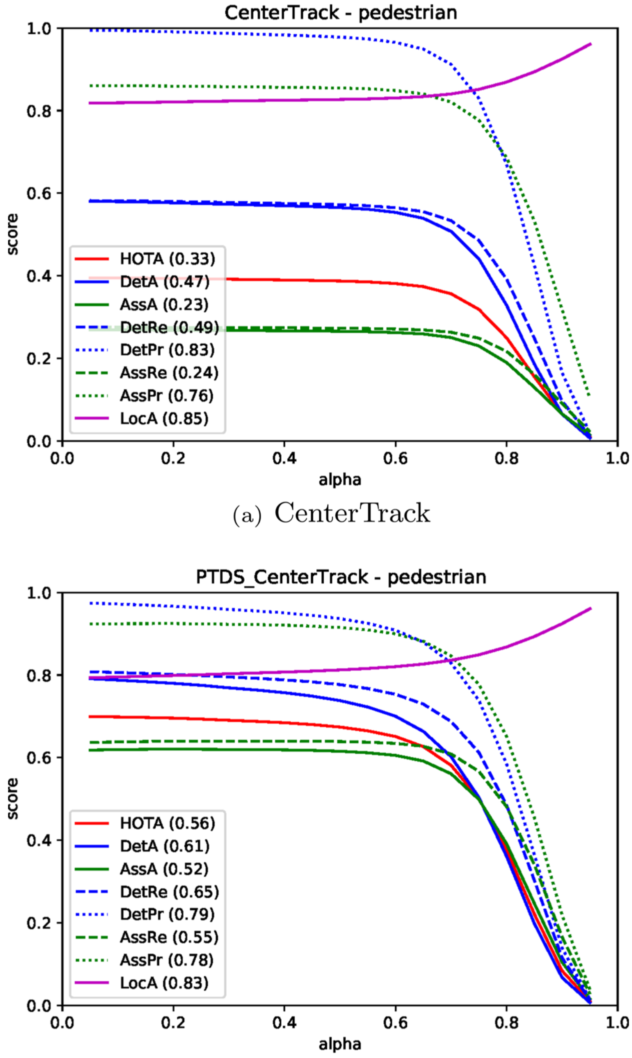
Method	MODA↑	MOTA↑	IDF1↑
FF	76.7	68.6	70.1
FFA	**77.3**	**68.9**	**70.7**
FFA-i	76.0	68.2	69.8
FFA-o	75.8	68.1	69.7

* The above methods all apply the 'GR' and 'ReID+' improvements
Bold marks indicate optimal results

is adopted to use the low-confidence results as the tracking basis. However, it can be seen from the data in the table that under the tracking framework of this paper, even if the secondary association strategy is not used, better tracking results can be obtained. Compared with the 'secondary association' method, the 'feature bank' we adopt focuses on objects in a longer time dimension, and it achieves an improvement of 0.3 and 0.6 percentage points in both MOTA and IDF1 respectively.

4.4.5 Summary of results

In this paper, the above experimental results and the tracking results of the optimal network model combination are integrated, as shown in Table 10, the bold font indicates the significant improvement results. Among them, ✓ indicates that the model adopts the corresponding improvement method or applies this method, and the last line indicates the tracking result under the optimal situation of the network line. Compared with the original CenterTrack, the improved network improves by 12.6 percentage points in MOTA, 34.0 percentage points in IDF1, and reduces the number of identity transitions by 85.0%.



(a) CenterTrack

(b) PTDS CenterTrack

Fig. 14 The original CenterTrack and the PTDS CenterTrack HOTA metrics curve

In addition, we also compared the parameter quantities and model sizes of the improved models. As shown in Table 7, better tracking results are obtained at the cost of increasing the parameter quantity by 9.06%. In addition, we also verified the changes in algorithm efficiency. The CenterTrack (base) model runs the most efficiently, and as our additive improvements increase, the running efficiency decreases. However, compared to more complex detection and tracking networks such as TransCenter (8.7FPS), the overall processing speed of PTDS CenterTrack remains high.

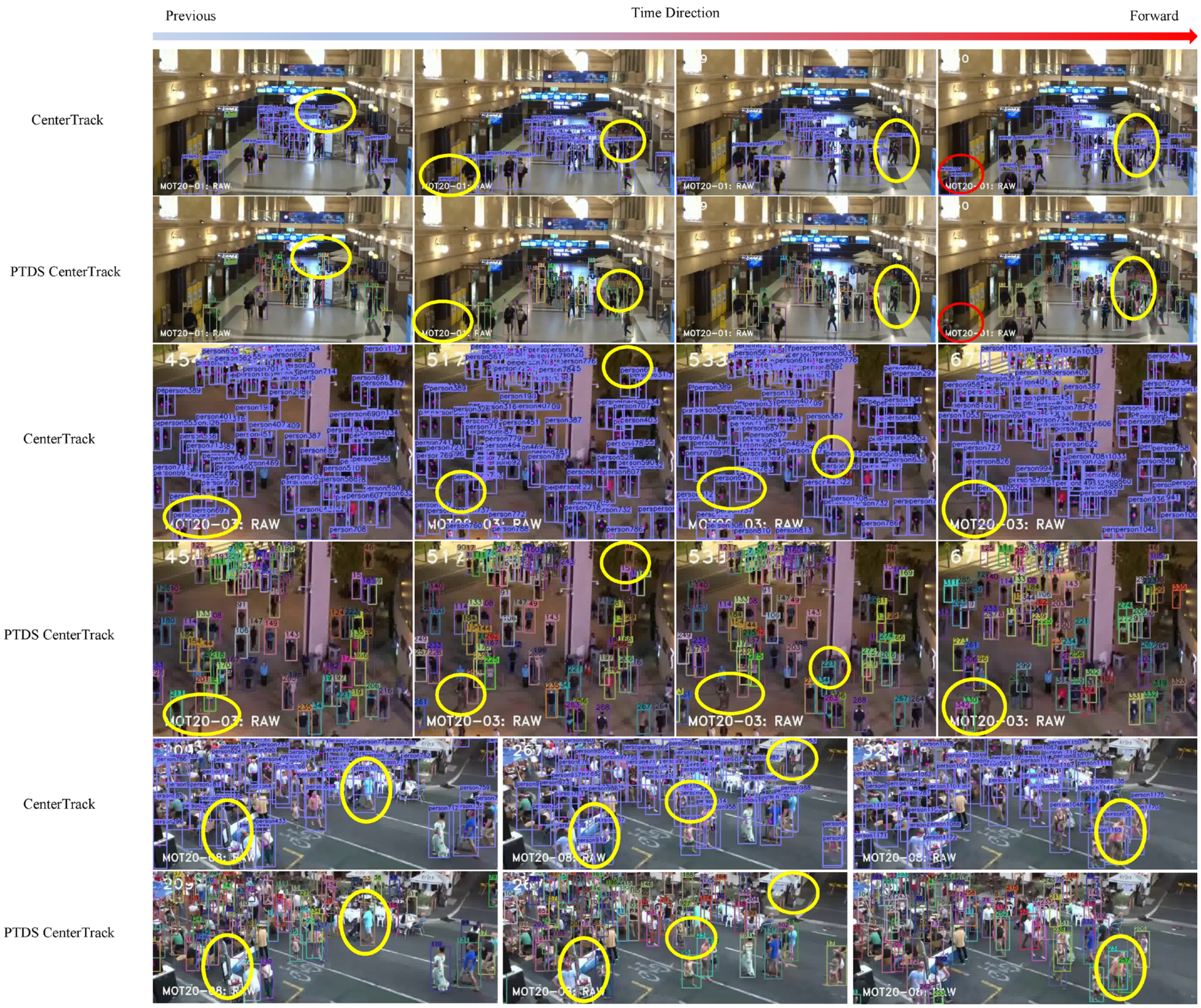


Fig. 15 Visual tracking results comparison between CenterTrack and PTDS CenterTrack. We select some frames in 6 video sequences in chronological order for algorithm comparison. We use yellow circles to denote some areas of improved tracking, and red circles to denote areas of decreased tracking

Furthermore, we also explored the marginal effects of improvements to the 'ReID' and 'FF' parts, as shown in Table 8. For the 'ReID' part, we explore further deepening its network layer (ReID is originally 4 layers, ReID+ is 5 layers), and extend the convolutional layer to a deeper level of 7 and 10 layers (the network structure is expanded in the form of ReID+, and applied 'GR' and 'FFA') and examined their impact on tracking accuracy. We have observed that after increasing the number of 'ReID+' network layers, the tracking effect does not change significantly, and may even decline. At the same time, such changes will make the training process longer and make the model more likely to overfit. Therefore, we did not continue to increase the number of network layers in the 'ReID+' part.

For the 'FF' part, we considered the impact of adding hybrid attention mechanism modules from different locations as shown in Table 9. 'FFA' is a symmetrical hybrid attention mechanism module added to the input and output parts of the feature hybrid network. We also consider the case of adding the attention mechanism only to the input and adding the attention mechanism only to the output, and name them respectively' FFA-i''FFA-o' (apply 'GR' and 'ReID+'). We evaluated the impact of different change methods from two levels: detection and tracking. We have observed that when using a unilateral attention mechanism access method, the performance of both detection and tracking will decrease, especially the detection part.

At the same time, we also use the HOTA evaluation standard to compare CenterTrack and the improved model, as

Table 10 Summary of ablation experiment results

ReID	ReID+	dot	cos	Euclidean	Feature Fusion	Attention	MOTA↑	IDF1↑	IDs↓
							56.3	36.7	15,326
✓		✓					67.9	69.6	2,472
	✓	✓					68.5	69.6	2,390
✓			✓				61.5	39.8	50,696
✓				✓			63.9	39.6	30,701
	✓	✓			✓		68.6	70.1	2,425
	✓	✓			✓	✓	**68.9**	**70.7**	**2,304**

* ’dot’, ’cos’, and ’Euclidean’ are the three similarity measurement methods
Bold marks indicate optimal results

shown in Fig. 14. We improved by 23, 14, and 29 percentage points on HOTA, DetA, and AssA respectively. We also show the visual tracking results of CenterTrack and the improved model in Fig. 15. It can be seen from the above results that the PTDS CenterTrack can significantly reduce the number of identity switches and obtain more accurate detection and tracking results (Table 10).

5 Conclusions

In this study, we propose a multi-object tracking method named PTDS CenterTrack, which utilizes re-identification features to construct a cost matrix for predicting the inter-frame displacement of objects. Simultaneously, we develop an inter-frame feature fusion network based on a hybrid attention mechanism and feed the object displacement information back to the detection module, thereby leveraging tracking cues to enhance detection performance. Additionally, we introduce a novel heat map acquisition method to improve the learning of object center point features during the training process. We compared with the baseline and similar algorithms from various perspectives, and achieved 68.9%MOTA, 70.7%IDF1, and 55.6%MOTA, 45.1%HOTA on the validation and test set of the extremely challenging benchmark MOT20, respectively. These results confirm that the relationship between detection tasks and tracking tasks extends beyond conventional sequential structures, underscoring the significance of tracking cues. Our observations from ablation experiments demonstrate that the comprehensive utilization of both motion and appearance information between frames significantly enhances the robustness of multi-object tracking. Specifically, the addition of a re-identification branch that focuses on an entire frame to existing lightweight trackers directly improves multi-object tracking performance. Going forward, we aim to delve deeper into the correlation between detection and tracking tasks within the MOT community and endeavor to develop a more efficient joint tracking model.

Acknowledgements This study was supported by the National Natural Science Foundation of China (Grant No.62271166) and the Interdisciplinary Research Foundation of HIT (No.IR2021104).

Funding This study was funded by the National Natural Science Foundation of China (Grant No.62271166) and the Interdisciplinary Research Foundation of HIT (No.IR2021104).

Declarations

Conflict of interest None

References

1. Artacho, B., Savakis, A.: Unipose+: a unified framework for 2d and 3d human pose estimation in images and videos. IEEE Trans. Pattern Anal. Mach. Intell. **44**(12), 9641–9653 (2022). https://doi.org/10.1109/TPAMI.2021.3124736
2. Ban, Y., Ba, S., Alameda-Pineda, X., et al.: Tracking multiple persons based on a variational bayesian model. In: European Conference on Computer Vision, Springer, pp 52–67, https://doi.org/10.1007/978-3-319-48881-3_5 (2016)
3. Bergmann, P., Meinhardt, T., Leal-Taixe, L.: Tracking without bells and whistles. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 941–951, https://doi.org/10.1109/ICCV.2019.00103 (2019)
4. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance: the clear mot metrics. EURASIP J. Image Video Process. **2008**, 1–10 (2008). https://doi.org/10.1155/2008/246309
5. Bertasius, G., Feichtenhofer, C., Tran, D., et al.: Learning temporal pose estimation from sparsely-labeled videos. In: Wallach H, Larochelle H, Beygelzimer A, et al (eds) Advances in neural information processing systems, vol 32. Curran Associates, Inc., https://proceedings.neurips.cc/paper_files/paper/2019/file/7137debd45ae4d0ab9aa953017286b20-Paper.pdf (2019)
6. Bewley, A., Ge, Z., Ott, L., et al.: Simple online and realtime tracking. In: 2016 IEEE International Conference on Image Processing (ICIP), IEEE, pp 3464–3468, https://doi.org/10.1109/ICIP.2016.7533003 (2016)
7. 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, pp 6247–6257, https://doi.org/10.1109/CVPR42600.2020.00628 (2020)
8. Chen, D., Zhang, S., Ouyang, W., et al.: Person search via a mask-guided two-stream CNN Model. pp 734–750, https://

openaccess.thecvf.com/content_ECCV_2018/html/Di_Chen_Person_Search_via_ECCV_2018_paper.html (2018)

9. Chen, X., Fang, H., Lin, T.Y., et al.: Microsoft coco captions: data collection and evaluation server. arXiv:1504.00325 https://doi.org/10.48550/arXiv.1504.00325 (2015)
10. Ciaparrone, G., Luque Sánchez, F., Tabik, S., et al.: Deep learning in video multi-object tracking: a survey. Neurocomputing **381**, 61–88 (2020). https://doi.org/10.1016/j.neucom.2019.11.023
11. Dai, J., Li, Y., He, K., et al.: R-fcn: object detection via region-based fully convolutional networks. Adv. Neural Inform. Process. Syst. (2016). https://doi.org/10.5555/3157096.3157139
12. Dai, J., Qi, H., Xiong, Y., et al.: Deformable convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision, pp 764–773, https://doi.org/10.1109/ICCV.2017.89 (2017)
13. Dave, A., Khurana, T., Tokmakov, P., et al.: Tao: a large-scale benchmark for tracking any object. In: European Conference on Computer Vision, Springer, pp 436–454, https://doi.org/10.1007/978-3-030-58558-7_26 (2020)
14. Dendorfer, P., Rezatofighi, H., Milan, A., et al.: Mot20: A benchmark for multi object tracking in crowded scenes. arXiv preprint https://doi.org/10.48550/arXiv.2003.09003 (2020)
15. Ge, Z., Liu, S., Wang, F., et al.: Yolox: Exceeding yolo series in 2021. arXiv preprint arXiv:2107.08430 https://doi.org/10.48550/arXiv.2107.08430 (2021)
16. Geiger, A., Lenz, P., Stiller, C., et al.: Vision meets robotics: the kitti dataset. Int. J. Robot. Res. **32**(11), 1231–1237 (2013). https://doi.org/10.1177/0278364913491297
17. Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE International Conference on Computer Vision, pp 1440–1448, https://doi.org/10.1109/ICCV.2015.169 (2015)
18. Jaouedi, N., Boujnah, N., Bouhlel, M.S.: A new hybrid deep learning model for human action recognition. J. King Saud Univ.-Comput. Inform. Sci. **32**(4), 447–453 (2020). https://doi.org/10.1016/j.jksuci.2019.09.004
19. Karthik, S., Prabhu, A., Gandhi, V.: Simple unsupervised multi-object tracking. arXiv preprint arXiv:2006.02609 https://doi.org/10.48550/arXiv.2006.02609 (2020)
20. Kasturi, R., Goldgof, D., Soundararajan, P., et al.: Performance Evaluation Protocol for Face, Person and Vehicle Detection and Tracking in Video Analysis and Content Extraction (vace-ii). Computer Science & Engineering University of South Florida, Tampa (2006)
21. Kong, T., Sun, F., Liu, H., et al.: Foveabox: beyound anchor-based object detection. IEEE Trans. Image Process. **29**, 7389–7398 (2020). https://doi.org/10.1109/TIP.2020.3002345
22. Law, H., Deng, J.: Cornernet: detecting objects as paired keypoints. In: Proceedings of the European Conference on Computer Vision (ECCV), pp 734–750, https://doi.org/10.1007/s11263-019-01204-1 (2018)
23. Leal-Taixé, L., Milan, A., Reid, I., et al.: Motchallenge 2015: Towards a benchmark for multi-target tracking. arXiv preprint arXiv:1504.01942 https://doi.org/10.48550/arXiv.1504.01942 (2015)
24. Li, G.B., Yang, L.L., Wang, W.J., et al.: Id-score: a new empirical scoring function based on a comprehensive set of descriptors related to protein-ligand interactions. J. Chem. Inf. Model. **53**(3), 592–600 (2013). https://doi.org/10.1021/ci300493w
25. Lin, T.Y., Goyal, P., Girshick, R., et al.: Focal loss for dense object detection. IEEE Trans. Pattern Anal. Mach. Intell. **42**(2), 318–327 (2020). https://doi.org/10.1109/TPAMI.2018.2858826
26. Liu, W., Anguelov, D., Erhan, D., et al.: Ssd: Single shot multibox detector. In: European Conference on Computer Vision, Springer, pp 21–37, https://doi.org/10.1007/978-3-319-46448-0_2 (2016)
27. Luiten, J., Osep, A., Dendorfer, P., et al.: Hota: a higher order metric for evaluating multi-object tracking. Int. J. Comput. Vision **129**(2), 548–578 (2021). https://doi.org/10.1007/s11263-020-01375-2
28. Maekawa, T., Ohara, K., Zhang, Y., et al.: Deep learning-assisted comparative analysis of animal trajectories with deephl. Nat. Commun. **11**(1), 1–15 (2020). https://doi.org/10.1038/s41467-020-19105-0
29. Milan, A., Leal-Taixé, L., Reid, I., et al.: Mot16: a benchmark for multi-object tracking. arXiv preprint arXiv:1603.00831 https://doi.org/10.48550/arXiv.1603.00831 (2016)
30. Pang, B., Li, Y., Zhang, Y., et al.: Tubetk: Adopting tubes to track multi-object in a one-step training model. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 6308–6318, https://doi.org/10.1109/CVPR42600.2020.00634 (2020)
31. Papakis, I., Sarkar, A., Karpatne, A.: A graph convolutional neural network based approach for traffic monitoring using augmented detections with optical flow. In: 2021 IEEE International Intelligent Transportation Systems Conference (ITSC), IEEE, pp 2980–2986, https://doi.org/10.48550/arXiv.2010.00067 (2021)
32. Pedersen, M., Haurum, J.B., Dendorfer, P., et al.: MOTCOM: the multi-object tracking dataset complexity metric. In: Avidan, S., Brostow, G., Cissé, M., et al (eds) Computer Vision - ECCV 2022. Springer Nature Switzerland, Cham, Lecture Notes in Computer Science, pp 20–37, https://doi.org/10.1007/978-3-031-20074-8_2 (2022)
33. Peng, J., Wang, C., Wan, F., et al.: Chained-tracker: chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking. In: European Conference on Computer Vision, Springer, pp 145–161, https://doi.org/10.1007/978-3-030-58548-8_9 (2020)
34. Redmon, J., Farhadi, A.: Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767 https://doi.org/10.48550/arXiv.1804.02767 (2018)
35. Reid, D.: An algorithm for tracking multiple targets. IEEE Trans. Autom. Control **24**(6), 843–854 (1979). https://doi.org/10.1109/TAC.1979.1102177
36. Ren, S., He, K., Girshick, R., et al.: Faster r-cnn: towards real-time object detection with region proposal networks. Adv. Neural Inform. Process. Syst. (2015). https://doi.org/10.1109/TPAMI.2016.2577031
37. Ren, W., Wang, X., Tian, J., et al.: Tracking-by-counting: using network flows on crowd density maps for tracking multiple targets. IEEE Trans. Image Process. **30**, 1439–1452 (2020). https://doi.org/10.1109/TIP.2020.3044219
38. Schulter, S., Vernaza, P., Choi, W., et al.: Deep network flow for multi-object tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp 6951–6960, https://doi.org/10.1109/CVPR.2017.292 (2017)
39. Shao, S., Zhao, Z., Li, B., et al.: Crowdhuman: A benchmark for detecting human in a crowd. arXiv preprint arXiv:1805.00123 https://doi.org/10.48550/arXiv.1805.00123 (2018)
40. Sun, P., Cao, J., Jiang, Y., et al.: Transtrack: Multiple object tracking with transformer. arXiv preprint arXiv:2012.15460 https://doi.org/10.48550/arXiv.2012.15460 (2020)
41. Tian, Z., Shen, C., Chen, H., et al.: Fcos: Fully convolutional one-stage object detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision, pp 9627–9636, https://doi.org/10.1109/CVPR.2019.00094 (2019)
42. Vaswani, A., Shazeer, N., Parmar, N., et al.: Attention is all you need. Advances in neural information processing systems 30. https://doi.org/10.48550/arXiv.1706.03762 (2017)
43. Wang, G., Wang, Y., Gu, R., et al.: Split and connect: a universal tracklet booster for multi-object tracking. IEEE Trans. Multimedia (2022). https://doi.org/10.1109/TMM.2022.3140919

44. Wang, Q., Zheng, Y., Pan, P., et al.: Multiple object tracking with correlation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp 3876–3886, https://doi.org/10.1109/CVPR46437.2021.00387 (2021a)
45. Wang, Y., Kitani, K., Weng, X.: Joint object detection and multi-object tracking with graph neural networks. In: 2021 IEEE International Conference on Robotics and Automation (ICRA). IEEE Press, p 13708-13715, https://doi.org/10.1109/ICRA48506.2021.9561110, (2021b)
46. Wang, Z., Zheng, L., Liu, Y., et al.: Towards real-time multi-object tracking. In: European Conference on Computer Vision, Springer, pp 107–122, https://doi.org/10.1007/978-3-030-58621-8_7 (2020)
47. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric. In: 2017 IEEE International Conference on Image Processing (ICIP), IEEE, pp 3645–3649, https://doi.org/10.1109/icip.2017.8296962 (2017)
48. Wu, J., Cao, J., Song, L., et al.: Track to detect and segment: An online multi-object tracker. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pp 12352–12361, https://doi.org/10.48550/arXiv.2103.08808 (2021)
49. Xu, Y., Ban, Y., Delorme, G., et al.: Transcenter: Transformers with dense representations for multiple-object tracking. IEEE Transactions on Pattern Analysis and Machine Intelligence pp 1–16. https://doi.org/10.1109/TPAMI.2022.3225078 (2022)
50. Xu, Y., Ban, Y., Delorme, G., et al.: Transcenter: transformers with dense representations for multiple-object tracking. IEEE Trans. Pattern Anal. Mach. Intell. **45**(6), 7820–7835 (2023). https://doi.org/10.1109/TPAMI.2022.3225078
51. Zhang, H., Chang, H., Ma, B., et al.: Cascade retinanet: Maintaining consistency for single-stage object detection. British Machine Vision Conference https://doi.org/10.48550/arXiv.1907.06881, https://api.semanticscholar.org/CorpusID:196831468 (2019)
52. Zhang, L., Li, Y., Nevatia, R.: Global data association for multi-object tracking using network flows. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, pp 1–8, https://doi.org/10.1109/CVPR.2008.4587584 (2008)
53. Zhang, Y., Sheng, H., Wu, Y., et al.: Multiplex labeling graph for near-online tracking in crowded scenes. IEEE Internet Things J. **7**(9), 7892–7902 (2020). https://doi.org/10.1109/JIOT.2020.2996609
54. Zhang, Y., Wang, C., Wang, X., et al.: Fairmot: on the fairness of detection and re-identification in multiple object tracking. Int. J. Comput. Vision **129**(11), 3069–3087 (2021). https://doi.org/10.1007/S11263-021-01513-4
55. Zhang, Y., Sun, P., Jiang, Y., et al.: Bytetrack: Multi-object tracking by associating every detection box. In: Avidan, S., Brostow, G., Cissé, M., et al. (eds.) Computer Vision - ECCV 2022, pp. 1–21. Springer Nature Switzerland, Cham (2022)
56. Zhang, Z., Cheng, D., Zhu, X., et al.: Integrated object detection and tracking with tracklet-conditioned detection. arXiv preprint arXiv:1811.11167 https://doi.org/10.48550/arXiv.1811.11167 (2018)
57. Zhou, X., Wang, D., Krähenbühl, P.: Objects as points. arXiv preprint arXiv:1904.07850 https://doi.org/10.48550/arxiv.1904.07850 (2019a)
58. Zhou, X., Zhuo, J., Krahenbuhl, P.: Bottom-up object detection by grouping extreme and center points. In: Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition, pp 850–859, https://doi.org/10.1109/CVPR.2019.00094 (2019b)
59. Zhou, X., Koltun, V., Krähenbühl, P.: Tracking objects as points. In: European Conference on Computer Vision, Springer, pp 474–490, https://doi.org/10.1007/978-3-030-58548-8_28 (2020)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.