# Combined Multilevel Detection and Trajectory Prediction with FairMOT for Pedestrian Multi-object Tracking

Bao Liu, Jiaxuan Wang, Zhiming Wang

*Abstract*—**This paper presents a multilevel detection-trajectory prediction FairMOT (MD-TPFairMOT) pedestrian multi-object tracking approach to address the challenges of missed tracking and frequent identity switches in dense scenes. These issues are often exacerbated by factors such as object occlusion, irregular motion patterns, and the high visual similarity among pedestrians. The MD-TPFairMOT approach combines a multilevel detection (MLD) and LSTM-based trajectory prediction network (TP-LSTM). Specifically, the MLD categorizes pedestrian objects into five levels based on their scales for detection, and employs an adaptive pedestrian central sampling region to reduce the missed tracking in complex environments. The TP-LSTM uses the object bounding box and velocity information from previous frames to address prediction failures caused by occlusion. Moreover, the network fuses the appearance and motion features during the data association, thereby mitigating excessive reliance on the appearance feature. Finally, the superiority of MD-TPFairMOT over other algorithms (see, e.g., FairMOT, CTrackerV1, and CenterTrack) is verified on the MOTChallenge dataset. The results indicate that MD-TPFairMOT exhibits superior occlusion resistance and accuracy.**

*Index Terms*—**pedestrian multi-object tracking, multilevel detection, trajectory prediction, data association, FairMOT**

## I. Introduction

IN the field of computer vision, pedestrian multi-object tracking (MOT) has extensive applications in areas including the automatic driving, video surveillance, action recognition, and motion analysis [1]-[2]. The prevailing framework for pedestrian MOT revolves around the detection-and-tracking paradigm in the realm of deep learning. The framework initially detects all pedestrian objects in each video frame, subsequently utilizing discriminatory features, such as appearance and motion, to correlate the same pedestrian across different video frames, thereby constructing the motion trajectories and achieving the goal of pedestrian MOT [3].

Tracking multiple pedestrian objects in dense scenarios will face numerous problems due to the complexity of objects and environments [4]. The frequent occlusion, uncertainty in movement, and high similarity in appearance may exert negative impacts on pedestrian detection and subsequent identity (ID) matching [5]. For this reason, some scholars have proposed detection-based MOT methods. Based on the degree of integration within the algorithmic framework, the existing algorithms can be categorized into two-step, one-step, and end-to-end methods. Among these methods, SORT [6], DeepSort [7], and other two-step methods could choose the optimal model when realizing the sub-tasks of detection, re-identification (Re-ID), and motion prediction. However, these sub-tasks are heavily reliant on the performance of the detector and will increase the redundant computation and introduce the redundancy in computation, rendering the algorithms unable to fulfill real-time requirements. Although MOT can be implemented using a single network, such an approach often results in significant computational time due to the extensive size and complexity of the network model. One-step methods such as DAN [8], MOTR [9], Track R-CNN [10], JDE [11], and FairMOT [12] combine the detection and Re-ID tasks, which reduces the redundant computation and the dependence of the Re-ID task on the object detection. These methods offer faster training and deployment times, and are more straightforward to implement, compared to traditional detection-tracking models, while still achieving the state-of-the-art performance.

To tackle the challenges of ID switching and tracking failure in dense scenes, which are often caused by frequent occlusions, irregular motion patterns, and high appearance similarity among objects, this paper proposes a novel MOT method called MD-TPFairMOT. This method is built upon the FairMOT algorithm from the one-step approach and integrates multilevel detection with trajectory prediction. The main contributions are summarized as follows.

(1) This paper proposes a multilevel detection method, MLD, to solve the problem of missed detection and following of pedestrians caused by frequent occlusion. The method utilizes the different sizes of pedestrians during occlusion to assign pedestrians of different scales to five separate levels for detection. In addition, the method solves the centroid offset problem through the multilevel detection. Finally, MLD designs an adaptive pedestrian central sampling region so that the detection frame contains more accurate pedestrian object features.

Manuscript received August 15, 2024; revised February 5 2025.

This work was supported in part by grant for Beilin District Science and Technology Plan Project (GX2231), the Key Research and Development Program of Shaanxi (2021GY-131), and Yulin Science and Technology Plan Project (CXY-2020-037).

Bao Liu is an associate professor of Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (corresponding author, +86-18149067968, e-mail: xiaobei0077@163.com).

Jiaxuan Wang is a postgraduate student of Electrical and Control Engineering, Xi'an University of Science and Technology, Xi'an 710054, China (e-mail: 1578687173@qq.com).

Zhiming Wang is an algorithm engineer of Geovis Insighter Technology Co., Ltd. Xi'an Branch, Xi'an 710100, China (e-mail: 1529265829@qq.com).

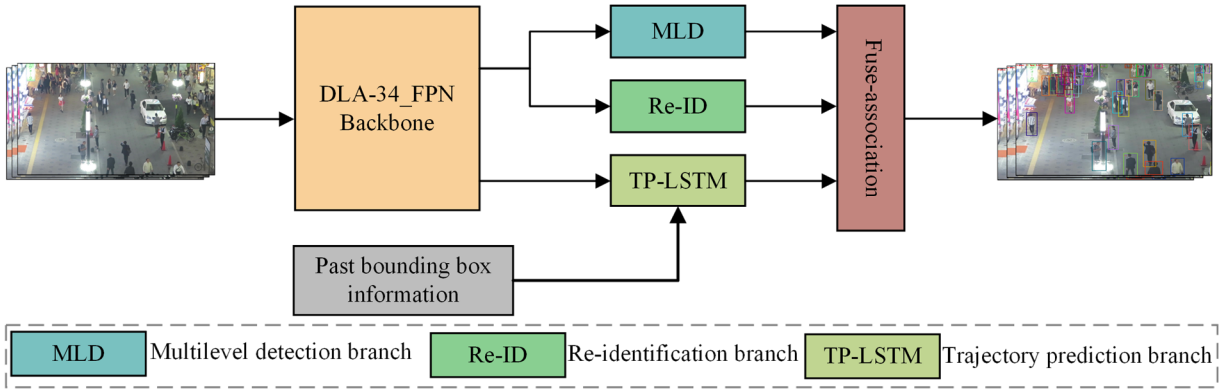



Fig. 1. The model architecture of MD-TPFairMOT

(2) In order to solve the problem of inaccurate prediction due to the uncertainty in pedestrian motion state, this paper proposes LSTM-based trajectory prediction (TP-LSTM). This method utilizes the object bounding box and velocity information from previous frames to predict the current frame's position, and compares the prediction information with the real information to adjust the corresponding weights, thereby enhancing the accuracy of pedestrian motion prediction.

(3) During the matching process, it integrates appearance and motion features, thereby minimizing the reliance on appearance features alone and substantially enhancing the matching accuracy. Addressing the problem of ID matching error caused by the high similarity of pedestrian appearance, a data association method based on fused features (Fuse-association) is proposed. It fuses the appearance and motion features during the matching process, which reduces the dependence on appearance features and improves the matching accuracy.

This paper is an expanded and refined version of our conference paper [13], with much more contents (e.g., the length is more than doubled). It provides more innovations and experiments based on [13], including mainly the following: (a) A multilevel detection method which is more suitable for pedestrian multi-object tracking is proposed. (b) A data association method based on fusion features is proposed. (c) The working principle and steps of the trajectory prediction branch are described in more detail, and (d) ablation experiments and comparison experiments are added, and more results and better discussions are obtained.

This paper is organized as follows. Section II introduces the overall framework of the MD-TPFairMOT and the details of each branch. In Section III, the experimental scheme and results are introduced. The conclusions are presented in Section IV.

## II. Proposed Method

### A. Overall Framework

In crowded pedestrian scenes, individuals frequently obstruct each another, causing the bounding boxes to have closely positioned or even overlapping center points. In this case, CenterNet [14], the detection branch of FairMOT, will only detect and return one object, which will result in the missed detections and missed follow-ups, and when occluded pedestrians reappear, they may be matched incorrectly, causing frequent ID changes. In addition, the original algorithm uses Kalman filtering for trajectory prediction to assist with the Re-ID features when completing the matching of the trajectory with the new frame detection. Pedestrian movements often influence one another and do not adhere to linear motion patterns, but Kalman filtering ignores these interactions between pedestrians and is only applicable to predicting the trajectory state of linear motion.

The MD-TPFairMOT algorithm is proposed to solve the above problems existing in the pedestrian MOT method, and its network architecture is shown in Fig. 1. For accommodating the multilevel detection, the original backbone network DLA-34 has been enhanced to extract more accurate features, and then the proposed MLD is combined with the Re-ID branch to carry out the detection and feature extraction tasks at the same time. In addition, the proposed TP-LSTM method leverages the past bounding box information to predict the pedestrian's trajectories [15]. Finally, the Fuse-association fuses the Re-ID features with the motion features to match the detection results and finally complete the pedestrian MOT task. Each part is explained in detail in the following sections.

### B. Feature Extraction Network

The detection branch of the MD-TPFairMOT adopts the multilevel detection method to assign pedestrian objects to different levels for being detected. Inspired by the algorithmic structure of the FPN [16], this study designs a DLA-34_FPN backbone network for multilevel detection, which is based on the DLA-34 feature extraction network of FairMOT. The schematic structures of DLA-34_FPN and DLA-34 are shown in Fig. 2. In DLA-34_FPN, feature maps $C_2$, $C_3$, $C_4$, and $C_5$ are taken to generate $P_3$, $P_4$, and $P_5$ levels. The $P_4$ and $P_5$ levels are down-sampled to generate $P_6$ and $P_7$, respectively. $P_3$ to $P_7$ are defined as five different levels that are used for the final prediction of the MLD, replacing the single final detection level $P$ from DLA-34. $s$ is the down-sampling ratio of the feature map at the level to the input image. DLA-34_FPN is constructed by downsampling the original output feature maps and subsequently integrating these downsampled maps with the features sourced from the corresponding convolutional layers within the DLA-34 backbone network. This process enables the DLA-34_FPN to leverage multi-scale feature information, thereby enhancing its detection capabilities. By effectively combining the low-level details from early convolutional layers with the high-level semantics from later layers, DLA-34_FPN achieves robust performance in multi-level detection tasks.

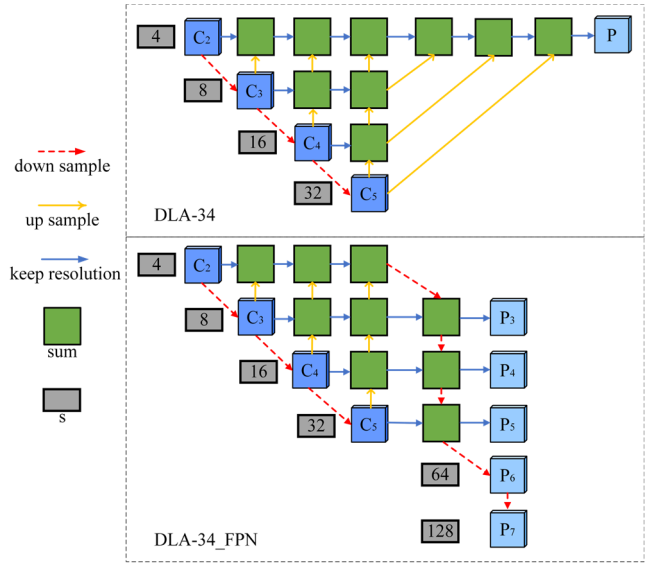



Fig. 2. Schematic structures of DLA-34 and DLA-34_FPN

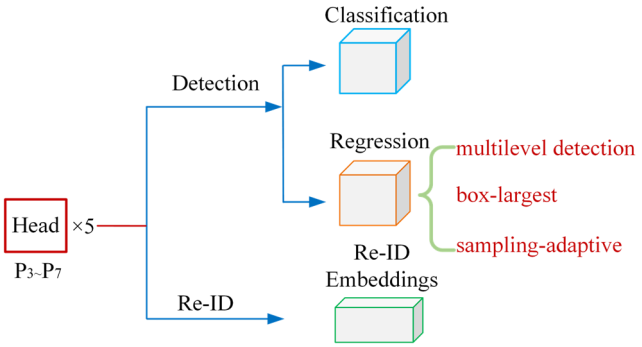


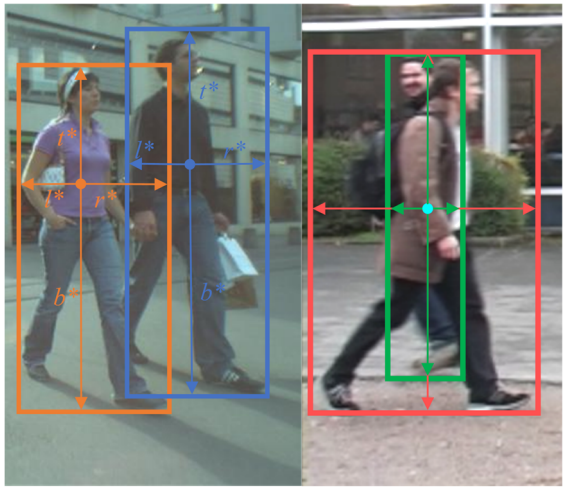

Fig. 3. Shared output head for detection and Re-ID

### C. MLD for the MD-TPFairMOT

The MD-TPFairMOT adopts the identical one-step architecture as FairMOT. The input image is initially processed through DLA-34_FPN to obtain the multi-level feature map, which is subsequently fed into the detection and Re-ID branches for detection and feature extraction purposes. Both detection and Re-ID share a common output head, and the structure is shown in Fig. 3. The classification task is identical to FairMOT, requiring only the distinction between pedestrians and the background. The heat map prediction employed in the original algorithm is unsuitable for multilevel detection, as it requires a larger feature map to prevent center point offset. This paper introduces pixel-by-pixel detection for multilevel detection and proposes two methods, largest-area bounding box regression (box-largest) and adaptive pedestrian central sampling region (sampling-adaptive), specifically designed for the unique characteristics of pedestrian objects. The Re-ID component has been modified accordingly.

(1) Multilevel detection

The DLA-34_FPN provides various levels for multilevel detection of the MLD, and the subsequent step is to assign the objects to these levels according to the size of the bounding box. The idea of the hierarchical level of the MLD is to assign the smaller-scale pedestrian objects to the larger feature maps to extract more information, so as to prevent obscured or distant pedestrians from being missed.




(a) (b)

Fig. 4. (a) Example figure of the distances from the positivity sample point to the four edges of the bounding box and (b) schematic of the regression of the ambiguity point

Given the varying sizes at each level, the points on each feature level must be mapped back to the original image to accurately assess the object size during model training. Then learn the category information of each pixel point and the distance of each point to the left, top, right, and bottom edges of the object bounding box: $l^*$, $t^*$, $r^*$, and $b^*$, which allows the model to learn how to infer the bounding box information from the pixel features and to ensure good consistency between predictions from different pixel points. The range of sizes that need to be regressed for different feature levels is defined as $M_i$. $M_3$, $M_4$, $M_5$, $M_6$, and $M_7$ are (0, 64], (64, 128], (128, 256], (256, 512], and (512, $\infty$), respectively. If the $\max(l^*, t^*, r^*, b^*) \in M_i$, then the pedestrian detection box belongs to the level $i$, otherwise, it is set as the negative sample. The distances from the positive sample to the four sides of the bounding box are shown in Fig. 4 (a).

(2) Largest-area bounding box regression

After defining the range of regression objects at each level, this paper still encounters a scenario where a point needs to regress multiple bounding boxes for pedestrian objects. As shown in Fig. 4 (b), such a point is designated as the regression point. The MLD selects the pedestrian object with the largest bounding box area as the primary object for regression at that specific point. In video sequences, pedestrians that obscure other objects are usually closer to the camera and have a larger bounding box, and regression ambiguity typically arises on objects that are close to the camera. Hence, selecting the pedestrian object with a larger bounding box when the ambiguity point occurs is justifiable, as it enables the model to capture more precise appearance features, thereby facilitating the subsequent trajectory prediction and ID matching tasks.

(3) Adaptive pedestrian central sampling region

By utilizing each pixel point for classification and regression of the bounding box, this paper observes an increase in the number of positive samples. However, this methodology simultaneously introduces a significant amount of interfering information. When directly applied to the MOT of the pedestrian, the tracking performance will be affected [17]. FCOS [18] proposes a square central sampling region to

avoid introducing too much interfering information in the detection process. However, it fails to adapt to the boundary boxes of pedestrians with unequal height and width, which leads to inaccurate Re-ID feature selection that does not fully represent the object. To address this issue, the MLD proposes an adaptive pedestrian central sampling region method, which dynamically adjusts the central sampling region to a rectangular shape that conforms to the aspect ratio of the pedestrian boundary box. The adaptive pedestrian central sampling region sets the position coordinates of the central region as $(c_x - r's, c_y - rs, c_x + r's, c_y + rs)$, where $(c_x, c_y)$ is the central position coordinate of the real object boundary box. $r' = (w/h) \times r$, $h$ and $w$ are the height and width of the object boundary box, respectively, $r$ is a hyperparameter and sets to 2.

Fig. 5 shows the comparison between the square central sampling region and the adaptive pedestrian central sampling region. The large blue boxes demarcate the boundaries of the object bounding box, while the small yellow boxes depict the sampling region of the corresponding pedestrian object. The red circles highlight the incorrect features of the pedestrian objects. The adaptive pedestrian central sampling region contains more accurate pedestrian object features, and can adaptively adjust the center sampling range according to the aspect ratio of the pedestrian object bounding box, thereby ensuring that the sampling area corresponds to the shape of the object.

(4) Re-ID for the MD-TPFairMOT

The task of Re-ID also necessitates feature extraction from multiple levels. The size of the feature maps in each level is 1/8, 1/16, 1/32, 1/64, and 1/128 of the input image, respectively. Even when the feature maps from all levels are summed together, the resultant size is significantly smaller compared to the feature map outputted by FairMOT, which is 1/4 the size of the input image.

MD-TPFairMOT combines the tasks of object detection and Re-ID, which share the feature output of the backbone network. The combined loss function for the detection and Re-ID branches is used for joint training, and the overall loss function is

$$L_{total1} = \frac{1}{2}\left(\frac{1}{e^{w_1}} L_{\det} + \frac{1}{e^{w_2}} L_{identity} + w_1 + w_2\right) \quad (1)$$

where the $L_{identity}$ and $L_{\det}$ are the loss functions for detection and Re-ID, respectively. The $w_1$ and $w_2$ are learnable parameters that can be used to balance the conflict better between the detection and Re-ID.

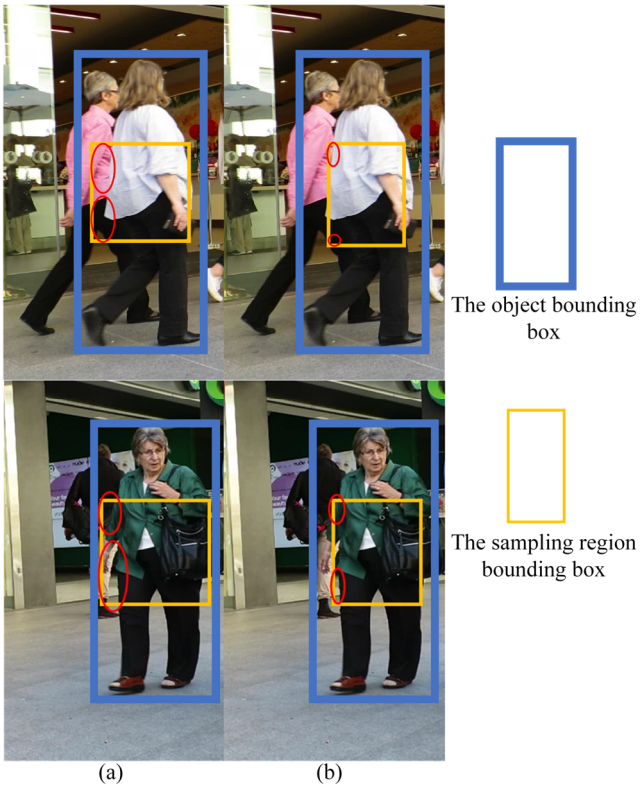



Fig. 5. The comparison of pedestrian sampling regions: (a) square central sampling region and (b) adaptive pedestrian central sampling region

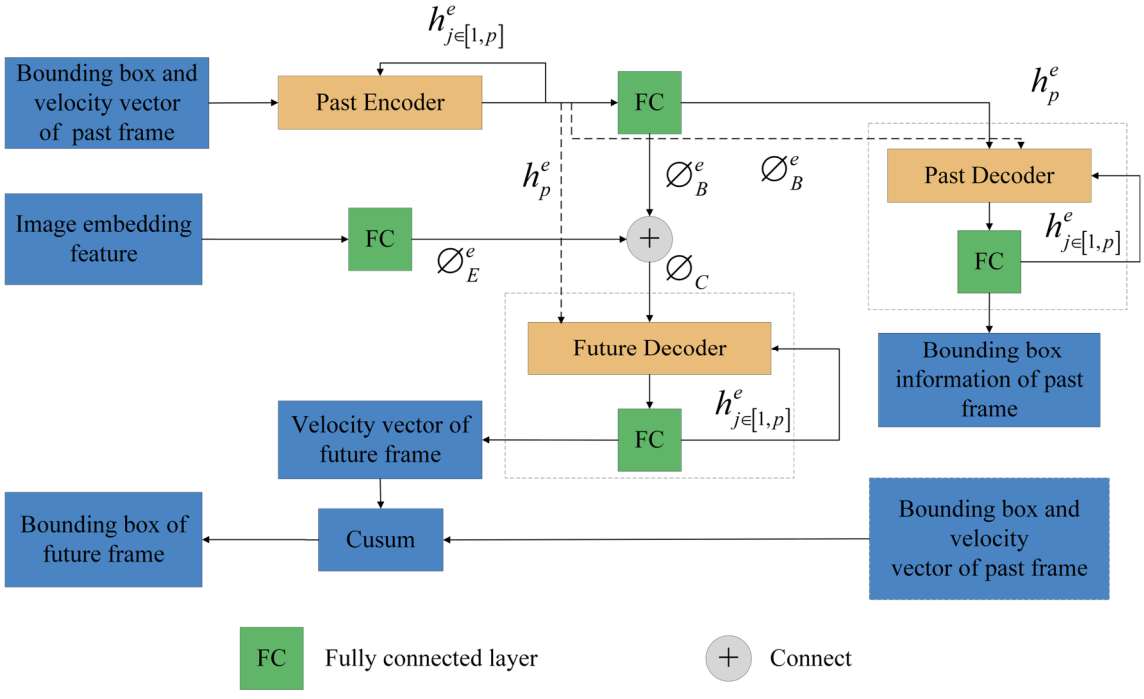



Fig. 6. The network structure of TP-LSTM

### D. TP-LSTM for the MD-TPFairMOT

The TP-LSTM method predicts the object's position in the future frame by learning the bounding box information of the object in the past frame. This innovative approach circumvents the constraints associated with the trajectory prediction capabilities of the Kalman filter used in FairMOT, particularly in terms of the object's motion state. Consequently, this prediction method enhances the accuracy of object position prediction, especially when the object is occluded. The network structure of the TP-LSTM is shown in Fig. 6. The functions and working processes of the modules in the network are described as follows.

**Step 1** The Past Encoder continuously iterates the bounding box position and velocity information of each object through LSTM [19]. At frame $t$, the bounding box information of the past $p$ frames of each object $k$ is expressed as set $\{b_{t-p}^k\}$, $\{b_{t-p+1}^k\}$, ⋯, $\{b_{t-1}^k\}$, and $b_t^k = (x_t^k, y_t^k, w_t^k, h_t^k, \Delta x_t^k, \Delta y_t^k, \Delta w_t^k, \Delta h_t^k)$, where $(x_t^k, y_t^k)$ denotes the center position of the corresponding bounding box, and $(w_t^k, h_t^k)$ denotes the width and height of the bounding box. $\Delta$ is the change between continuous time steps, and $(\Delta x_t^k, \Delta y_t^k, \Delta w_t^k, \Delta h_t^k)$ denotes the velocity information, The calculation is as follows.

$$\Delta g_t^k = g_t^k - g_{t-1}^k, \forall g \in \{x, y, w, h\} \tag{2}$$

**Step 2** The feature encoding level is used to encode the appearance features of the object. These features are extracted from the feature extraction branch of the DLA-34_FPN. The information provided by the feature encoding level provides crucial visual context information for the objects being predicted in the current frame. During the process of object trajectory prediction, utilizing the information of these appearance features not only enhances the accuracy of matching objects with similar appearances but also significantly improves the ability to match IDs when occluded objects reappear.

**Step 3** In order to check the accuracy of the past frame information learned by the LSTM network and promptly adjust the corresponding weights, the trajectory prediction network uses the Past Decoder to reconstruct the bounding boxes and velocities of the decoded past frames. The hidden state vector of the Past Decoder is initialized by the final result of the Past Encoder, and the hidden level state vector includes $h_p^e$ and memory cells in the LSTM. The output of the decoder is the predicted value of the past frame information, which can be expressed as $\hat{B} \in \mathbb{R}^{P\times 8}$. The L1 loss function is used to adjust the weight parameters of the Past Decoder as follows.

$$L_{past} = \frac{1}{K \times p \times 8} \sum_{k=1}^{K} \sum_{j=t-p}^{t-1} \left\| B_j^k - \hat{B}_j^k \right\| \tag{3}$$

where $p$ is the number of frames using the object information of the past frames, and $K$ is the total number of objects

**Step 4** The $K$ objects that have been detected are predicted by the Future Decoder to predict their velocities in future frames. As shown in Fig. 6, the coding vector $\varnothing_B^e$ of the bounding box and speed information of the past frame is connected with the embedding feature $\varnothing_B^e$ extracted by the DLA-34_FPN to synthesize the feature vector $\varnothing_C$. At each time step, the previous hidden state vector needs to be updated, and the predicted future frame velocity vector $\hat{V} \in \mathbb{R}^{q\times 4}$ is generated by the fully connected level, where $\hat{V}_t^k = (\Delta\hat{x}_t^k, \Delta\hat{y}_t^k, \Delta\hat{w}_t^k, \Delta\hat{h}_t^k)$.

**Step 5** The bounding box regression level regresses the bounding box of the object in the future frame, utilizing the predicted velocity vectors of the future frame and the known bounding box information of the previous frame. As shown in Fig. 6, the "Cusum" is used to calculate the cumulative sum of the predicted velocities of the future frames. The value of the "Cusum" and the information of the previous frame are used to calculate the predicted bounding box information $\hat{F} \in \mathbb{R}^{q\times 4}$ of the future frame, which $\hat{F} \equiv \{\{\hat{b}_t^k\}, \{\hat{b}_{t+1}^k\}, \cdots, \{\hat{b}_{t+q}^k\}\}$.

The calculation formula of the "Cusum" is

$$\hat{S}_j^K = \begin{cases} \hat{V}_j^k, & j = 1 \\ \hat{V}_j^k + \hat{S}_{j-1}^k, & j = 2 \cdots q \end{cases} \tag{4}$$

where $\hat{S}_j^k$ denotes the predicted position of the object in frame $j$. The calculation formula of the bounding box information of the future frame is

$$\hat{b}_{t+j-1}^k = b_{t-1}^k + \hat{S}_j^k, j = 1 \cdots q \tag{5}$$

The accurate measurement of future frame information prediction is consistent with the Past Decoder, L1 loss function is adopted, and the calculation formula of $L_{future}$ is as follows.

$$L_{future} = \frac{1}{K \times q \times 4} \sum_{k=1}^{K} \sum_{j=s}^{t+q} \left\| b_j^k - \hat{b}_j^k \right\| \tag{6}$$

So the loss function of the predicted branch can be defined as

$$L_{pre} = L_{past} + L_{future} \tag{7}$$

In the TP-LSTM branch, the inputs to the network consist of the object bounding box and velocity vector of the previous frame, and the image embedding is the feature extracted by the backbone network. The initial three frames of the past frame are initialized to zero, and then computed based on the position information of the bounding box of the first two frames to predict the position of the object in the future frame. The Past Encoder, Past Decoder, and Future Decoder utilize LSTM to encode and decode the object information of the past frames, thereby predicting the object information of the future frames. In addition, the proposed method dispenses with the need to use the bounding box information of the past frames at each level, instead, it feeds the image embedding information, derived from the integrated output features of the backbone network, into the trajectory prediction branch. By integrating object detection, Re-ID, and trajectory prediction, the Uncertainty loss function is employed for multi-task training. The formulation of this loss function is as follows.

**Algorithm 1:** The Fuse-association of MD-TPFairMOT

**Input:** The detection and Re-ID information: $D$ and $R$, the trajectory of previous frames: $T$, and the predicted trajectory box: $Box_{pre}$

**Output:** The matched trajectory: $T_{match}$ and the new trajectory: $T_{new}$

**Parameter:** $\lambda_1 = 0.7$, $time_{\max} = 30$, $\lambda_2 = 0.5$

**Initialization:** In the first frame, all detection results with confidence higher than the threshold are initialized as $T_{new}$

**for** number of frames **do**

Calculating cosine distance $d_c$ : $d_c = 1 - \cos(R, D)$

Calculating IOUdistance $d_i$ : $d_i = \dfrac{\left|Box_{pre} \cap D\right|}{\left|Box_{pre} \cup D\right|}$

Fuse $d_c$ and $d_i$ : $d_{fuse} = \lambda_1 d_c + (1 - \lambda_1) d_{i\min}$

First match: Hungarian matching $(d_{fuse}, T)$

Second match: IOU match the unmatched trajectory and the unmatched detection

Initialize the unmatched detection as $T_{new}$

Define the unmatched trajectory as $T_{lost}$

Predict the location of the $T_{lost}$ : $Box_{lost}$

Converting the number of lost frames $C_{lost}$ : $C_{lost} = time_{lost} / time_{\max}$

Calculate the consumption $cost$ : $cost = \lambda_2 Box_{lost} + (1 - \lambda_2) C_{lost}$

When the $cost < 0.55$, match $Box_{lost}$ and $T_{lost}$

**end for**

$$L_{total2} = \frac{1}{2}\left(\frac{1}{e^{w_1}} L_{\det} + \frac{1}{e^{w_2}} L_{id} + \frac{1}{e^{w_3}} L_{pre} + w_1 + w_2 + w_3\right) \quad (8)$$

where $L_{pre}$ is the loss function and $w_3$ is the learnable weight parameters for trajectory prediction.

### *E. Fuse-association for the MD-TPFairMOT*

The data association step of MD-TPFairMOT is accomplished through the collaboration of three tasks: object detection, Re-ID, and trajectory prediction. The cosine distance $d_c$ between the Re-ID features and detections and the IOU distance $d_i$ between the predicted positions by TP-LSTM and the detections are fused to match the existing trajectories with the recognizable objects in the current frame. Re-ID features may be less reliable in dense scenarios [20], and relying only on them will lead to too much object ID switching. Hence, if the minimum IOU distance between the detection result and the trajectory prediction is excessively large at the first matching, the cosine distance of the Re-ID feature can be increased appropriately. After matching Re-ID features and predicted positions, unmatched trajectories and detections are further correlated using IOU distance, with the Hungarian algorithm used for bisection matching. Unmatched detections after these two matches are initialized as new trajectories, while unmatched trajectories are assessed for matching or removal. The algorithmic pseudocode is outlined in Algorithm 1.

## III. Experimental Results and Analysis

### *A. Experiments Settings*

(1) Datasets

This article utilizes the identical training dataset, MOTChallenge, as employed by FairMOT, and proceeds to compare the tracking outcomes of our algorithm against those of FairMOT, as well as other one-step pedestrian multi-object tracking algorithms, across the MOT15, MOT16, MOT17, and MOT20 datasets. These datasets offer a diverse range of scenarios, including urban streets, pedestrian zones, and public transportation hubs, which pose different challenges for tracking algorithms. The exhaustive details concerning these datasets are presented in Table I.

(2) Evaluation indicators

The MOTChallenge evaluation standard is selected to evaluate the performance of the designed network, which drew on the CLEAR index and IDF1 value [21]. This benchmark provides a comprehensive set of metrics to assess tracking accuracy and robustness. In this evaluation framework, MOTA denotes the overall tracking accuracy, taking into account errors such as false positives, false negatives, and ID switches. It is a crucial metric that reflects the precision of the tracking algorithm in localizing and identifying objects. IDF1 indicates the proportion of objects with the correct ID among the total objects, emphasizing the consistency of object identities over time. MT denotes the proportion of hit trajectories to the total trajectories, and the hit trajectory is defined as the trajectories whose length is more than 80% of the true value. Conversely, ML denotes the proportion of lost trajectories to the total trajectories, and the lost trajectories are defined as the trajectories whose length is less than 20% of the true value. FPS stands for the frame rate of the tracking algorithm, which measures the speed of the algorithm in processing video frames. Additionally, IDs denotes the total number of object ID switches during the tracking process. A lower number of ID switches indicates that the algorithm maintains a more consistent and accurate tracking of object identities.

TABLE I
THE DETAILS OF DATASETS MOT15, MOT16, MOT17, AND MOT20

| Dataset | Number of Video sequences | Average length (s) | Average number of objects/ frame | Characteristic |
|---|---|---|---|---|
| MOT15 | 22 | 45.3 | 9.0 | The scene is diverse and low-density. |
| MOT16 | 14 | 33.1 | 20.9 | It improves the density of the object and standardizes the annotation. |
| MOT17 | 14 | 33.1 | 26.7 | More attention is paid to some difficult samples, and a larger number of public detectors are provided. |
| MOT20 | 8 | 66.9 | 156.8 | The density is the highest and the mutual occlusion is serious. |

TABLE II
THE RESULTS OF ABLATION EXPERIMENTS OF THE MLD

| Multilevel detection | Box-largest | Sampling-adaptable | MOTA↑ | IDF1↑ | IDs↓ |
|---|---|---|---|---|---|
| × | × | × | 67.1 | 70.9 | 441 |
| √ | × | × | 69.8 (++4.02%) | 71.6 (++0.98%) | 416 (--5.67%) |
| √ | √ | × | 70.2 (++0.57%) | 72.0 (++0.56%) | 364 (--12.5%) |
| √ | √ | √ | **70.6 (++0.57%)** | **72.3 (++0.42%)** | **335 (--7.97%)** |

(3) Parameter settings

The experimental environment includes Python3.7, CUDA11.2, and PyTorch 1.12.0. The GPU type is Nvidia GeForce RTX 2080Ti, and the operating system is Windows 11. On the basis of the COCO pre-trained model, train another 30 epochs, with a learning rate initially set to $10^{-4}$ and reduced by 10 times after 20 epochs, and the momentum factor set to 0.9. The input image is resized to 1088 × 608.

### *B. Experimental Results and Analysis*

(1) Experiment of MLD

The ablation experiments are conducted on the MLD branch, using 50% of the training set provided by the MOT17 dataset for training purposes, while the other remaining 50% for evaluation. The results are shown in Table II, where the "√" denotes that the corresponding algorithm is used, and "×" denotes that the corresponding algorithm is not used. "++" represents the percentage increase in performance compared to the previous experimental results, while "--" represents the percentage decrease. "↑" means a larger value for a better tracking effect, while "↓" means a smaller value for a better tracking effect, and the bold numbers represent the optimal results, the same as below.

From Table II, it is evident that after using the multilevel detection, the MOTA and IDF1 are improved by 4.02% and 0.98%, respectively, and the IDs is reduced by 5.67%. The largest-area bounding box regression contributes to a 0.57% increase in MOTA and a 0.56% increase in IDF1, while decreasing the number of ID switches by 12.5%. This indicates that the method is effective in significantly minimizing the number of ID changes for pedestrian objects. Incorporating the positive sample sampling region for adaptable pedestrians into the algorithm results in a 0.57% increase in MOTA and a 0.42% increase in IDF1, while reducing the number of ID switches by 7.97%. This approach effectively takes into account the unique characteristics of adaptable pedestrians during sampling, ensuring that the algorithm can better capture relevant features.The adaptive pedestrian central sampling area reduces the influence of ambiguity sampling points on the algorithm performance, thus improving the accuracy of Re-ID feature selection and reducing the number of pedestrian ID switches.

TABLE III
COMPARISON RESULTS OF DLA-34_FPN AND OTHER MULTI-SCALE BACKBONE NETWORKS

| Backbone Network | MOTA↑ | IDF1↑ | IDs↓ |
|---|---|---|---|
| Baseline | 67.1 | 70.9 | 441 |
| ResNet-34_FPN | 64.4 | 69.6 | 369 |
| ResNet-50_FPN | 65.1 | 70.1 | 355 |
| **DLA-34_FPN(ours)** | **70.6** | **72.3** | **335** |

To validate the superiority of DLA-34_FPN, this paper constructs multi-scale feature fusion architectures by combining ResNet-34 and ResNet-50 from the ResNet [22] series with the FPN, respectively, namely ResNet-34_FPN and ResNet-50_FPN, for comparison with DLA-34_FPN. Half of the training set provided by the MOT17 dataset was used for training, while the remaining half was utilized for evaluation. As shown in Table III, compared to the Baseline, the three multi-scale feature fusion backbone networks exhibit improvements in the MOTA, IDF1, and IDs metrics, with DLA-34_FPN demonstrating the largest enhancement. Specifically, its MOTA is 6.2 and 5.5 higher than that of ResNet-34_FPN and ResNet-50_FPN, respectively. Additionally, its IDF1 is 2.7 and 2.2 higher, while the number of IDs is reduced by 34 and 20, respectively. These results indicate that DLA-34_FPN has a significant advantage in improving tracking performance.

(2) Experiment of TP-LSTM and Fuse-association

In order to verify the performance improvement of the TP-LSTM and Fuse-association on FairMOT, comparative experiments are conducted on the trajectory prediction branch (TP branch) and data association branch (DA branch) of FairMOT on datasets MOT16, MOT17, and MOT20, respectively. In Table IV, The symbol "—" signifies the utilization of the original method in FairMOT and "TP-RNN" represents the substitution of LSTM with RNN within the TP-LSTM framework. "+" represents the percentage improvement in the performance compared to the original method, while "-" means the percentage decrease in the performance, the same as below.

As shown in Table IV, in dataset MOT16, the combined effect of the TP-LSTM and Fuse-association increased MOTA by 0.13%, IDF1 by 3.85%, and MT by 1.1%, and the method also reduced the number of ID switches by 44.3%.

The TP-LSTM is designed to address the problem of nonlinear motion prediction, and its performance surpasses other trajectory prediction methods, with an improvement of 1.65% in IDF1 and a reduction of 43.95% in IDs compared to the baseline. Fuse-association reduces the dependence of data association on appearance features. After using Fuse-association, the performance of each trajectory prediction method is improved, and the combined effect of TP-LSTM and Fuse-association is the best one. As Table IV demonstrates, similar patterns emerge with even more significant improvements in the MOT17 and MOT20 datasets. This is attributed to TP-LSTM's ability to better leverage its strength in predicting the trajectories of occluded pedestrians as crowd density increases, especially in the more densely populated MOT17 and MOT20 datasets, while Fuse-association effectively addresses more challenging pedestrian matching scenarios with high appearance similarity in these same datasets.

(3) Experiment of MD-TPFairMOT

In order to verify the gains of MLD, TP-LSTM, and Fuse-association in pedestrian MOT algorithms, 50% of the MOT17 training set is selected for training, and the remaining 50% of the data is used for evaluation. Ablation experiments are conducted on these three parts, and the results are shown in Table V. "+" represents the percentage increase in performance compared to FairMOT, and "-" represents the percentage decrease.

After adding the MLD, the tracking metrics increased by 5.1% MOTA and 2.0% IDF1, and decreased by 21.8% IDs. These results suggest that MLD can effectively solve the detection problem during occlusion and improve tracking accuracy. TP-LSTM is designed to address the challenge of position prediction following occlusion. With the addition of TP-LSTM, MOTA, and IDF1 increased by 1.01% and 2.23% respectively, and IDs decreased by 10.63%, indicating its significant effect. Fuse-association analyzes that pedestrians with high appearance similarity in dense scenes may lead to matching errors, and fuses two features for matching. After integrating the fuse-association, MOTA increased by 0.79%, IDF1 increased by 0.47%, and IDs decreased by 10.87%, which proves that it alleviates the problem of high appearance similarity.

The experimental validation of MD-TPFairMOT has been conducted using the MOT15, MOT16, MOT17, and MOT20 datasets, which are renowned benchmarks in the field of MOT. In a comparative analysis with several state-of-the-art one-step MOT methods, MD-TPFairMOT demonstrates superior performance, as evident from Table VI. Specifically, MD-TPFairMOT achieves the highest scores in both MOTA and IDF1, highlighting its robustness and accuracy in tracking pedestrian objects. The enhancement in detection accuracy achieved by the MLD method is a key factor contributing to MD-TPFairMOT's superior performance. By assigning pedestrians of different scales to separate detection levels, MLD effectively reduces missed detections and improves the precision of Re-ID feature selection. This, in turn, enhances the representation ability of MD-TPFairMOT for pedestrian objects, leading to more accurate tracking results. Furthermore, TP-LSTM addresses the challenges posed by the dynamic motion states of objects. By leveraging past frame information, including object bounding boxes and velocity data, TP-LSTM accurately predicts the future positions of pedestrian objects. This predictive capability enhances the robustness of MD-TPFairMOT in handling complex tracking scenarios, where the motion patterns of pedestrians can be highly unpredictable.

TABLE IV
COMPARISON RESULTS ON THE DATASETS MOT16, MOT17, AND MOT20 OF TP-LSTM AND FUSE-ASSOCIATION

| Dataset | TP branch | DA branch | MOTA↑ | IDF1↑ | MT↑ | IDs↓ |
|---|---|---|---|---|---|---|
| MOT16 | — | — | 74.9 | 72.8 | 44.7% | 1074 |
| | — | Fuse-association | **75.0 (+0.13%)** | 73.4 (+0.82%) | 45.0% (+0.3%) | 889 (-17.22%) |
| | TP-RNN | — | 74.0 (-1.2%) | 73.3 (+0.69%) | 44.2% (-0.5%) | 875 (-18.53%) |
| | TP-RNN | Fuse-association | 73.8 (-1.47%) | 75.0 (+3.02%) | 44.9% (+0.2%) | 719 (-33.06%) |
| | TP-LSTM | — | 74.3 (-0.8%) | 74.0 (+1.65%) | 44.7% (+0%) | 603 (-43.95%) |
| | TP-LSTM | Fuse-association | 74.2 (-0.93%) | **75.6 (+3.85%)** | **45.8% (+1.1%)** | **598 (-44.33%)** |
| MOT17 | — | — | 73.7 | 72.3 | 43.2% | 3303 |
| | — | Fuse-association | 73.6 (-0.14%) | 73.5 (+1.66%) | 43.8% (+0.6%) | 2703 (-18.17%) |
| | TP-RNN | — | 73.9 (+0.27%) | 73.7 (+1.93%) | 44.3% (+1.1%) | 2630 (-20.38%) |
| | TP-RNN | Fuse-association | 74.0 (+0.41%) | 74.0 (+2.35%) | 45.1% (+1.9%) | 2092 (-36.67%) |
| | TP-LSTM | — | 74.1 (+0.54%) | 74.8 (+3.46%) | 45.0% (+1.8%) | 2109 (-36.12%) |
| | TP-LSTM | Fuse-association | **74.4 (+0.95%)** | **75.3 (+4.15%)** | **46.0% (+2.8%)** | **1825 (-44.75%)** |
| MOT20 | — | — | 61.8 | 67.3 | 68.8% | 5243 |
| | — | Fuse-association | 61.4 (-0.65%) | 67.8 (+0.74%) | 69.2% (+0.4%) | 4269 (-18.56%) |
| | TP-RNN | — | 60.4 (-2.26%) | 67.3 (+0%) | **59.6% (-9.2%)** | 3532 (-32.45%) |
| | TP-RNN | Fuse-association | 60.2 (-2.59%) | 68.7 (+2.08%) | 59.8% (-9.0%) | 2780 (-0.74%) |
| | TP-LSTM | — | 61.2 (-0.97%) | 67.5 (+0.3%) | 61.5% (-7.3%) | 3025 (-46.95%) |
| | TP-LSTM | Fuse-association | **61.9 (+0.16%)** | **69.1 (+2.68%)** | 62.4% (-6.4%) | **2325 (-55.66%)** |

TABLE V
THE RESULTS OF ABLATION EXPERIMENTS OF MD-TPFAIRMOT

| MLD | TP-LSTM | Fuse-association | MOTA↑ | IDF1↑ | IDs↓ |
|---|---|---|---|---|---|
| × | × | × | 67.1 | 70.9 | 441 |
| √ | × | × | 70.6 (+5.1%) | 72.3 (+2.0%) | 335 (-21.8%) |
| √ | √ | × | 71.2 (+6.11%) | 73.9 (+4.23%) | 298 (-32.43%) |
| √ | √ | √ | **71.7 (+6.9%)** | **74.2 (+4.7%)** | **249 (-43.3%)** |

When evaluating the IDs of tracked objects, MD-TPFairMOT demonstrates the lowest ID count across all datasets, outperforming FairMOT and other methods significantly. The number of IDs directly reflects the ability of a tracking algorithm to maintain the consistency of object identities. In complex tracking scenarios, if an algorithm can stably assign a unique ID to each object and keep it unchanged throughout the entire tracking process, then the number of IDs will be relatively low. Lastly, the experimental results also reveal that MD-TPFairMOT maintains a competitive FPS despite its increased complexity. This balance between performance and efficiency is crucial for real-time applications, where both accuracy and speed are essential. Fig. 7 depicts a point-line chart that graphically illustrates the three key indicators—MOTA, IDF1, and IDs—extracted from Table VI. It is evident that MD-TPFairMOT exhibits superior overall performance across these three indicators on all four datasets, demonstrating its excellence in tracking accuracy, target identification capability, and identity linkage performance.

TABLE VI
THE EXPERIMENTAL COMPARATIVE RESULTS OF MD-TPFAIRMOT WITH OTHER ONE-STEP MOT METHODS

| Dataset | Tracking algorithm | MOTA↑ | IDF1↑ | MT↑ | ML↓ | IDs↓ | FPS↑ |
|---|---|---|---|---|---|---|---|
| MOT15 | TubeTK [23] | 58.4 | 53.1 | 39.3% | 18.0% | 854 | 5.8 |
| | JDE | 58.2 | 52.8 | 40.7% | 16.2% | 741 | 20.4 |
| | CTrackerV1 [24] | 59.8 | 51.7 | 39.0% | 16.8% | 672 | 7.2 |
| | CenterTrack [25] | 60.3 | 53.4 | 42.7% | 15.9% | 647 | 16.8 |
| | FairMOT | 60.6 | 64.7 | 47.6% | **11.0%** | 591 | **30.5** |
| | **MD-TPFairMOT (ours)** | **62.0** | **65.6** | **48.3%** | 17.2% | **535** | 20.7 |
| MOT16 | TubeTK | 64.0 | 59.4 | 33.5% | 20.4% | 854 | 1.0 |
| | JDE | 64.4 | 55.8 | 35.4% | 20.0% | 741 | 18.5 |
| | CTrackerV1 | 67.6 | 57.2 | 32.9% | 18.1% | 672 | 6.8 |
| | CenterTrack | 67.9 | 65.6 | 35.2% | 18.3% | 647 | 16.9 |
| | FairMOT | 74.9 | 72.8 | 44.7% | **15.9%** | 591 | **25.9** |
| | **MD-TPFairMOT (ours)** | **75.4** | **75.9** | **46.2%** | 19.8% | **535** | 16.3 |
| MOT17 | TubeTK | 63.0 | 58.6 | 31.2% | 19.9% | 4137 | 3.0 |
| | JDE | 63.3 | 59.7 | 31.8% | 23.0% | 5327 | 18.5 |
| | CTrackerV1 | 66.6 | 57.4 | 32.2% | 24.2% | 5529 | 6.8 |
| | CenterTrack | 67.8 | 64.7 | 34.6% | 22.6% | 2583 | 17.5 |
| | FairMOT | 73.7 | 72.3 | 43.2% | **17.3%** | 3303 | **25.9** |
| | **MD-TPFairMOT (ours)** | **75.2** | **75.6** | **46.9%** | 20.7% | **1673** | 16.2 |
| MOT20 | TubeTK | 50.3 | 49.3 | 52.4% | 13.8% | 4448 | 6.9 |
| | JDE | 55.7 | 50.4 | 56.4% | 10.4% | 4661 | 10.5 |
| | CTrackerV1 | 57.6 | 60.2 | 57.9% | 8.9% | 5197 | 6.8 |
| | CenterTrack | 58.8 | 59.3 | 61.6% | 9.8% | 4818 | 18.7 |
| | FairMOT | 61.8 | 67.3 | 68.8% | **7.6%** | 5243 | **23.2** |
| | **MD-TPFairMOT (ours)** | **61.5** | **68.0** | **62.2%** | 10.3% | **2220** | 11.5 |

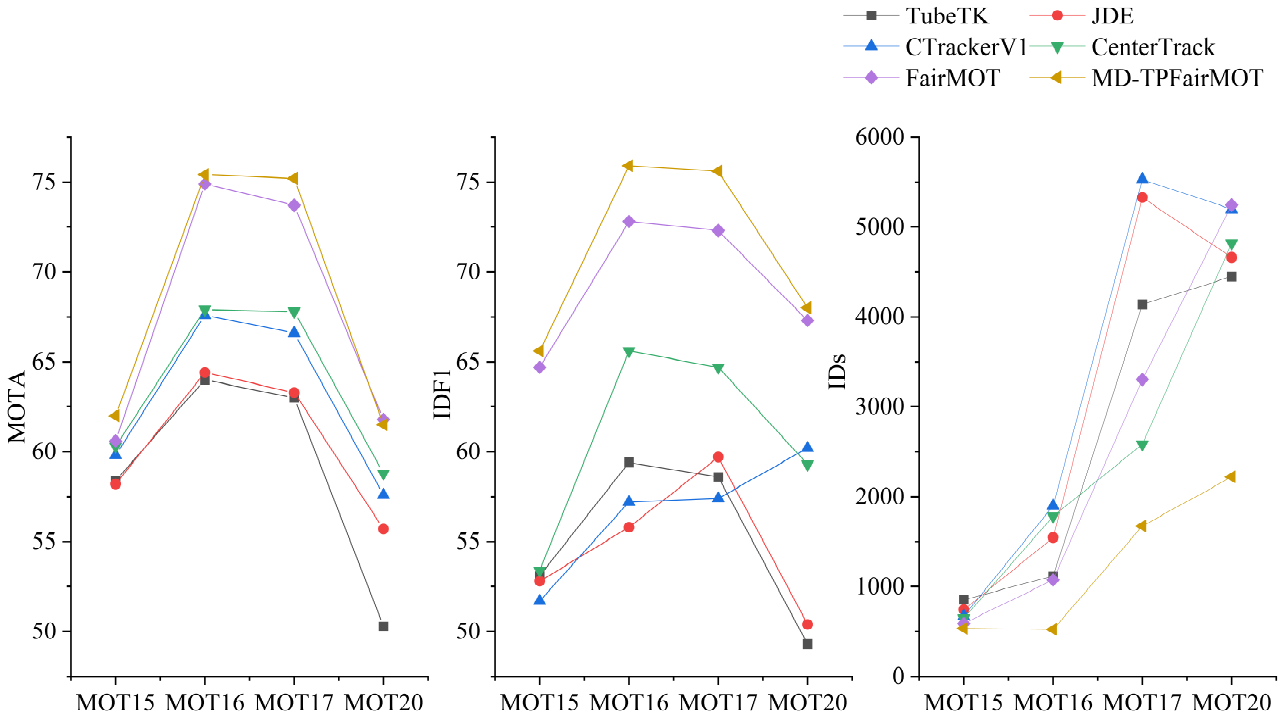



Fig. 7. The point-line chart comparing MD-TPFairMOT with other one-step methods in terms of MOTA, IDF1, and IDs indicators

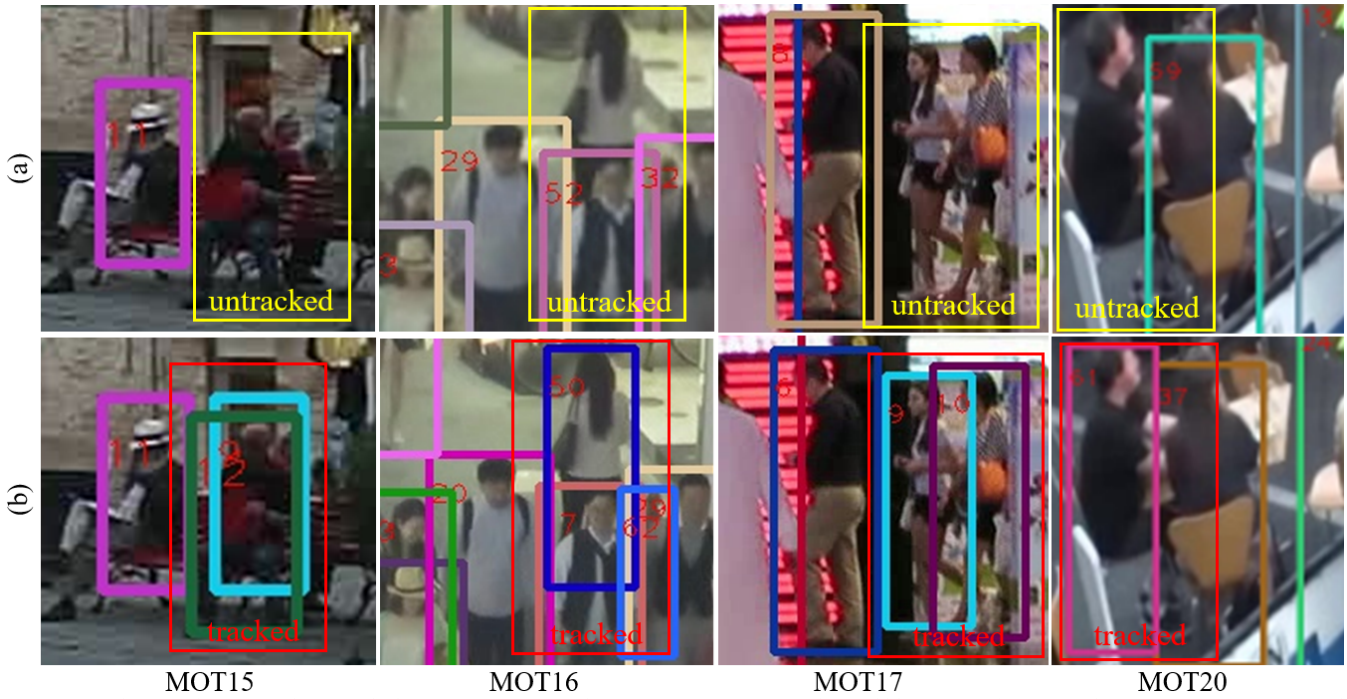



Fig. 8. Visual comparison of tracking results between FairMOT and MD-TPFairMOT in complex and blurry scenarios: (a) tracking images of FairMOT and (b) tracking images of MD-TPFaiMOT

To directly compare the tracking performance of MD-TPFairMOT and FairMOT under dense, occluded, and blurry conditions, Fig. 8 presents partial comparison images of the two algorithms on the MOT15, MOT16, MOT17, and MOT20 datasets. Fig. 8(a) shows the tracking results of FairMOT, with pedestrians that are not successfully tracked marked in yellow boxes. In contrast, Fig. 8(b) displays the tracking outcomes of MD-TPFairMOT, where the corresponding successfully tracked pedestrians are highlighted with red boxes. It is evident that in scenes with high density, occlusion, or blurring, FairMOT fails to correctly track some pedestrians, whereas MD-TPFairMOT is able to capture those that FairMOT misses. MD-TPFairMOT demonstrates a superior ability to track pedestrians more effectively and exhibits better performance in handling dense and occluded situations. Furthermore, when tracking pedestrians, it is necessary to consider not only the influence of other pedestrians but also the impact of the background environment on tracking performance. This indirectly demonstrates that our proposed algorithm improves the accuracy of ReID feature selection and pedestrian motion prediction for pedestrian objects, leading to more precise tracking of pedestrian objects.

## IV. Conclusion

This paper addresses the challenge of degraded tracking performance in dense scenes for the pedestrian tracking method FairMOT, stemming from frequent pedestrian occlusions and the high similarity of their appearance. By analyzing the characteristics of large-scale differences and uncertain motion states of pedestrians in dense scenarios, this paper proposes a new pedestrian MOT method, MD-TPFairMOT. The method uses MLD for hierarchical detection of pedestrians of different sizes, and its largest-area bounding box regression and adaptive pedestrian central sampling region are more in line with the characteristics of pedestrian occlusion, improving the accuracy of object detection. Simultaneously, using TP-LSTM for trajectory prediction and utilizing past frame information to predict pedestrian positions solves the problem of nonlinear motion prediction. In addition, Fuse-association is used to fuse Re-ID features and motion features to avoid incorrect matching caused by high appearance similarity. Experiments on the datasets MOT15, MOT16, MOT17, and MOT20 demonstrate that the proposed method offers improved accuracy and occlusion resistance in dense scenes.

## References

[1] Z. F. Hu, H. L. Yu, and K. H. Linghu, "Siamese network tracker based on dynamic convolution and attention fusion of shallow and deep information," *Engineering Letters*, vol. 32, no. 1, pp. 30-42, 2024.

[2] A. Gullapelly and B.G. Banik, "Multiple object tracking with behavior detection in crowded scenes using deep learning," *Journal of Intelligent & Fuzzy Systems*, vol. 44, no. 3, pp. 5107-5121, 2023.

[3] S. H. Park, B. D. Kim, C. M. Kang, C. C. Chung, and J. W. Choi, "Sequence-to-sequence prediction of vehicle trajectory via LSTM encoder-decoder architecture," *IEEE Intelligent Vehicles Symposium (IV)*, June 26–30, 2018, Changshu, Suzhou, China, pp. 1672-1678.

[4] X.P. Dai, "Visual recognition and performance prediction of athletes based on target tracking EIA algorithm," *Journal of Intelligent & Fuzzy Systems*, vol. 40, no. 4, pp.7233-7246, 2021.

[5] D. Merad, K. E. Aziz, R. Iguernaissi, B. Fertil, and P. Drap, "Tracking multiple persons under partial and global occlusions: Application to customers' behavior analysis," *Pattern Recognition Letters*, vol. 81, no. 1, pp. 11-20.

[6] A. Bewley, Z. Y. Ge, L. Ott, F. Ramos, and B Upcroft, "Simple online and realtime tracking," 2*016 IEEE International Conference on Image Processing (ICIP)*, 25-28 September, 2016, Phoenix, Arizona, USA, pp. 3464-3468.

[7] N. Wojke, A. Bewley, D. Paulus, "Simple online and real-time tracking with a deep association metric," *2017 IEEE International Conference on Image Processing (ICIP)*, 17-20 September, 2017, Beijing, China, pp. 3645-3649.

[8] S. J. Sun, N. Akhtar, H. S. Song, A. Mian, and S. Mubarak, "Deep affinity network for multiple object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 1, pp. 104-119, 2021.

[9] F. Zeng, B. Dong, Y. Zhang, T. Wang, X. Y. Zhang, et al, "Motr: end-to-end multiple-object tracking with transformer," *European Conference on Computer Vision*, 23-27 October, 2022, Tel Aviv, Israel, pp. 659-675.

[10] P. Voigtlaender, M. Krause, A. Osep, J. Luiten, B. B. G. Sekar, et al, "Mots: Multi-object tracking and segmentation," *Proceedings of The ieee/cvf Conference on Computer Vision and Pattern Recognition*, 16-20 June, 2019, Long Beach, CA, USA, pp. 7942-7951.
[11] Z. D. Wang, L. Zheng, Y. X. Liu, S. J. Wang, "Towards real-time multi-object tracking," *European Conference on Computer Vision*, 23-27 October, 2020, Glasgow, UK, pp.107-122.
[12] Y. Zhang, C. Wang, X. Wang, W. Zeng, and Y. Wen, "Fairmot: On the fairness of detection and re-identification in multiple object tracking," *International Journal of Computer Vision*, vol. 129, pp. 3069-3087, 2021.
[13] B. Liu, Z. M. Wang, W. Y. Chen, and J. X. Wang, "Trajectory prediction combined with FairMOT for multi-object tracking," *Eighth International Symposium on Advances in Electrical, Electronics and Computer Engineering (ISAEECE 2023)*, 31 May, 2023, Hangzhou, China, pp. 693-696.
[14] K. W. Duan, S. Bai, L. X. Xie, H. G. Qi, Q. M. Huang, et al, "Centernet: Keypoint triplets for object detection," *Proceedings of The IEEE/CVF International Conference on Computer Vision*, 15-20 June, 2019, Long Beach, CA, USA, pp. 6569-6578.
[15] O. S. Berot, H. Canot, P. Durand, B. Hassoune-Rhabbour, H. Acheritobehere, C. Laforet, V. Nassiet, "Choice of parameters of an LSTM network, based on a small experimental dataset," *Engineering Letters*, vol. 32, no. 1, pp. 59-71, 2024.
[16] T.Y. Lin, P. Dollár, R. Girshick, K. M. He, B. Hariharan, et al, "Feature pyramid networks for object detection," *Proceedings of The IEEE Conference on Computer Vision and Pattern Recognition*, 21-26 July, 2017, Honolulu, HI, USA, pp. 2117-2125.
[17] F. Yu, D. Wang, E. Shelhamer, and T. Darrell, "Deep level aggregation," *Proceedings of The IEEE Conference on Computer Vision and Pattern Recognition*, 18-23 June, 2018, Salt Lake City, UT, USA, 2018. pp. 2403-2412.
[18] Z. Tian, C. Shen, H. Chen, and T. He, "FCOS: A simple and strong anchor-free object detector," *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 15-20 June, 2019, Long Beach, CA, USA, pp. 9627-9636.
[19] Y. Yuan, L. Lin, L. Z. Huo, Y.L. Huo, Y. L. Kong, et al, "Using an attention-based LSTM encoder-decoder network for near real-time disturbance detection," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 13, pp. 1819-1832, 2020.
[20] X. P. Chen and Y. Xu, "A Multi-Dimensional Attention Feature Fusion Method for Pedestrian Re-identification," *Engineering Letters*, vol. 31, no. 4, pp. 1365-1373, 2023.
[21] E. Ristani, F. Solera, R. Zou, R. Cucchiara, and C. Tomasi, "Performance measures and a data set for multi-object, multi-camera tracking," *European Conference on Computer Vision*, 11-14 October, 2016, Amsterdam, The Netherlands. 2016. pp. 17-35.
[22] K. He, X. Zhang, S. Ren, et al, "Deep residual learning for image recognition," *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770-778, June, 2016, Las Vegas, NV, USA pp.1063-6919.
[23] B. Pang, Y. Li, Y. Zhang, M. Li, C. Lu, "Tubetk: Adopting tubes to track multi-object in a one-step training model," *Proceedings of The IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13-19 June, 2020, Seattle, WA, USA, pp.6308-6318.
[24] J. L. Peng, C. G. Wang, F. B. Wan, Y. Wu, Y. B. Wang, et al, "Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking," *Computer Vision–ECCV 2020*, 23-28 August, 2020, Glasgow, UK, pp.145-161.
[25] X. Zhou, V. Koltun, and P. Krähenbühl, "Tracking objects as points," *European Conference on Computer Vision*, 23-27 October, 2020, Glasgow, UK, pp.474-490.