Learning key lines for multi-object tracking

Yi-Fan Li [a], Hong-Bing Ji [a,*], Xi Chen [b], Yong-Liang Yang [b], Yu-Kun Lai [c]

[a] School of Electronic Engineering, Xidian University, Xi'an Shaanxi Province, 710071, China
[b] Department of Computer Science, University of Bath, Bath, BA2 7AY, United Kingdom
[c] School of Computer Science and Informatics, Cardiff University, Cardiff, CF24 4AG, United Kingdom

ARTICLE INFO

Communicated by Michael Felsberg

MSC:
41A05
41A10
65D05
65D17

Keywords:
Multi-object tracking
Base representation
Key lines

ABSTRACT

Most online multi-object tracking methods utilize bounding boxes and center points inherited from detectors as the base models to represent targets. Limited performance is obtained with these base models alone for tracking. Complex networks are generally applied on top to extract high-level discriminative features such as appearance embeddings and motion predictions for data association. However, the weakness in the feature representation of bounding boxes and center points degrades the tracking performance. In this paper, we propose a novel base model that represents targets with key lines for tracking, which can provide discriminative features and accurate target affinity measurements. Besides, we use the proposed key lines to select low-scored detections and unmatched tracks to recover missed targets and enhance identity consistency. Based on this, we apply the proposed line-based modeling strategy to existing trackers and propose a line-based Cascade Tracking algorithm to associate targets in three stages, and very competitive results are achieved on MOTChallenge benchmarks. Extensive experiments with improved performances demonstrate the effectiveness and generalization of key lines in providing discriminative features and enhancing tracking performance.

1. Introduction

Multi-object tracking (MOT) is one of the fundamental problems in computer vision with various applications such as video surveillance and autonomous vehicles. The tracker localizes objects of interest in video frames and links identical objects to form trajectories. Thus, the MOT algorithm can be divided into two tasks: object detection and data association. Although prominent progress has been achieved, tracking with high accuracy and efficiency is still very challenging.

With the advances in object detection, many MOT methods follow the tracking-by-detection (TBD) paradigm, where objects are obtained by off-the-shelf detectors and associated with existing tracks to form trajectories across frames. Methods of this paradigm usually utilize extra networks to extract high-level features for data association, including appearance features and motion predictions. Despite the superior performance, separating detection and data association impedes the TBD paradigm from end-to-end tracking. The joint-detection-and-tracking (JDT) paradigm has been proposed to mitigate the separation issue in TBD, where detection and feature extraction are unified in a single network and performed simultaneously. Most JDT trackers are developed from detectors by adding tracking-related sub-networks, and state-of-the-art (SOTA) results are achieved. For instance, Tracktor (Bergmann et al., 2019) and QDTrack (Pang et al., 2021) are developed from Faster R-CNN (Ren et al., 2015), CenterTrack (Zhou et al., 2020) and FairMOT (Zhang et al., 2021) are transformed from CenterNet (Zhou et al., 2019). Two commonly used target base representations, including bounding boxes and center points, are naturally inherited from detectors to represent targets and extract features. As shown in Fig. 1, we denote the bounding boxes and center points as base models, which apply to TBD and JDT paradigms but are produced at different stages.

Tracking can be performed only with base features (including positional relations between target bounding boxes and center points), e.g., based on bounding box overlaps (Bewley et al., 2016; Bochinski et al., 2017), but achieves limited performance in crowded scenes. Thus, high-level features, including appearance embeddings and motion predictions, are typically utilized for enhancing the accuracy of similarity measurement and identity (ID) assignment, and base features are typically employed to provide accessory information where high-level feature fails. Recent methods primarily focus on generating discriminative appearance features and predicting reliable positions, and the representation of base models is ignored.

We argue that the existing base models are not optimal for MOT due to the following observations. As shown in Fig. 2(a), in box-based representation, the target bounding box significantly overlaps with nearby targets in crowded scenes, which can be wrongly suppressed by post-processing in the tracking procedure, thus vulnerable to occlusions. Besides, bounding boxes can capture features of surrounding targets and backgrounds, reducing the temporal consistency and reliability of

* Corresponding author.
E-mail address: hbji@xidian.edu.cn (H.-B. Ji).

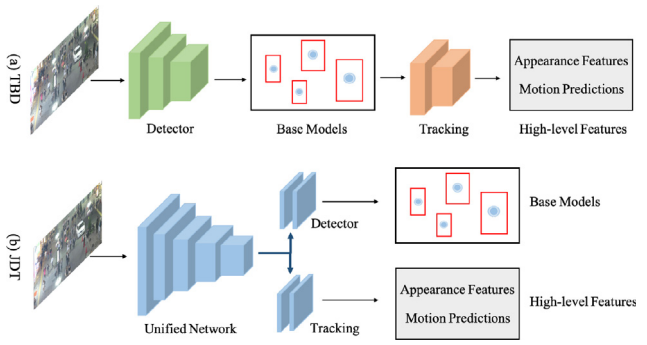

Fig. 1. Comparison of the TBD and JDT paradigms regarding the base models and high-level features extraction. For TBD, the base models are obtained from the detector first, and then extra networks are used to produce high-level features. For JDT, a unified network simultaneously produces base models and high-level features.

Fig. 2. Comparison of different base models. (a) The box-based model uses bounding boxes to describe targets. (b) The point-based model represents targets by center points. (c) The proposed key line comprises the center and top points and a line linking them, which is located inside the target and can provide discriminative features.

target appearance embeddings. On the other hand, the point-based representation shown in Fig. 2(b) only accesses position information with the center points. Extracting center-based identity embeddings is widely adopted, which can be contaminated by occlusions and camera motion. Furthermore, coarse base features provided by base models are normally employed with high-level features (Zhang et al., 2021; Zhou et al., 2020) for similarity measurement, resulting in false identity assignments in crowded scenes.

In this paper, we propose a novel base model for targets in MOT, in which the target is represented by a key line formed by the center point, top point, and the line linking them, as shown in Fig. 2(c). The centers and tops are the two most prominent and consistent positions for targets. One can recognize each target from the crowds with these points easily. Unlike bounding boxes and center points, the proposed key lines are more representative since they are located inside the targets and are associated tightly with the position and size of the targets. Thus, they can provide accurate and informative base features. Besides, high-level features captured based on key lines are more discriminative in crowded scenes, leading to accurate affinity measurements and identity assignments. Experiments demonstrate that the identity consistency of existing trackers can be enhanced by applying the proposed key lines.

We establish key lines by predicting the centers and tops in a unified network, LineNet. A point grouping scheme is introduced to process the predicted positions to construct key lines. Besides, a line-based measuring and scoring strategy is presented to calculate similarity with base features provided by key lines. Meanwhile, the low-scored detections and unmatched tracks are selected and reused, which can help to recover missed targets and enhance the features of targets. Finally, a novel line-based Cascade Tracking algorithm is proposed to exploit the base and high-level features produced based on key lines, where targets are associated in a cascade way.

We apply the proposed key lines to six state-of-the-art trackers by replacing their original base models, and notable improvements in IDF1 are achieved for all trackers, demonstrating that the key line is a generic base representation with superior generalization and can help to improve the identity consistency of tracks. Furthermore, by integrating the proposed line-based modeling strategy with existing methods, we propose two trackers, namely CenterLine and FairLine, by predicting line-based displacements and extracting line-based appearance embeddings. Significant improvements and very competitive performances are achieved on the MOTChallenge benchmark, showing that line-based features are more discriminative and robust for tracking.

In summary, this paper makes the following contributions:

- We propose a novel base representation in which targets are represented by key lines.
- We propose a line-based measuring strategy for similarity measurement and identity assignment based on discriminative features of key lines.
- We propose a detection and key track selection strategy based on key lines to recover missed targets and enhance the features of targets.
- We apply key lines to two existing trackers to extract line-based high-level features and propose a novel Cascade Tracking algorithm to associate targets and handle challenging scenarios.

The rest of the paper is organized as follows. Related works are reviewed in Section 2. The algorithmic details are introduced in Section 3. The experimental results and discussions are presented in Section 4. Section 5 summarizes this work.

2. Related work

The close relationship with object detection enables MOT to benefit from significant progress in detection while inheriting the representation methods used in detectors. We review representation methods in detection and tracking below.

2.1. Box-based representation

Many MOT methods adopt off-the-shelf detectors, where the targets of interest are localized and represented with bounding boxes. In earlier approaches, overlaps of the bounding boxes are used as metrics in IOUTracker (Bochinski et al., 2017) and SORT (Bewley et al., 2016) for high-speed tracking. DeepSORT (Wojke et al., 2017) improves SORT by adopting the ReID models and associates targets with appearance features extracted from the target bounding boxes. Recent MOT methods follow the JDT framework to pursue an end-to-end tracking paradigm with box-based representations. Trakctor (Bergmann et al., 2019) is developed from Faster R-CNN (Ren et al., 2015) by reusing the bounding box regression head. QDTrack (Pang et al., 2021) adds an embedding branch on top of Faster R-CNN to extract appearance features from bounding boxes. SiamMOT (Shuai et al., 2021) builds region-based single object trackers upon Faster R-CNN to predict target positions by reusing box-based region features.

These methods directly inherit the bounding box representation and box-based feature extraction, thus are vulnerable to occlusions and distractors caused by contaminated regional visual features inside the bounding boxes and coarse post-processing procedure Non-Maximum-Suppression (NMS). Efforts have been made to address these drawbacks. Attention mechanisms are adopted in TADAM (Guo et al., 2021), guiding the tracker to focus more on targets inside bounding boxes and less on distractors. An occlusion handling strategy is proposed in TMOH (Stadler and Beyerer, 2021) that models the occluding and occluded tracks to improve box-based identity management. However, these methods rely heavily on detectors and are hard to generalize to other methods. The proposed key lines can provide discriminative target features and help enhance the robustness of existing trackers in crowded scenes.

2.2. Point-based representations

Objects are detected by locating center positions in the point-based detectors (Zhou et al., 2019), and point-based representations and feature extractions are also inherited. The affinity between targets is typically calculated by extracting center-based identity embeddings and predicting center-based motion predictions since center positions can only provide limited spatial information. CenterTrack (Zhou et al., 2020) adds a center-based offsets prediction branch on top of CenterNet (Zhou et al., 2019) to update the target position. FairMOT (Zhang et al., 2021) adds a ReID branch upon CenterNet to extract center-based identity embeddings. SOTMOT (Zheng et al., 2021) is built from CenterNet by adding a single object tracker branch and treating objects as points while tracking.

Despite the simplicity, center-based features are vulnerable to occlusions and small targets, and efforts have been made to enhance the robustness of center-based methods. PermaTrack (Tokmakov et al., 2021) proposes a spatial–temporal recurrent memory module to predict occluded target center locations. Besides, learnable points are proposed to improve the center-based modeling. MTrack (Yu et al., 2022a) proposes representing targets as multiple adaptively selected key points to enrich the representative features. AdaMOT (Liang et al., 2022a) uses a set of learnable points as the target descriptor for robust feature extractions. Despite their effectiveness, these methods require dedicated networks and complex training strategies and are hard to generalize to other methods. However, the proposed key lines can enhance the identity consistency of existing trackers by simple replacement without re-training, and the line-based modeling strategy can be generalized to existing methods for better tracking performance.

2.3. Other representations

There are some other base representations used in detection and tracking. TubeTK (Pang et al., 2020) introduces bounding tubes that combine the spatial–temporal locations of objects by linking target bounding boxes. However, the bounding tube process tracks offline with sizeable computational complexity. This tube-based representation is also employed in video object detection (Kang et al., 2017; Tang et al., 2019). Besides, the objects are detected by locating the top-left and the bottom-right corners in CornerNet (Law and Deng, 2018), and it requires an additional Associative Embedding method (Newell et al., 2017) to group the points into target positions. Moreover, the learnable points representation is proposed in RPT (Ma et al., 2020) for visual tracking, in which the target is represented as a set of representative points. The dense points method in RPT is unsuitable for MOT since the postures of targets are comparatively simple, and the number of targets is unlimited in MOT. Thus, this point set will introduce additional confusion and computational burdens in crowded scenes for MOT.

3. Methodology

In this section, we first introduce the key lines learning network, LineNet, line-based measuring and scoring strategies. We then illustrate the detection and key track selection strategies. Finally, we apply key lines to existing trackers and introduce the proposed Cascade Tracking algorithm.

3.1. Key lines learning network

We build LineNet based on CenterNet, which has three output branches, including a heatmap of center positions H^t_{cent}, the predictions of center offset off^t_{cent}, and the sizes of targets, i.e., widths and heights. The center offsets refine the center positions to reduce the influence of network downsampling. An overview of our network is shown in Fig. 3, where two consecutive frames and a single-channel heatmap H^{t-1}_{in} are used as input. The input heatmap is rendered by encoding centers of key tracks of the previous frame. The key tracks denote true positive target tracks, selected with the proposed selection strategy introduced in Section 3.4. The input heatmap incorporates the temporal and historical information of tracks in past frames, and it helps to enhance target features despite their matching states, guiding the network to localize occluded targets accurately.

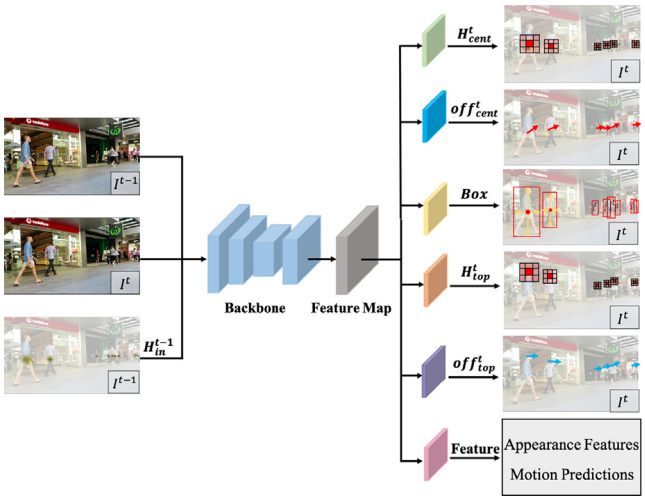

Fig. 3. Overview of the proposed LineNet, which takes two consecutive frames along with an input heatmap as inputs. The outputs contain heatmaps and offsets of the centers and tops, the size of targets. Line-based high-level features can be obtained by adding corresponding networks.

As shown in Fig. 3, we add two prediction branches on top of backbone features to obtain the heatmap H^t_{top} and offsets off^t_{top} of top positions. These branches are constructed similarly to center prediction networks by stacking convolutional layers. The centroids of top lines in ground truth bounding boxes are rendered with the Gaussian functions to form training labels. The rendered heatmap $\hat{R}_{xy}$ at the location (c_x, c_y) is:

$$\hat{R}_{xy} = \exp\left(-\frac{(x-c_x)^2 + (y-c_y)^2}{2\sigma^2}\right), \tag{1}$$

where the standard deviation value σ is a function that changes according to the target size (Zhou et al., 2019). The Gaussian functions in Eq. (1) are also used for input heatmap rendering by encoding the centers of key tracks. We use Focal Loss (Lin et al., 2017) to calculate the training objective of the top prediction heatmap as follows:

$$L^t_{xy} = -\frac{1}{N}\sum_{xy}\begin{cases}(1-R_{xy})^{\alpha}\log R_{xy}, & if\ \hat{R}_{xy}=1,\\ (1-\hat{R}_{xy})^{\beta}(R)^{\alpha}_{xy}\log(1-R_{xy}), & otherwise.\end{cases} \tag{2}$$

in which R_{xy} is the estimated heatmap at location (c_x, c_y), N denotes the number of objects, α and β are set to 2 and 4, which are hyper-parameters of the Focal Loss. The training objectives of the top offsets prediction are calculated with L1 Loss, the same as the center offsets prediction of CenterNet.

The centers and tops of targets are detected based on the response on heatmaps H^t_{cent} and H^t_{top}, respectively, and key lines can be constructed with a grouping scheme. Moreover, to extract discriminative line-based high-level features, we add corresponding sub-networks, as shown in Fig. 3, and build two line-based trackers on top of existing methods.

3.2. Constructions of key lines

On frame I^{t-1}, existing tracks are denoted as $\mathcal{T}^{t-1} = \{T^{t-1}_1, T^{t-1}_2, \ldots\}$, where t is the time step, and T^{t-1}_k represents the kth track that contains a series of bounding boxes with the same identity. The detections of

the frame I^t are a set of bounding boxes denoted as $\mathcal{D}^t = \{b_1^t, b_2^t, \dots\}$, where the ith bounding box is represented by $b_i^t = (x_i^t, y_i^t, w_i^t, h_i^t)$, i.e., the center coordinates, width, and height. With two adjacent frames and a rendered heatmap as input, the LineNet outputs the heatmaps of the centers H_{cent}^t and the tops H_{top}^t of the frame I^t. The width w and height h of each target are also obtained.

The proposed key line is the one that can represent the target with two prominent points, namely the center point and the top point, as well as the line linking them. Key lines remain smooth and continuous in movement and are easily identified in crowds. Given the outputs of LineNet, we can obtain a set of centers $C_{cent}^t = \{c_1^t, c_2^t, \dots, c_N^t\}$ from the H_{cent}^t with NMS, where N is the number of centers. Similarly, M top positions $T_{top}^t = \{t_1^t, t_2^t, \dots, t_M^t\}$ are obtained from the heatmap H_{top}^t. Ideally, the number of centers and tops are equal if all target representative points are found and can be matched correctly. However, occlusion, distractors, and camera motion will lead to noisy detections and missed targets, resulting in missing representative points. Therefore, we propose a point grouping scheme to process the predicted points to build key lines for targets.

To be more specific, given N detected center points C_{cent}^t, M detected top points T_{top}^t, and estimated heights of all targets, where the ith center point is represented by horizontal and vertical coordinates as $c_i^t = \left(x_{cent,i}^t, y_{cent,i}^t\right)$, the ith top point being $t_i^t = \left(x_{top,i}^t, y_{top,i}^t\right)$, and its height is h_i^t. We can calculate the top $ct_i^t = \left(x_{ct,i}^t, y_{ct,i}^t\right)$ for each detected center by referring to the corresponding height, where $x_{ct,i}^t = x_{cent,i}^t$, $y_{ct,i}^t = y_{cent,i}^t + \frac{h_i^t}{2}$. Doing so allows us to obtain the calculated top set $T_{ct}^t = \{ct_1^t, ct_2^t, \dots, ct_N^t\}$ for all predicted centers.

Detections of high quality should accurately estimate the position and size of targets. We can associate the detected tops T_{top}^t and calculated tops T_{ct}^t of the same target based on spatial similarity. The detected and calculated tops are spatially close to each other for high-quality localization. Thus, we compute the pairwise Euclidean distance between every top point in the predicted and calculated tops and link them with the greedy algorithm. After that, we evaluate the matching pairs as below:

$$d^{i,j,t} = \sqrt{\left(x_{top,i}^t - x_{ct,j}^t\right)^2 + \left(y_{top,i}^t - y_{ct,j}^t\right)^2}, \tag{3}$$

$$s^{i,j,t} = \exp\left(-\frac{d^{i,j,t}}{\left|y_{cal,i}^t - y_{ct,i}^t\right|}\right), \tag{4}$$

where $d^{i,j,t}$ is the Euclidean distance between the ith predicted top $(x_{top,i}^t, y_{top,i}^t)$ and the jth calculated top $(x_{ct,j}^t, y_{ct,j}^t)$, $s^{i,j,t}$ reveals the matching quality of two linked points, normalized by the corresponding size. The ith predicted top and the jth calculated top are considered successful matching if they are linked by the greedy algorithm and their matching score $s^{i,j,t}$ is higher than a predefined threshold η_1. We use the predicted tops to build key lines if matched with predicted centers, and the calculated tops are used to build key lines for the unmatched centers. By doing so, all objects are represented with key lines.

3.3. Line-based measuring and scoring

Line-based Measuring. A key line is denoted as $L_i^t = \left(c_i^t, t_i^t, l_i^t, v_i^t\right)$, where $c_i^t = \left\{x_{cent,i}^t, y_{cent,i}^t\right\}$ and $t_i^t = \left\{x_{top,i}^t, y_{top,i}^t\right\}$ are the center and top coordinates, respectively. l_i^t and v_i^t represent the length and velocity of this key line. The velocity of the key line is denoted as $v_i^t = \left(\Delta v_{x,i}^t, \Delta v_{y,i}^t, \Delta v_{l,i}^t\right)$, representing the changes in position and length. The velocity is updated after matching each frame with the moving average strategy to prevent extreme changes, which can be calculated as follows:

$$\Delta v_{x,i}^t = 0.8 \cdot \Delta v_{x,i}^{t-1} + 0.2 \cdot \left(x_{cent,i}^t - x_{cent,i}^{t-1}\right), \tag{5}$$

$$\Delta v_{y,i}^t = 0.8 \cdot \Delta v_{y,i}^{t-1} + 0.2 \cdot \left(y_{cent,i}^t - y_{cent,i}^{t-1}\right), \tag{6}$$

$$\Delta v_{l,i}^t = 0.8 \cdot \Delta v_{l,i}^{t-1} + 0.2 \cdot \left(l_i^t - l_i^{t-1}\right). \tag{7}$$

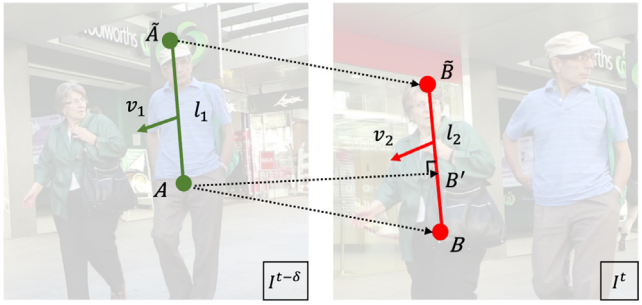

Fig. 4. The spatial distance between an existing track (shown in the green key line $A\tilde{A}$) and a detected target (shown in the red key line $B\tilde{B}$). l_1 and l_2 are the lengths of the two key lines, and v_1 and v_2 are corresponding velocities. Only the measurements of two targets are shown for clarity. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

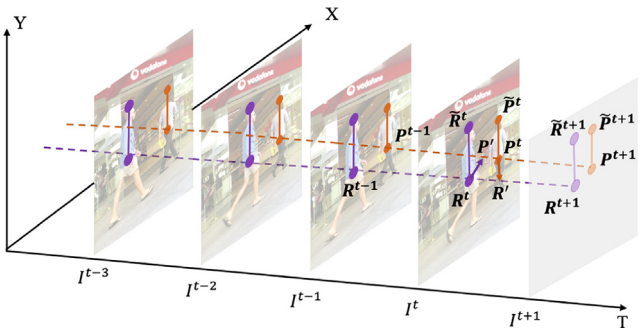

Fig. 5. The temporal motion model. We build a temporal motion model with target key lines to model the dynamics of tracks with five frames under the constant velocity assumption. The dashed lines represent trajectories, and purple and orange circles represent different IDs. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

As shown in Fig. 4, the existing track on the δ previous frame $I^{t-\delta}$ is represented by a green key line $A\tilde{A}$, and a red key line $B\tilde{B}$ represents a newly detected object on the frame I^t. The line-based spatial distance between these two targets is calculated as follows:

$$d_{spa} = \|AB\| + \left\|\tilde{A}\tilde{B}\right\| + \left\|AB'\right\| + \|l_1 - l_2\|, \tag{8}$$

where $\|AB\|$ and $\left\|\tilde{A}\tilde{B}\right\|$ are the center and top distances between two targets, respectively, and $\left\|AB'\right\|$ is the distance from center A to key line $B\tilde{B}$. Therefore, $\|AB\|+\left\|\tilde{A}\tilde{B}\right\|+\left\|AB'\right\|$ in Eq. (8) represents the positional distance between two targets. Also, $l_1 = \left\|A\tilde{A}\right\|$ and $l_2 = \left\|B\tilde{B}\right\|$ are the lengths of key lines. Thus, $\|l_1 - l_2\|$ provides information on the size difference.

Temporal features are vital for tracking as they focus on global and historical information, complementary to local and spatial features. In order to explore temporal features, we build a line-based motion model on the temporal domain with the constant velocity assumption. As shown by the two dashed lines in Fig. 5, the trajectories of targets are formed on the X-axis, Y-axis, and T-axis. The lengths of key lines are included in building motion dynamics to introduce more information on top of temporal positions. Five frames are used to construct the motion model, which is experimentally demonstrated to be optimal. If the trajectory length of a target is less than five (for instance, three frames), we duplicate the first frame twice in this case. We update the state of targets on this motion model after matching in every frame and make predictions for unmatched targets by referring to their velocity.

With this motion model, we can measure the distance at the trajectory level, which is distinguishable in occlusions and intersections. As shown in Fig. 5, on frame I^t, two targets are represented by key lines $P^t\tilde{P}^t$ and $R^t\tilde{R}^t$, and their trajectories are P^tP^{t-1} and R^tR^{t-1},

respectively. Based on this, we can obtain the distance from the center point R^t to trajectory $P^t P^{t-1}$, i.e., $\|R^t P'\|$, by referring to the distance of the point-to-line method on the X-axis and T-axis. The trajectory distance $\|P^t R'\|$ can be obtained similarly. On top of this, the trajectory distance d_{traj} on frame I^t between these two targets can be obtained by combining the mutual trajectory distances as follows:

$$d_{traj} = \|P^t R'\| + \|R^t P'\|. \tag{9}$$

Line-based motion cues can also be explored by predicting positions and lengths of key lines in future frame I^{t+1} for all targets with their velocity. The unmatched tracks are also predicted until their inactive period reaches the predefined maximum length δ_{max}. As shown in Fig. 5, assuming the predicted key lines of these two targets are $P^{t+1}\tilde{P}^{t+1}$ and $R^{t+1}\tilde{R}^{t+1}$, respectively. The motion distance on frame I^{t+1} between these two targets can be obtained as follows:

$$d_{mo} = d_{spa}\left(P^{t+1}\tilde{P}^{t+1}, R^{t+1}\tilde{R}^{t+1}\right) + \left\|v^t_{P^t\tilde{P}^t} - v^t_{R^t\tilde{R}^t}\right\| \tag{10}$$

where d_{spa} is the spatial distance defined in Eq. (8).

The proposed spatial distance calculates the similarity between targets on the X-axis and Y-axis on frame I^t. The trajectory distance provides the affinity information between different targets on the X-axis and T-axis on frame I^t. Moreover, the motion distance measures the similarity on the X-axis and Y-axis on frame I^{t+1} by inferring on the T-axis, which is a temporal extension of spatial distance on the future frame with velocity difference included. Thus, motion distance can help to distinguish intersected targets and re-identify re-appeared targets, as proved in experiments.

The overall cost can be obtained by combining the proposed d_{spa}, d_{traj}, and d_{mo} between two frames, the cost matrix can be formed, and identity assignment is solved by the greedy algorithm. Note that the unmatched tracks are predicted for two continuous frames each time, in which way they can have the same affinity measurements as matched tracks.

Line-based Scoring. Most methods solve the identity assignments based on local pairwise matching cost, lacking the criterion of identifying falsely assigned identities, degrading the quality of trajectories. As illustrated in Li et al. (2023), the tracker should be able to identify and deactivate false assignments for better identity consistency. Hence, we propose a post-processing procedure that evaluates each identity assignment quality by comparing key lines of matched pairs, which can be assessed as below:

$$s_{as} = \exp\left(-\frac{d_{spa} + d_{traj} + d_{mo}}{h}\right), \tag{11}$$

where d_{spa}, d_{traj}, and d_{mo} are the spatial, trajectory, and motion distances of linked pairs, respectively, and h is the height of the matched track before matching. The obtained s_{as} reflects the quality of matched pairs, and we set a threshold η_2 to determine the matching quality that empowers the tracker to identify and deactivate false assignments, i.e., matchings with scores higher than η_2 are deemed valid, and matchings with scores lower than η_2 are invalid and inactivated.

3.4. Detections and key tracks selection

Detection Selection Strategy. Parameter tuning is vital and sensitive for most methods, which is time-consuming and makes the tracker biased to a specific dataset. The detection threshold that distinguishes the positives and negatives is essential for trackers since existing tracks match with detections on a frame-wise basis for identity assignments. A large detection threshold will filter out more positive responses, decrease false positives (FP), introduce false negatives (FN), and vice versa. Besides, as shown in Fig. 6, some low-scored detections are true positives, which may contain valuable information for tracking. Therefore, a more objective criterion for selecting detections is of great need, for one thing, to reduce the sensitivity to parameters and reliance on detectors, and for another thing, to recover the missed targets.

Fig. 6. Examples of missing targets. The targets highlighted by red arrows are mistakenly suppressed by (a) small-sized target, (b) severe occlusion, and (c) large camera motion. All three cases are processed under threshold 0.4, causing false negatives and track terminations. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

We propose a detection selection strategy to recover missed targets and mitigate the above sensitivity and reliance issues. Assume θ is the detection threshold, and the detections with scores higher than θ, denoted as $\mathcal{D}_h$, are directly used for data association. As for the detections with scores lower than θ, $\mathcal{D}_l = \{b_l^1, b_l^2, \ldots\}$, we propose to recover true positive targets within $\mathcal{D}_l$ by measuring the similarity between their predicted tops and calculated tops. To be more specific, given the ith low-scored detection $b_l^i = \left\{x^i_{l,cent}, y^i_{l,cent}, w^i_l, h^i_l\right\}$, the corresponding predicted top is $t^i_{l,top} = \left\{x^i_{l,top}, y^i_{l,top}\right\}$ if it existed. We first obtain the calculated top $t^i_{l,ct} = \left\{x^i_{l,ct}, y^i_{l,ct}\right\}$ by referring to its center and height as $x^i_{l,ct} = x^i_{l,cent}$, $y^i_{l,ct} = y^i_{l,cent} + \frac{h^i_l}{2}$. Then, we can evaluate the similarity between predicted and calculated tops as:

$$s^{i,t}_{det} = \exp\left(-\frac{\sqrt{\left(x^i_{l,top} - x^i_{l,ct}\right)^2 + \left(y^i_{l,top} - y^i_{l,ct}\right)^2}}{h^i_l}\right). \tag{12}$$

If $s^{i,t}_{det}$ is higher than the threshold η_3, representing the key line of this target is high in quality, and the corresponding detection is considered a high-quality case. By doing so, a set of detections $\mathcal{D}_{lh}$ can be recovered from $\mathcal{D}_l$ and used for data association, allowing the tracker to fully utilize the information provided by detectors and reducing the sensitivity to the detection threshold. Note that η_3 is less sensitive than θ as η_3 is tuned with low-scored detections included, which is less relevant to the training datasets and the performance of detectors.

Key Tracks Selection Strategy. The threshold τ that determines what to encode on the LineNet input heatmap is also essential and sensitive. Existing tracks with high confidence are sometimes false positives, which will misguide the network if used in input heatmap rendering. Likewise, some unmatched tracks still existed in the tracking scene as false negatives. Those unmatched tracks can guide the network to recover targets in future frames by enhancing corresponding features in the input heatmap. Thus, we propose a key track selection strategy, which can filter out mistake targets from matched tracks and select missed targets from unmatched tracks.

The key tracks represent the true positive tracks, including matched key tracks $\mathcal{T}_{k,ma}$ and unmatched key tracks $\mathcal{T}_{k,un}$, which are used for input heatmap encoding. The matched key tracks are selected from matched tracks with false positives removed, and unmatched key tracks are the false negatives that are picked among unmatched tracks. The key track selection relies on key lines and a scoring strategy. More specifically, given a set of matched tracks $\mathcal{T}^{t-1}_{ma} = \{T^{t-1}_{ma,1}, T^{t-1}_{ma,2}, \ldots\}$ on frame I^{t-1}, we evaluate the matching quality(between the ith track and the jth detection) of the ith track by scoring as follows:

$$s^{i,j,t}_{mat} = \exp\left(-\frac{d^{i,j,t}_{spa}}{h^{i,t-1}}\right), \tag{13}$$

where $d_{spa}^{i,j,t}$ denotes the spatial distance between the ith track and the jth detection proposed in Eq. (8), and $h^{i,t-1}$ is the height of the ith track on frame I^{t-1}.

The matched tracks with scores higher than η_2 are regarded as key tracks and are used for input heatmap rendering, and the tracks with lower scores are potential false positives that are wrongly matched. Note that although matching quality has been evaluated by scoring shown in Eq. (11), we still perform selection in Eq. (13) since the former scores in Eq. (11) show the similarity before identity assignments, which may make mistakes in the crowded scene. The scores in Eq. (13) reflect the matched pairs after assignment and work as a post-selection to prevent false positives from being encoded.

The unmatched tracks are predicted with motion models and are selected based on their last matching states before removal. The longer the unmatched period is, the less reliable the prediction is. Thus, for the ith unmatched track, we attenuate its score according to the age of being inactive to obtain its confidence of being a key track as follows:

$$s_{unt}^{i,t} = s_{last}^{i} \cdot \exp(-\frac{a_{unt}^{t}}{\delta_{max}}), \tag{14}$$

where s_{last}^{i} is the last matching score of the ith unmatched track, a_{unt}^{t} accumulates the age of the unmatched period, and δ_{max} is the max-age of the being inactivation before removal. We only consider the unmatched tracks within $\delta_{key} = 8$ frames as verified to be optimal in experiments. The unmatched tracks with scores higher than η_3 are deemed key tracks and used for the input heatmap encoding.

3.5. Line-based high-level features extraction

The proposed key lines proved a superior way for extracting high-level features, which are more discriminative and robust than the box-based and point-based features and can enhance the tracking performance. To prove the effectiveness of line-based high-level feature extraction, we build two line-based trackers, namely CenterLine and FairLine, which are built by predicting line-based displacements to enhance CenterTrack (Zhou et al., 2020) and extracting line-based identity embeddings to enhance FairMOT (Zhang et al., 2021).

As shown in Fig. 7, line-based high-level features are extracted by adding corresponding sub-networks on top of LineNet. Specifically, to produce CenterLine, we add a top displacement prediction network on top of CenterTrack by stacking a 3×3 convolution layer and a 1×1 convolution layer. We learn to predict top displacements using the same regression objective as center displacements in CenterTrack, and the prediction loss is calculated as follows:

$$L_{dis}^{t} = \frac{1}{N} \sum_{i=1}^{N} \left| \hat{D}_{p_i^t} - d_i^t \right|, \tag{15}$$

where $\hat{D}_{p_i^t}$ is the predicted top displacement on frame I^t for the ith target, $d_i^t = p_i^{t-1} - p_i^t$ is the top displacement labels between adjacent frames, p_i^{t-1} and p_i^{t-1} are the ground truths of target positions in two frames, and N is the number of targets.

The top and center displacements are used separately to update corresponding positions and obtain the locations of target key lines on previous frames. The similarity between different targets is computed with the proposed line-based measuring strategy, and identities are assigned with the greedy algorithm.

Similarly, as shown in Fig. 7, to build FairLine, we add a 1×1 convolution layer with 128 kernels on top of the backbone features of FairMOT to extract top-based appearance embeddings, and fuse them with center-based appearance embeddings by channel concatenation. The training labels are identity-dependent, i.e., we treat each identity as a class in training the feature embedding network. Therefore, we transform the fused center-based and top-based features by applying a 1×1 convolution layer to reduce the feature dimension to obtain the final line-based identity embeddings, which align with corresponding one-hot annotations.

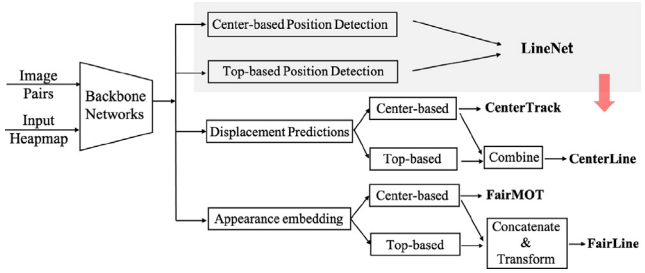

Fig. 7. The architecture sketch of CenterLine and FairLine, which are built on top of LineNet by adding a top-based displacement prediction branch and a top-based appearance embedding branch, respectively.

Finally, a fully connected layer and a Softmax function are employed to map the line-based identity embeddings to a class distribution vector $p(k)$. The training loss of the line-based appearance embedding network is shown as follows:

$$L_{ID} = -\sum_{i=1}^{N} \sum_{k=1}^{K} L^{i}(k) log\,(p(k)), \tag{16}$$

where N is the number of targets, K is the number of different identities in the training data, and $L^{i}(k)$ denotes the one-hot class label of different identities. The proposed FairLine is trained with uncertainty loss (Kendall et al., 2018) to balance detection and feature embeddings.

3.6. Cascade tracking algorithm

To associate targets with base and high-level features of key lines for CenterLine and FairLine, we propose the Cascade Tracking algorithm, which requires three steps to accomplish data association by dealing with targets of different types.

Given the detections $\mathcal{D}_h^t$, existing tracks $\mathcal{T}_{ma}^{t-1}$, and unmatched tracks $\mathcal{T}_{un}^{t-1}$, we first divide the unmatched tracks $\mathcal{T}_{un}^{t-1}$ into two subsets by the key track selection strategy: the unmatched key tracks $\mathcal{T}_{un,key}^{t-1}$ and the unmatched non-key tracks $\mathcal{T}_{un,nok}^{t-1}$. Then, we select a set of high-quality subset $\mathcal{D}_{lh}^t$ from low-scored detections $\mathcal{D}_l^t$ by the detection selection strategy. We combine existing tracks $\mathcal{T}_{ma}^{t-1}$ and unmatched key tracks $\mathcal{T}_{un,key}^{t-1}$ to form a track pool $\mathcal{T}_1^t$.

For the first matching stage, the high-scored detections $\mathcal{D}_h^t$ are matched with tracks in $\mathcal{T}_1^t$ by measuring with line-based high-level features. In detail, for CenterLine, we use the predicted line-based displacements to update the positions of detections and associate them with tracks by line-based measuring. For FairLine, the similarity is obtained by calculating the cosine distance of line-based appearance features. Note that only unmatched key tracks are associated in this stage instead of all unmatched ones. The unmatched detections of $\mathcal{D}_h^t$ in this stage are denoted as $\mathcal{D}_{uh1}^t$, and the unmatched tracks from $\mathcal{T}_1^t$ are denoted as $\mathcal{T}_{u1}^t$.

In the second stage, we try to recover the low-scored true positives from $\mathcal{D}_{lh}^t$ by matching the detections in $\mathcal{D}_{lh}^t$ with tracks in $\mathcal{T}_{u1}^t$. The similarity is calculated by the proposed line-based measuring strategy with base features of key lines, and the matching is solved by the greedy algorithm. After matching, the proposed line-based scoring is followed as a track deactivation post-processing to enhance identity consistency. The remaining unmatched detections in $\mathcal{D}_{lh}^t$ are no longer considered to reduce false positives.

In the third stage, we intend to solve the long-term tracking problem. Most tracks in $\mathcal{T}_{un,nok}^{t-1}$ are unmatched for several frames, and we combine tracks $\mathcal{T}_{un,nok}^{t-1}$ with tracks $\mathcal{T}_{u2}^t$, which are left from the second stage, to form another track pool $\mathcal{T}_2^t$. Then, we match $\mathcal{T}_2^t$ with unmatched detections in $\mathcal{D}_{uh1}^t$ with the line-based measuring strategy. With similarity calculated by base features of key lines, the cost of matching candidates is obtained with local and global information

Algorithm 1 Line-based Cascade Tracking.

Input:

- The high-scored detections $\mathcal{D}_h^t$ on frame I^t
- The low-scored detections $\mathcal{D}_l^t$ on frame I^t
- The existing tracks $\mathcal{T}_{ma}^{t-1}$ on frame I^{t-1}
- The unmatched tracks $\mathcal{T}_{un}^{t-1}$ on frame I^{t-1}

Output:

- The tracks $\mathcal{T}_{ma}^t$ on frame I^t

Main Algorithm:

/ * *First Stage* * /
Divide the unmatched tracks $\mathcal{T}_{un}^{t-1}$ into unmatched key tracks $\mathcal{T}_{un,key}^{t-1}$ and unmatched non-key tracks $\mathcal{T}_{un,nok}^{t-1}$ with the key track selection strategy;
Combine the existing tracks $\mathcal{T}_{ma}^{t-1}$ with $\mathcal{T}_{un,key}^{t-1}$ to form the track pool $\mathcal{T}_1^t$;
Compute the cost matrix of matching candidates between $\mathcal{D}_h^t$ and $\mathcal{T}_1^t$ based on the line-based high-level features (line-based displacement predictions for CenterLine and line-based appearance embeddings for FairLine);
Associate targets between $\mathcal{D}_h^t$ and $\mathcal{T}_1^t$;
Output matched detections $\mathcal{D}_{mh1}^t$, matched tracks $\mathcal{T}_{m1}^t$, unmatched detections $\mathcal{D}_{uh1}^t$, and unmatched tracks $\mathcal{T}_{u1}^t$;
/ * *Second Stage* * /
Select detections $\mathcal{D}_{lh}^t$ from $\mathcal{D}_l^t$ with the detection selection strategy;
Compute the cost matrix of matching candidates between $\mathcal{D}_{lh}^t$ and $\mathcal{T}_{u1}^t$ based on the line-based measuring strategy;
Associate targets between $\mathcal{D}_{lh}^t$ and $\mathcal{T}_{u1}^t$;
Output matched detections $\mathcal{D}_{mlh}^t$, matched tracks $\mathcal{T}_{m2}^t$, and unmatched tracks $\mathcal{T}_{u2}^t$;
/ * *Third Stage* * /
Combine the unmatched non-key tracks $\mathcal{T}_{un,nok}^{t-1}$ and unmatched tracks $\mathcal{T}_{u2}^t$ to form the track pool $\mathcal{T}_2^t$;
Compute the cost matrix of matching candidates between $\mathcal{D}_{uh1}^t$ and $\mathcal{T}_2^t$ based on the line-based measuring strategy;
Associate targets between $\mathcal{D}_{uh1}^t$ and $\mathcal{T}_2^t$;
Output matched detections $\mathcal{D}_{mh2}^t$, matched tracks $\mathcal{T}_{m3}^t$, and unmatched detections $\mathcal{D}_{uh2}^t$;
/ * *Track Initialization* * /
Initialize unmatched detections $\mathcal{D}_{uh2}^t$ as new tracks.

included, enabling the targets to be re-identified accurately. The unmatched tracks that are matched in this stage can re-activate their identity and reduce ID switches. A few matching candidates are formed in the third stage, with predictions of unmatched tracks being the majority. Thus, a smaller threshold η_4 is used in the scoring procedure as the threshold. Finally, the remaining unmatched detections in $\mathcal{D}_{uh1}^t$ are initialized as new tracks.

In the proposed line-based Cascade Tracking Algorithm, most easy targets with high detection confidence are tracked in the first stage. Besides, the unmatched tracks $\mathcal{T}_{u1}^t$ used in the second stage are mostly 'hard' cases, which are hard to track owing to occlusions and distractors. The low-scored detections $\mathcal{D}_{lh}^t$, generally ignored in the previous methods, are activated when matched. Thus, tracking helps with detection in the second stage by recovering low-scored ones. Moreover, the unmatched tracks are kept with predictions, and long-range trajectories with consistent ID can be formed with track re-identification in the third stage, enhancing the long-term tracking ability of trackers. Alg. 1 summarizes the proposed line-based Cascade Tracking algorithm.

4. Experiments

The tracking performance of our method is evaluated on the MOTChallenge benchmarks, including MOT16, MOT17 (Milan et al., 2016), and MOT20 (Dendorfer et al., 2020). We first introduce the datasets and parameters of our methods. Since there is no validation dataset in the MOTChallenge benchmarks, we split the MOT17 training dataset into two halves. We use the first half for training and the second half for validation to conduct ablation studies and verify the effectiveness of each component. Finally, we compare our trackers with state-of-the-art methods and discuss the limitations of our methods.

4.1. Datasets and metrics

The commonly used MOTChallenge benchmark datasets are MOT16, MOT17, and MOT20, which include tracking scenes with various conditions and online test servers. The MOT17 is the most popular dataset with 7 training sequences and 7 sequences for testing. MOT16 has the same tracking sequences as MOT17 but differs in the provided detections and annotations. The detections of MOT17 are provided by DPM (Felzenszwalb et al., 2009), Faster R-CNN, and SDP (Yang et al., 2016), and only DPM is provided for MOT16. The MOT20 is a newly released dataset recorded from extremely crowded scenes, containing 4 train and 4 test sequences with detections provided by Faster R-CNN.

The tracking performance is usually evaluated from different aspects with several metrics. The two most important metrics are MOTA (Multiple Object Tracking Accuracy) (Bernardin and Stiefelhagen, 2008) and IDF1 (ID F1 score) (Ristani et al., 2016). MOTA reveals the tracking convergence, and IDF1 describes the identity consistency. Other metrics are also employed for evaluation, such as Most Tracked (MT), Most Lost (ML), False Positives (FP), False Negatives (FN), ID Switches (IDS), and FPS (Frames Per Second).

4.2. Implementation details

We build our LineNet upon CenterNet (Zhou et al., 2019), a variant of DLA34 (Yu et al., 2018) adopted in FairMOT (Zhang et al., 2021) is used as the backbone network in our method, where more skip connections between low-level and high-level features are added for feature fusion, and up-sampling is performed by deformable convolution (Dai et al., 2017). The training processes and parameters of CenterLine and FairLine mainly follow CenterTrack and FairMOT, respectively. Our network is pre-trained on CrowdHuman (Shao et al., 2018) and fine-tuned on the MOT17/MOT20 with Adam (Kingma and Ba, 2014). We train CenterLine for 80 epochs with a starting learning rate of $3.125e-4$ and drop by a factor of 10 at 60 epochs, and the batch size is 8. We train FairLine for 35 epochs with a starting learning rate of $10e-4$ and drop by a factor of 10 at 25 epochs, and the batch size is 10.

In our method, the parameters θ and τ are 0.4 and 0.5, respectively, following the baseline trackers CenterTrack and FairMOT to reduce the influence of parameters, and δ_{max} is set to 30 for track rebirth following baseline trackers. η_1 is 0.9 for constructing key lines, η_2 is 0.2 for identifying false assignments and selecting key tracks, η_3 is 0.9 to filter out detections of low quality, and η_4 is 0.7 for matching evaluation. δ_{key} is 8 for key tracks encoding on the input heatmap. These parameters are selected empirically by experiments and proved optimal.

4.3. Effectiveness of key lines

The effectiveness of key lines. We propose the key line as a generic and distinguishable base model with superior representative ability. In order to prove the effectiveness of the key lines in providing discriminative base features, we perform ablations regarding different base models by tracking with base features and testing on the MOT17 training set. Tracking results are shown in Table 1. CBox denotes the tracker that represents targets with bounding boxes and tracks by measuring the overlaps of target bounding boxes. CPoint represents the tracker that uses center points as the base model and tracks by measuring the center-based spatial distance. CLine is the tracker that models targets with the proposed key lines and calculates similarity

Table 1
Comparisons of tracking performance obtained by different base models on the MOT17 training dataset. BR represents the base representation of each method, and TR denotes the track rebirth strategy.

Method	BR	TR	MOTA↑	IDF1↑	FP↓	FN↓	IDS↓
CBox	Box		62.2	49.1	2434	15278	2658
CPoint	Point		65.8	63.8	2450	15294	692
CLine	Line		68.4	65.6	1801	14569	664
CBox	Box	✓	62.2	47.5	2163	15919	2299
CPoint	Point	✓	65.7	69.0	2112	15967	393
CLine	Line	✓	68.2	72.6	1585	15220	358

Table 2
The performance of the state-of-the-art methods with different base representations on the MOT17 training dataset. The Features column gives the different features used for data association, including combinations of A (Appearance), M (Motion), B (Box), P (Point), and L (Line).

Method	Features	MOTA↑	IDF1↑	FP↓	FN↓	IDS↓
CenterTrack	M+P	66.2	69.2	2113	15888	219
CenterTrack†	M+L	68.0	71.1(+1.9)	1298	15665	270
JDE	A+B	63.5	64.1	6178	33428	1363
JDE†	A+L	63.5	64.4(+0.3)	6219	33410	1323
TraDeS	M+P	68.2	71.7	1913	14962	291
TraDeS†	M+L	68.2	72.0(+0.3)	1912	14962	282
FairMOT	A+B	69.1	72.8	1976	14443	299
FairMOT†	A+L	69.3	73.4(+0.6)	2307	14009	296
PermaTrack	M+P	69.5	71.9	2655	13528	255
PermaTrack†	M+L	69.8	75.5(+3.6)	2606	13487	199
GSDT	A+B	72.7	72.2	9948	19965	798
GSDT†	A+L	72.7	72.5(+0.3)	9930	19974	805

using the proposed line-based measuring strategy. All methods use the detections provided by CenterNet for a fair comparison, and TR denotes the track rebirth.

Table 1 shows that the proposed line-based method outperforms the box-based and point-based methods. The vulnerability of bounding box overlaps in crowded scenes often leads to false assignments, thus resulting in the lowest IDF1 and the largest ID switches in CBox. Tracking results of CBox worsen when using track rebirth, showing that box-based re-identifications are inaccurate and unreliable. Besides, CPoint precedes CBox, demonstrating the discrimination of point-based modeling in crowded scenes. However, it still lags behind CLine since line-based base features are more distinguishable, which can increase the accuracy of point-based affinity measurement and boost MOTA (from 65.8 to 68.4) and IDF1 (from 63.8 to 65.6) without track rebirth. A more significant increment of 3.6 (from 69.0 to 72.6) in IDF1 is observed with track rebirth, demonstrating the effectiveness of line-based target re-identification. In addition, a significant increment of 7.0 (from 65.6 to 72.6) in IDF1 is achieved with track rebirth for CLine, proving the superiority of key lines in forming long-term tracks. Thus, the results of Table 1 prove that the proposed key line is a generic and informative base representation, which can provide discriminative base features and is more reliable in long-term predictions and target re-identification.

The generalizations of key lines. We apply key lines to six recently published state-of-the-art methods, including CenterTrack (Zhou et al., 2020), JDE (Wang et al., 2020b), TraDeS (Wu et al., 2021), FairMOT (Zhang et al., 2021), PermaTrack (Tokmakov et al., 2021), and GSDT (Wang et al., 2020a) to demonstrate the generalization and effectiveness of the proposed line-based representations. The results are obtained by testing on the MOT17 training set and are shown in Table 2. In the table, key lines are built with calculated tops by referring to target center positions and heights to replace original base models (bounding box or center point) in each method. Therefore, key line replacements in Table 2 do not need to re-train networks, and extra computations are negligible.

Table 3
The performance comparison of base model constructing strategies and network settings of LineNet.

Setting	MOTA↑	IDF1↑	FP↓	FN↓	IDS↓
key line + bottom	67.9	71.4	1600	15322	395
key line + left + right	68.0	71.9	1648	15244	356
w/o top prediction	68.1	72.4	1629	15207	366
w/o input heatmap	64.6	68.8	998	17644	410
key line (Ours)	68.2	72.6	1585	15220	358

In Table 2, the methods with † indicate the variant tracker with key lines as base models. For example, CenterTrack utilizes motion predictions for data association with point-based representation (motion + point, M+P). We apply key lines to CenterTrack to construct CenterTrack†, which adopts the key lines to represent targets and measure similarity with line-based measurement (motion + line, M+L). Likewise, data association in FairMOT is performed with center-based appearance features, and the box overlaps are used for matching after appearance embeddings (appearance + box, A+B). We apply key lines to FairMOT and replace bounding boxes for matching at the second stage of data association (appearance + line, A+L), and this variant is denoted as FairMOT†.

Evident improvements in IDF1 are achieved with key line replacement on all trackers in Table 2, especially for methods with motion predictions, i.e., CenterTrack and PermaTrack. The reason is that key lines can provide more discriminative features to compensate for the weakness of point-based predictions and measurements. The enhancement is less significant in trackers using appearance features since they are already discriminative and can handle most targets. However, the increments show that key lines can work parallel with appearance embeddings to improve the similarity measurement and enhance identity consistency. Thus, Table 2 demonstrates the generalization of the proposed key lines, which can be easily applied to existing trackers as base models, compensate for coarse similarity provided by bounding boxes and center points to enhance identity consistency, and are compatible with high-level features.

4.4. Line-based modeling and measuring

Ablation experiment on base model construction. Adding more representative points to form a complicated base model is intuitive. As in MTrack (Yu et al., 2022a) and AdaMOT (Liang et al., 2022a), a set of points is used to obtain a finer semantic representation of targets. Alternative options include the bottom, left, and right points. Table 3 shows the comparisons of different options in building the key lines.

Specifically, we build the network to predict the target bottom points upon LineNet and measure affinities with added bottom points. The results show that adding bottom points degrades MOTA and IDF1, increasing FP, FN, and IDS. Similar phenomena are observed when adding left and right points. The reason is that the bottom points of targets in MOT17 are usually feet and ground, which change significantly in appearance and position while walking, thus deteriorating the temporal consistency of affinity between identical targets. The left and right points change more frequently while walking, making the horizontal lines less reliable in similarity measurement. Thus, those representative points are inconsistent, and confusion is inevitable if used in data association. On the contrary, the proposed key lines are more prominent and consistent in crowded scenes, which can provide discriminative features with low computational complexity.

Ablation experiment on LineNet structure. We also investigate the role of the top prediction branch and the input heatmap in LineNet. As shown in Table 3, the tracking performance degrades slightly in MOTA and IDF1 without the top prediction branch, in which key lines are constructed with calculated tops, demonstrating the importance of predicted tops in building key lines. Besides, the effectiveness of the

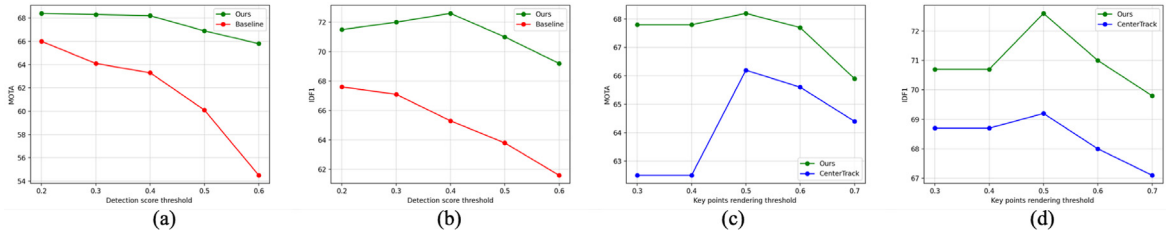

Fig. 8. The sensitivity analysis of parameters. (a) and (b) show the comparisons between the baseline and our method in MOTA and IDF1 regarding detection parameters θ. (c) and (d) show the comparisons of the encoding threshold τ between CenterTrack and our CenterLine in MOTA and IDF1.

Table 4
The ablations regarding the influence of different components proposed in the line-based measuring strategy.

Spatial	Trajectory	Motion	MOTA↑	IDF1↑	FP↓	FN↓	IDS↓
	✓	✓	68.1	68.1	2243	14580	353
✓		✓	68.1	72.4	1585	15218	376
✓	✓		68.0	70.6	1628	15172	427
✓	✓	✓	68.2	72.6	1585	15220	358

input heatmap is shown by the dramatic decrease in MOTA and IDF1 and increased FN of the method w/o input heatmap in Table 3, proving that the input heatmap can enhance the features of true positives and guide the network to recover missed targets.

Ablation experiment on line-based measuring. When calculating affinity with base features, the proposed line-based measuring considers spatial, trajectory, and motion distances. We investigate the influence of each component, and the results are shown in Table 4. It is evident that spatial distance is the most discriminative component since it focuses on differences regarding position and size to distinguish targets, which are vital in crowded scenes, as shown by a significant drop of 4.5 (from 72.6 to 68.1) in IDF1 without spatial distance. Besides, the trajectory distance is measured at the track level by considering the temporal information. Thus, it can enhance identity consistency, as witnessed by the reduced IDS (from 376 to 358) and improved IDF1 (from 72.4 to 72.6) when using trajectory distance. Moreover, the reduced MOTA and IDF1 and increased IDS show that motion distance can help to recover and distinguish targets. On the one hand, unmatched tracks are predicted and recovered if matched. On the other hand, the velocity difference and temporal motion predictions help to distinguish targets in interacting scenarios.

4.5. Parameters sensitivity analysis

The sensitivity analysis on the detection threshold. As discussed in Section 3.4, the detection threshold θ is a vital yet sensitive parameter for most trackers, and we employ key lines to select low-scored detections to reduce the sensitivity of related parameters. Fig. 8(a) depicts the comparison of the MOTA between the baseline and our method. The baseline tracker only utilizes high-scored detections for data association. On the contrary, our method also matches detections with scores lower than θ by selection on top of the baseline. As shown in Fig. 8(a), our approach is more robust than the baseline in a border range of θ, and superior performance is achieved without costly tuning. A similar trend in IDF1 can be found in Fig. 8(b).

Besides, using all low-scored detections without selection will dramatically increase FP (from 1585 to 3024). Although MOTA increases slightly by 0.2 (from 68.2 to 68.4), IDF1 drops by 1.9 (from 72.6 to 70.7), which verifies that using all low-scored targets without selections will severely deteriorate identity consistency. Therefore, the proposed detection selection strategy can better balance MOTA and IDF1 in tracking.

The sensitivity analysis on the encoding threshold. As shown in Figs. 8(c) and (d), the MOTA and IDF1 between our CenterLine and CenterTrack are compared regarding the encoding parameter τ. Our tracker selects key tracks for rendering in the input heatmap, while all existing tracks are rendered in CenterTrack. Obviously, our method is more robust to the encoding threshold τ and performs better with a broader range. The key tracks are true positive targets in tracking scenes despite the matching state, incorporating helpful information that can boost localization accuracy. Therefore, selecting key tracks for rendering can reduce the sensitivity of the encoding threshold τ and improve tracking performance.

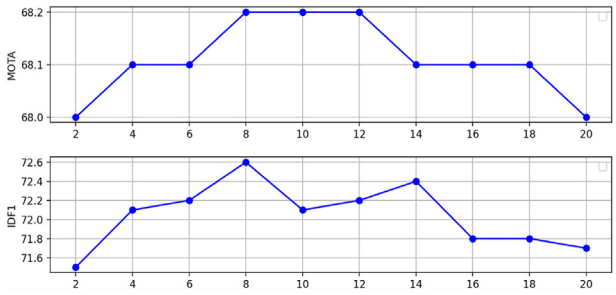

Fig. 9. The performance of MOTA and IDF1 regarding the numbers of frames δ_{key} used for rendering in the input heatmap.

Table 5
Comparisons with the state-of-the-art methods on the test sequence MOT17-06. The best results are highlighted in bold.

Method	BR	MOTA↑	IDF1↑	MT↑	ML↓	IDS↓
CenterTrack	Point	**62.2**	47.2	77	41	169
PermaTrack	Point	61.1	48.0	82	44	220
TraDeS	Point	59.9	57.3	73	50	211
CenterLine	Line	60.5	**62.2**	**84**	**39**	**161**

The analysis of the encoding frames. Our method uses key tracks from δ_{key} frames for input heatmap rendering. As in Fig. 9, MOTA is relatively steady with different numbers of frames. However, IDF1 changes dramatically and peaks when using 8 frames. With more frames, such as 10 and 12, MOTA remains at 68.2, but the number of FP increases significantly introduced by incorrect rendering. The more frames used, the more false positives are produced, resulting in increased false assignments. Hence, we chose $\delta_{key} = 8$ frames as it can balance FP and FN, and yield the best performance.

4.6. Robustness analysis

The line-based Cascade Tracking algorithm is proposed to fully utilize the information provided by key lines, which performs data association in a cascade fashion to cope with challenging scenarios in MOT. We compare the tracking performance of CenterLine with three state-of-the-art methods to prove the effectiveness and discrimination of line-based features in camera motions and low frame rates scenarios.

As shown in Table 5, we compare CenterLine with CenterTrack (Zhou et al., 2020), PermaTrack (Tokmakov et al., 2021), and TraDeS (Wu et al., 2021) in the test sequence MOT17-06, which is captured under

Fig. 10. Qualitative results of different methods in long-term occlusions, camera motions, and low frame rate scenarios. The results are obtained on testing sequence MOT17-06. Taking the woman with a white shirt as an example, CenterTrack, TraDeS, and FairMOT make continuous ID switches shown by red arrows. Our CenterLine can preserve the ID and recover targets under occlusions. Different colors represent different IDs. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Fig. 11. Typical failure cases with ID switches. (a) shows typical failure cases from scale variance caused by occlusions and camera motion. (b) shows typical failure cases from fast camera motion and target small size. The failure cases are highlighted with red arrows. Different colors represent different IDs.

low frame rates (14 FPS) and camera motions scenarios. CentetrLine utilizes line-based motion displacements and the line-based measuring strategy for data association. CenterTrack, PermaTrack, and TraDeS share the same base detector with CenterLine and associate targets with center-based motion predictions. We summarize the performance comparison in sequence MOT17-06 in Table 5. Our method achieves the best performance in IDF1, MT, ML, and IDS, demonstrating the superiority of key lines in coping with camera motions and low frame rates scenarios, and longer trajectories are formed with consistent IDs shown by the superiority in MT, IDS, and IDF1.

Fig. 10 shows qualitative comparisons in the test sequence MOT17-06 between CenterLine and three state-of-the-art trackers, including CenterTrack, TraDeS, and FairMOT. FairMOT and TraDeS extract appearance features for data association. We can see from the figure that CenterTrack, TraDeS, and FairMOT make continuous ID switches in challenging scenarios, as shown by red arrows. On the contrary, CenterLine can tackle frequent occlusions and camera motions, reducing ID switches and preserving the identities of targets to improve the temporal consistency of trajectories. Therefore, key lines can provide an accessible and reliable solution for long-term tracking under continuous occlusion.

4.7. Benchmark comparison

We compare our method with state-of-the-art approaches in three multi-object tracking benchmarks, including MOT16, MOT17, and MOT20. The results of peer-reviewed trackers on the MOTChallenge benchmark are listed in Table 6. As illustrated in Section 3.5, we build two line-based trackers for comparison, including CenterLine and FairLine, in which targets are associated with the proposed Cascade Tracking algorithm and tested on private detection protocol.

Table 6 shows that our CenterLine outperforms the baseline tracker CenterTrack (Zhou et al., 2020) in MOT17, showing the superiority of line-based motion predictions. Key lines enable CenterLine to access the suppressed low-scored detections, and obtain accurate motion predictions and similarity measurements, thus achieving superior performance. Likewise, FairLine improves baseline FairMOT in MOTA and IDF1 in all datasets, demonstrating that line-based appearance embeddings are more distinguishable than point-based appearance features. Moreover, compared with CenterTrack and FairMOT, the inference speeds (FPS in Table 6) of the proposed CenterLine and FairLine only drop slightly, showing the potential of key lines in real-time applications. Thus, it can be observed that high-level features extracted based on key lines are more discriminative and can effectively improve the tracking performance.

Two similar MOT methods that exploit different representations for appearance feature extraction are included in Table 6, namely MTrack (Yu et al., 2022a) and AdaMOT (Liang et al., 2022a). MTrack uses multiple adaptively selected key points to enrich the representative features of targets. AdaMOT adopts learnable points for robust feature representations. Our FairLine extracts line-based appearance embeddings and performs better in MOT16 and MOT17, verifying the superiority of the proposed line-based appearance features in target association. Although FairLine lags behind MTrack and AdaMOT in MOTA in the MOT20 dataset, FairLine achieves superior IDF1 and outperforms MTrack by a large margin, showing the superiority of line-based embeddings in providing distinguishable features in extremely crowded scenes.

Compared with the SOTA method CorrTracker (Wang et al., 2021), which introduces spatial and temporal correlation modules to improve the local features of targets, our FairLine achieves the second-best MOTA in MOT16 and MOT17 datasets and exceeds CorrTracker in IDF1 in MOT20 dataset, showing the effectiveness of line-based appearance embeddings in deal with small-sized target associations. Furthermore, FairLine outperforms some of the recently proposed Transformer-based methods, including TrackFormer (Meinhardt et al., 2022), MeMOT (Cai et al., 2022), TransTrack (Sun et al., 2020), and GTR (Zhou et al., 2022), showing the superiority of key lines in MOT.

4.8. Discussion

Despite the superior performance of our methods, there is still plenty of room to improve. Fig. 11 shows some typical failure cases of our trackers, highlighted by red arrows. The figures in Fig. 11(a) are selected from the test sequence MOT17-12 produced by FairLine, and the figures in Fig. 11(b) are from MOT17-14 produced by CenterLine.

The line-based appearance features of FairLine are inconsistent in the tracking scenes shown in Fig. 11(a) owing to scale variance caused by occlusions and camera motion. Likewise, the accuracy of motion predictions of CenterLine is decreased in the tracking scenes shown in Fig. 11(b) owing to fast camera motion and the small size of targets.

Table 6
Comparisons with the state-of-the-art methods on MOT benchmark datasets. The third column BR indicates the base representation. The best result of each metric is highlighted in bold, and the second best is underlined.

Dataset	Method	BR	MOTA↑	IDF1↑	MT↑	ML↓	FP↓	FN↓	IDS↓	FPS↑
MOT16	CTracker (Peng et al., 2020)	Box	67.6	57.2	32.9	23.1	8934	48305	1897	6.8
	QDTrack (Pang et al., 2021)	Box	69.8	67.1	41.6	19.8	9861	44050	1897	20.3
	TraDeS (Wu et al., 2021)	Point	70.1	64.7	37.3	20.0	**8091**	45210	1144	22.3
	MTrack (Yu et al., 2022a)	Point	74.3	72.9	**50.6**	15.7	19236	29554	642	–
	FairMOT (Zhang et al., 2021)	Point	74.9	72.8	44.7	15.9	10163	34484	1074	25.9
	CSTrack (Liang et al., 2022b)	Point	75.6	73.3	42.8	16.5	9646	33777	1121	15.8
	RelationTrack (Yu et al., 2022b)	Point	75.6	75.8	43.1	21.5	9786	34214	**448**	8.5
	AdaMOT (Liang et al., 2022a)	Point	76.2	**76.1**	49.8	16.3	15170	**27769**	688	**26.0**
	CorrTracker (Wang et al., 2021)	Point	**76.6**	74.3	47.8	13.3	10860	30756	979	15.6
	CenterLine	Line	70.2	67.2	39.3	16.2	11219	42018	1050	16.1
	FairLine	Line	76.2	74.0	46.8	**12.8**	12110	29930	2075	25.1
MOT17	CTracker (Peng et al., 2020)	Box	66.6	57.4	32.2	24.2	22284	160491	5529	6.8
	CenterTrack (Zhou et al., 2020)	Point	67.8	64.7	34.6	24.6	**18498**	160332	3039	17.0
	QDTrack (Pang et al., 2021)	Box	68.7	66.3	40.6	21.9	26589	146643	3378	20.3
	SOTMOT (Zheng et al., 2021)	Point	71.0	71.9	42.7	15.3	39537	118983	5184	16.0
	MeMOT (Cai et al., 2022)	Box	72.5	69.0	43.8	18.0	37221	115248	2724	–
	GSDT (Wang et al., 2020a)	Point	73.2	66.5	41.7	17.5	26397	120666	3891	4.9
	MTrack (Yu et al., 2022a)	Point	73.5	72.1	**49.0**	16.8	53361	101844	2028	–
	FairMOT (Zhang et al., 2021)	Point	73.7	72.3	43.2	17.3	27507	117477	3303	25.9
	PermaTrack (Tokmakov et al., 2021)	Point	73.8	68.9	43.8	17.2	28998	115104	3699	11.9
	RelationTrack (Yu et al., 2022b)	Point	73.8	74.7	41.7	23.2	27999	118623	**1347**	8.5
	TrackFormer (Meinhardt et al., 2022)	Box	74.1	68.0	47.3	**10.4**	34602	108777	2829	5.7
	CSTrack (Liang et al., 2022b)	Point	74.9	72.3	41.5	17.5	23847	114303	3567	15.8
	GTR (Zhou et al., 2022)	Box	75.3	71.5	–	–	26793	109854	2859	19.6
	AdaMOT (Liang et al., 2022a)	Point	75.7	**75.5**	48.5	17.2	39777	**95385**	2226	**26.0**
	CorrTracker (Wang et al., 2021)	Point	**76.5**	73.6	47.6	12.7	29808	99510	3369	15.6
	CenterLine	Line	69.7	66.7	38.3	18.0	29007	138945	3207	16.1
	FairLine	Line	76.1	73.3	45.6	13.4	29508	101385	4203	25.1
MOT20	UMA (Yin et al., 2020)	Box	53.1	54.4	21.5	31.8	22893	239534	2251	–
	FairMOT (Zhang et al., 2021)	Point	61.8	67.3	68.8	7.6	103440	88901	5243	13.2
	MeMOT (Cai et al., 2022)	Box	63.7	66.1	57.5	14.3	47882	137983	1938	–
	TransTrack (Sun et al., 2020)	Box	64.5	59.2	49.1	13.6	28566	151377	3565	**14.9**
	CorrTracker (Wang et al., 2021)	Point	65.2	69.1	66.4	8.9	79429	95855	5183	8.5
	CSTrack (Liang et al., 2022b)	Point	66.6	68.6	50.4	15.5	25404	144358	3196	4.5
	TrackFormer (Meinhardt et al., 2022)	Box	68.6	65.7	53.6	14.6	**20348**	140373	**1532**	5.7
	SOTMOT (Zheng et al., 2021)	Point	68.6	71.4	64.9	9.7	57064	101154	4209	8.5
	AdaMOT (Liang et al., 2022a)	Point	69.1	**71.4**	61.4	10.6	58471	99833	1792	12.1
	MTrack (Yu et al., 2022a)	Point	**69.2**	63.5	**68.8**	**7.5**	96123	**86964**	6031	–
	CenterLine	Line	61.5	59.2	46.1	18.2	26976	170388	2072	8.9
	FairLine	Line	65.2	70.2	60.7	10.6	61172	114836	4278	12.2

As a result, our trackers make continuous ID switches, shown by red arrows. These errors could be remedied with the help of a dedicated memory module that dynamically stores and updates the features of targets (Lu et al., 2023; Cai et al., 2022). Moreover, the proposed key line is applied to pedestrians in this paper, and our future research will concentrate on building a generic base model for vehicle tracking scenarios.

5. Conclusion

In this work, we proposed the key line as the generic base representation for MOT, which can provide discriminative features for data association, outperforming commonly used box-based and point-based models. Besides, we proposed the line-based measuring strategy, which aligns with line-based representation and provides accurate affinity of targets. Then, we proposed to utilize low-scored detections and unmatched tracks by selection based on key lines to recover missed targets. Finally, two line-based trackers are proposed by applying the line-based modeling strategy to existing trackers, and we presented a novel line-based Cascade Tracking algorithm that associates targets in three stages. Competitive results in MOT benchmarks and extensive experiments demonstrate that key lines can be easily applied to existing methods to improve identity consistency, and very competitive results can be obtained by applying line-based modeling to existing trackers. We thus believe that the proposed key line is a generic and effective base representation for multi-object tracking.

Declaration of competing interest

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

Data availability

No data was used for the research described in the article.

References

Bergmann, P., Meinhardt, T., Leal-Taixe, L., 2019. Tracking without bells and whistles. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. ICCV, pp. 941–951.

Bernardin, K., Stiefelhagen, R., 2008. Evaluating multiple object tracking performance: the CLEAR MOT metrics. EURASIP J. Image Video Process. 2008, 1–10.

Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B., 2016. Simple online and realtime tracking. In: 2016 IEEE International Conference on Image Processing. ICIP, IEEE, pp. 3464–3468.

Bochinski, E., Eiselein, V., Sikora, T., 2017. High-speed tracking-by-detection without using image information. In: 2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance. AVSS, IEEE, pp. 1–6.

Cai, J., Xu, M., Li, W., Xiong, Y., Xia, W., Tu, Z., Soatto, S., 2022. Memot: Multi-object tracking with memory. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8090–8100.

Dai, J., Qi, H., Xiong, Y., Li, Y., Zhang, G., Hu, H., Wei, Y., 2017. Deformable convolutional networks. In: Proceedings of the IEEE International Conference on Computer Vision. ICCV, pp. 764–773.

Dendorfer, P., Rezatofighi, H., Milan, A., Shi, J., Cremers, D., Reid, I., Roth, S., Schindler, K., Leal-Taixé, L., 2020. Mot20: A benchmark for multi object tracking in crowded scenes. arXiv preprint arXiv:2003.09003.

Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D., 2009. Object detection with discriminatively trained part-based models. IEEE Trans. Pattern Anal. Mach. Intell. 32 (9), 1627–1645.

Guo, S., Wang, J., Wang, X., Tao, D., 2021. Online multiple object tracking with cross-task synergy. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 8136–8145.

Kang, K., Li, H., Xiao, T., Ouyang, W., Yan, J., Liu, X., Wang, X., 2017. Object detection in videos with tubelet proposal networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 727–735.

Kendall, A., Gal, Y., Cipolla, R., 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7482–7491.

Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

Law, H., Deng, J., 2018. Cornernet: Detecting objects as paired keypoints. In: Proceedings of the European Conference on Computer Vision. ECCV, pp. 734–750.

Li, Y.-F., Ji, H.-B., Chen, X., Lai, Y.-K., Yang, Y.-L., 2023. Multi-object tracking with robust object regression and association. Comput. Vis. Image Underst. 227, 103586.

Liang, T., Li, B., Wang, M., Tan, H., Luo, Z., 2022a. A closer look at the joint training of object detection and re-identification in multi-object tracking. IEEE Trans. Image Process. 32, 267–280.

Liang, C., Zhang, Z., Zhou, X., Li, B., Zhu, S., Hu, W., 2022b. Rethinking the competition between detection and ReID in multiobject tracking. IEEE Trans. Image Process. 31, 3182–3196.

Lin, T.-Y., Dollar, P., Girshick, R., He, K., Hariharan, B., Belongie, S., 2017. Feature pyramid networks for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 2117–2125.

Lu, J., Li, S., Guo, W., Zhao, M., Yang, J., Liu, Y., Zhou, Z., 2023. Siamese graph attention networks for robust visual object tracking. Comput. Vis. Image Underst. 229, 103634.

Ma, Z., Wang, L., Zhang, H., Lu, W., Yin, J., 2020. Rpt: Learning point set representation for siamese visual tracking. In: European Conference on Computer Vision. ECCV, Springer, pp. 653–665.

Meinhardt, T., Kirillov, A., Leal-Taixe, L., Feichtenhofer, C., 2022. Trackformer: Multi-object tracking with transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8844–8854.

Milan, A., Leal-Taixé, L., Reid, I., Roth, S., Schindler, K., 2016. MOT16: A benchmark for multi-object tracking. arXiv preprint arXiv:1603.00831.

Newell, A., Huang, Z., Deng, J., 2017. Associative embedding: End-to-end learning for joint detection and grouping. Adv. Neural Inf. Process. Syst. 30.

Pang, B., Li, Y., Zhang, Y., Li, M., Lu, C., 2020. 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. CVPR, pp. 6308–6318.

Pang, J., Qiu, L., Li, X., Chen, H., Li, Q., Darrell, T., Yu, F., 2021. Quasi-dense similarity learning for multiple object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 164–173.

Peng, J., Wang, C., Wan, F., Wu, Y., Wang, Y., Tai, Y., Wang, C., Li, J., Huang, F., Fu, Y., 2020. Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking. In: European Conference on Computer Vision. ECCV, Springer, pp. 145–161.

Ren, S., He, K., Girshick, R., Sun, J., 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. Adv. Neural Inf. Process. Syst. 28.

Ristani, E., Solera, F., Zou, R., Cucchiara, R., Tomasi, C., 2016. Performance measures and a data set for multi-target, multi-camera tracking. In: European Conference on Computer Vision. ECCV, Springer, pp. 17–35.

Shao, S., Zhao, Z., Li, B., Xiao, T., Yu, G., Zhang, X., Sun, J., 2018. Crowdhuman: A benchmark for detecting human in a crowd. arXiv preprint arXiv:1805.00123.

Shuai, B., Berneshawi, A., Li, X., Modolo, D., Tighe, J., 2021. Siammot: Siamese multi-object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 12372–12382.

Stadler, D., Beyerer, J., 2021. Improving multiple pedestrian tracking by track management and occlusion handling. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 10958–10967.

Sun, P., Cao, J., Jiang, Y., Zhang, R., Xie, E., Yuan, Z., Wang, C., Luo, P., 2020. Transtrack: Multiple object tracking with transformer. arXiv preprint arXiv:2012.15460.

Tang, P., Wang, C., Wang, X., Liu, W., Zeng, W., Wang, J., 2019. Object detection in videos by high quality object linking. IEEE Trans. Pattern Anal. Mach. Intell. 42 (5), 1272–1278.

Tokmakov, P., Li, J., Burgard, W., Gaidon, A., 2021. Learning to track with object permanence. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. ICCV, pp. 10860–10869.

Wang, Y., Kitani, K., Weng, X., 2020a. Joint object detection and multi-object tracking with graph neural networks. arXiv:2006.13164.

Wang, Z., Zheng, L., Liu, Y., Li, Y., Wang, S., 2020b. Towards real-time multi-object tracking. In: Proceedings of the European Conference on Computer Vision. ECCV, Springer, pp. 107–122.

Wang, Q., Zheng, Y., Pan, P., Xu, Y., 2021. Multiple object tracking with correlation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 3876–3886.

Wojke, N., Bewley, A., Paulus, D., 2017. Simple online and realtime tracking with a deep association metric. In: 2017 IEEE International Conference on Image Processing. ICIP, IEEE, pp. 3645–3649.

Wu, J., Cao, J., Song, L., Wang, Y., Yang, M., Yuan, J., 2021. 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.

Yang, F., Choi, W., Lin, Y., 2016. Exploit all the layers: Fast and accurate CNN object detector with scale dependent pooling and cascaded rejection classifiers. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 2129–2137.

Yin, J., Wang, W., Meng, Q., Yang, R., Shen, J., 2020. A unified object motion and affinity model for online multi-object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 6768–6777.

Yu, E., Li, Z., Han, S., 2022a. Towards discriminative representation: multi-view trajectory contrastive learning for online multi-object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 8834–8843.

Yu, E., Li, Z., Han, S., Wang, H., 2022b. Relationtrack: Relation-aware multiple object tracking with decoupled representation. IEEE Trans. Multimed..

Yu, F., Wang, D., Shelhamer, E., Darrell, T., 2018. Deep layer aggregation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. CVPR, pp. 2403–2412.

Zhang, Y., Wang, C., Wang, X., Zeng, W., Liu, W., 2021. Fairmot: On the fairness of detection and re-identification in multiple object tracking. Int. J. Comput. Vis. 1–19.

Zheng, L., Tang, M., Chen, Y., Zhu, G., Wang, J., Lu, H., 2021. Improving multiple object tracking with single object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. CVPR, pp. 2453–2462.

Zhou, X., Koltun, V., Krähenbühl, P., 2020. Tracking objects as points. In: European Conference on Computer Vision. ECCV.

Zhou, X., Wang, D., Krähenbühl, P., 2019. Objects as points. arXiv preprint arXiv:1904.07850.

Zhou, X., Yin, T., Koltun, V., Krähenbühl, P., 2022. Global tracking transformers. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8771–8780.