


ORIGINAL ARTICLE

Open Access



End-to-End Joint Multi-Object Detection and Tracking for Intelligent Transportation Systems

Qing Xu¹, Xuewu Lin², Mengchi Cai^{1*} , Yu-ang Guo³, Chuang Zhang¹, Kai Li⁴, Keqiang Li¹, Jianqiang Wang¹ and Dongpu Cao⁵

Abstract

Environment perception is one of the most critical technology of intelligent transportation systems (ITS). Motion interaction between multiple vehicles in ITS makes it important to perform multi-object tracking (MOT). However, most existing MOT algorithms follow the tracking-by-detection framework, which separates detection and tracking into two independent segments and limit the global efficiency. Recently, a few algorithms have combined feature extraction into one network; however, the tracking portion continues to rely on data association, and requires complex post-processing for life cycle management. Those methods do not combine detection and tracking efficiently. This paper presents a novel network to realize joint multi-object detection and tracking in an end-to-end manner for ITS, named as global correlation network (GCNet). Unlike most object detection methods, GCNet introduces a global correlation layer for regression of absolute size and coordinates of bounding boxes, instead of offsetting predictions. The pipeline of detection and tracking in GCNet is conceptually simple, and does not require complicated tracking strategies such as non-maximum suppression and data association. GCNet was evaluated on a multi-vehicle tracking dataset, UA-DETRAC, demonstrating promising performance compared to state-of-the-art detectors and trackers.

Keywords Intelligent transportation systems, Joint detection and tracking, Global correlation network, End-to-end tracking

1 Introduction

Environment perception is one of the most critical technology of intelligent transportation systems (ITS), because its performance has an important impact on

subsequent process of decision making and vehicle control [1–4]. The complex motion interaction between multiple vehicles in ITS makes it important to perform multi-object tracking (MOT) from the view of both vehicles and roadside [5, 6]. MOT is a basic problem in environment perception, whose goal is to calculate the trajectories of all interested objects from consecutive frames of images. It has a wide range of application scenarios, such as autonomous driving, motion attitude analysis, and traffic monitoring. Recently, MOT has been receiving increasing attention.

Traditional MOT algorithms follow the tracking-by-detection framework, which is split into two modules: Detection and tracking. With the development of object detection, these algorithms achieve excellent

*Correspondence:

Mengchi Cai
caimengchi@tsinghua.edu.cn

¹ School of Vehicle and Mobility, Tsinghua University, Beijing 100084, China

² Horizon Information Technology Co., Ltd., Beijing 100096, China

³ School of Transportation Science and Engineering, Beihang University, Beijing 100191, China

⁴ Dongfeng USharing Technology Co., Ltd., Wuhan 430000, China

⁵ Department of Mechanical and Mechatronics Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada



© The Author(s) 2023. **Open Access** This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

performance, approximately dominating the entire MOT domain. The tracking module in a tracking-by-detection framework generally contains three parts: Feature extraction, data association, and lifecycle management. Early tracking methods use simple features to accomplish data association, such as location, shape, and velocity; however, these features have evident deficiencies. Later methods utilize appearance features, especially high-level features from deep neural networks. These appearance features can significantly improve the association accuracy and robustness; however, it leads to an increase in the required calculation. Currently, a few MOT algorithms integrate feature extraction into the detection module, which adds the ReID head to obtain instance-level features for data association. Although these algorithms require less computation, data association is still required to perform motion prediction and set complex tracking strategies, resulting in surplus hyperparameters and a cumbersome inference pipeline.

This paper presents a novel network for end-to-end joint detection and tracking. The network realizes bounding box regression and tracking in the same manner, known as global correlation. Notably, bounding box regression generally uses local features to estimate the offsets between the anchor and ground truth, or to estimate the box size and offset between the key point and feature location. In this paper, the proposed framework intends to regress the absolute coordinate and size of the bounding box, rather than the relative coordinate, or offset. However, in traditional convolutional neural networks, the local feature cannot contain global information when the receptive field is considerably small. The self-attention mechanism allows the features of each location to contain global information; however, its computational complexity is too large to be used

on a high-resolution feature map. Hence, this paper introduces the global correlation layer to encode global information into features at each location. Moreover, the correlation vectors generated by the global correlation layer can encode the correlation between the local feature vector Q with the global feature map K . Q and K from the image in the same frame are used while performing object detection; conversely, Q from the image in the previous frame and K from the image in the current frame are used while performing object tracking. In this manner, this paper unifies detection and tracking under the same framework.

This paper performs algorithm evaluation on a vehicle tracking dataset, UA-DETRAC, which is captured from a roadside view, and can be seen as a typical application of environment perception in ITS. GCNet demonstrated competitive performance with 74.04% average precision (AP) and 36 frame/s in detection, 19.10% PR-MOTA and 34 frame/s in tracking. Figure 1 shows some examples of tracking results. To summarize, the main contributions of this paper are as follows:

- (1) This paper proposes a novel network GCNet to realize end-to-end joint multi-object detection and tracking, serving for onboard and roadside perception of ITS.
- (2) This paper develops the global correlation layer of GCNet that can encode correlation between the local feature vectors with the global feature map without computational complexity.
- (3) This paper demonstrates the competitive performance of the GCNet by comparative experiments on UA-DETRAC dataset. The results show the advantages of the proposed framework in both detecting and tracking process.



Figure 1 Examples of tracking results on UA-DETRAC dataset

The following of this paper is organized as follows. Section 2 introduces the existing research that is related to this paper. Section 3 provides the methodology of this paper, including network components and implementation details. Section 4 conducts experiments and Section 5 gives the conclusions.

2 Related Works

2.1 Object Detection

With the advancements in deep learning, object detection technology has developed rapidly. Existing object detection algorithms can be divided into two categories: Anchor-based [7–9] and anchor-free [10–12]. Anchor-based algorithms set a series of anchor boxes and regress offsets between the anchor boxes and ground truth using local features. The methods based on region convolution neural network (R-CNN) utilizes heuristic algorithm [13] and region proposal network (RPN) [7, 14, 15] to generate region proposals as anchor. Most anchor-free algorithms use full convolution networks to estimate the key points of targets, and further obtain the bounding boxes through the key points. These algorithms consider local features for bounding box regression, such that they only obtain the offsets between the anchor boxes or key points and the ground truth, rather than absolute bounding box coordination. Detection transformer (DETR) [12] adopted an encoder-decoder architecture based on transformers to achieve object detection. A transformer can integrate global information into the features at each position; however, the self-attention mechanism of the transformer requires a considerable amount of computation and GPU memory, which is difficult to apply to high-resolution feature maps. In the proposed joint detection and tracking framework, the network detects objects in a single image, and tracks objects in different images. However, the offsets for the same object in different images are hard to define. Hence, this paper introduces a global correlation layer to embed global information into the features at each position for absolute coordinate regression, which can be applied to higher-resolution feature maps, rather than the transformer.

2.2 Tracking-by-Detection

With the improvement in detection accuracy, tracking-by-detection methods [16–18] have become mainstream in the field of MOT. Tracking is considered as a data association problem in tracking-by-detection frameworks. Features, such as motion [19], shape [20], and appearance [21, 22], are used to describe the correlation between detections and tracks, and thus, a correlation matrix is established. Algorithms including the Hungary algorithm [23], JPDA [16] and MHT [24], input the correlation matrix to complete data association. Although these algorithms have made

significant progress, there are certain drawbacks. First, they do not combine the detector and tracker efficiently, and a majority of them need to perform feature extraction separately, which involves unnecessary computation. Second, they often rely on complicated tracking rules for lifecycle management, resulting in numerous hyperparameters and difficult tuning. In the proposed approach, detection and tracking are performed in the same manner, such that they are well combined and the computation of feature extraction is reduced. Additionally, the proposed approach eliminates the complex tracking rules.

2.3 Joint Detection and Tracking

In the field of MOT, it is an important research direction to combine detection and tracking. With the quick maturity of multi-task learning in deep learning, many methods using a single network to complete detection and tracking tasks by adding ReID feature extraction to existing object detection networks [25–27]. Wang et al. [28] proposed the joint detection and embedding (JDE) method that allows target detection and appearance embedding to be learned in a shared model. Bergmann et al. [29] proposed a JDT method that adopts Faster-RCNN framework, and accomplishes tracking by region of interest (RoI) pooling and bounding box regression without data association. Zhou et al. [10] considered current and previous frames as well as a heatmap, rendered from tracked object centers, as inputs, and produces an offset map, which simplifies data association considerably. Peng et al. [30] converted the MOT problem into a pair-wise object detection problem, and proposed chained-tracker method realizing end-to-end joint object detection and tracking. Similarly, this study also provides a new idea for joint detection and tracking. Compared with trackformer [31], which formulate the MOT task as a frame-to-frame set prediction problem and propose a tracking-by-attention network based on DETR [12], the network structure of GCNet is simpler and can reach a higher inference speed.

3 Methodology of Global Correlation Network

The proposed network is designed to solve the online MOT problem. At time step t , the network obtains the object trajectories $\{T_1, T_2, \dots, T_n\}$ from time 0 to time $t - 1$, where $T_i = [B_{i,1}, B_{i,2}, \dots, B_{i,t-1}]$ and $B_{i,j}$ are the bounding box of the object i at time j . Considering an image of the current frame $I_t \in \mathbb{R}^{h \times w \times 3}$, the network assigns the bounding boxes $B_{x,t}$ of objects in the current frame to historical trajectories, or generates new trajectories. The following section introduces the proposed algorithm in detail.

3.1 Global Correlation Network

In this part, the global correlation layer and its application principle in end-to-end joint detection and tracking framework are introduced. Furthermore, the specific implementation of detection module and tracking module in the proposed GCNet are described.

Global correlation layer: The global correlation layer in GCNet encodes global information to generate the correlation vectors, which can be utilized in detection module and tracking module. Using feature map $F \in R^{h \times w \times c}$, two feature maps Q and K are obtained from the following two linear transformations:

$$Q_{ij} = W_q F_{ij}, K_{ij} = W_k F_{ij}, \quad (1)$$

where $F_{ij} \in R^c$ denotes the feature vector at the i th row and j th column of F . Further, for each feature vector Q_{ij} , the cosine distance between it and all K_{ij} is calculated. Following another linear transformation \dot{W} , the correlation vectors $C_{ij} \in R^{c'}$ is obtained:

$$C_{ij} = \dot{W} \cdot \text{flatten} \left(\begin{bmatrix} \frac{Q_{ij}K_{11}}{|Q_{ij}||K_{11}|} & \cdots & \frac{Q_{ij}K_{1w}}{|Q_{ij}||K_{1w}|} \\ \vdots & \ddots & \vdots \\ \frac{Q_{ij}K_{h1}}{|Q_{ij}||K_{h1}|} & \cdots & \frac{Q_{ij}K_{hw}}{|Q_{ij}||K_{hw}|} \end{bmatrix} \right). \quad (2)$$

These correlation vectors C_{ij} encode the correlation between the local feature vectors Q_{ij} with the global feature map K , such that it can be used to regress the absolute bounding boxes for the objects at the corresponding positions in the image. All of the correlation vectors C_{ij} can form a correlation map $C \in R^{h \times w \times c'}$, allowing us to obtain bounding boxes $B \in R^{h \times w \times 4}$ using a convolution layer with 1×1 kernel size. K and Q from the image in the same frame are used while performing object detection; conversely, Q from the image in the previous frame and K from the image in the current frame are used while performing object tracking. In this manner, detection and tracking are unified under the same framework.

Compared with the traditional self-attention layer, the global correlation layer has advantage in computation. The computation of traditional self-attention layer includes three parts: Computing attention weight, $c \times (h \times w) \times (h \times w)^T = ch^2w^2$; SoftMax, chw ; weighted summation, $c \times (h \times w) \times (h \times w)^T = ch^2w^2$. As shown in Eq. (2), the computation of global correlation layer is $c \times (h \times w) \times (h \times w)^T = ch^2w^2$, which is significantly less than that of the total computation of self-attention layer $(2c + 1)h^2w^2$.

In terms of object classification brunch, this study uses the same network structure and training strategy as CenterNet. When infers, a detection heatmap Y_d and tracking heatmap Y_t are obtained in each frame. The detection

heatmap Y_d denotes the detection confidence of the object centers in the current frame, while the tracking heatmap Y_t denotes the tracking confidence between the current and next frame. The peaks in the heatmaps correspond to the detection and tracking key points, and max-pooling is used to obtain the final bounding boxes, without applying box non-maximum suppression (NMS).

$$B_{ij} \in Result, \forall i, j \rightarrow \text{maxpool}(Y, 3, 1)_{ij} = Y_{ij}, \quad (3)$$

where $\text{maxpool}(H, a, b)$ represents a max-pooling layer with kernel size a and stride b . Hence, the GCNet can realize joint multi-object detection (MOD) and MOT, without complicated post-processes, such as NMS and data association, which have a concise pipeline.

Detection module: The detection module architecture is depicted as Figure 2, which contains three parts: Backbone, classification branch, and regression branch. The backbone is for high-level feature extraction. Because the classification is identical to CenterNet, each location of the feature map corresponds to an object center point, while the resolution of the feature map crucially affects the network performance. To obtain high resolution and retain a large receptive field, the same skip connection structure is acquired as a feature pyramid network (FPN); however, it only outputted the finest level feature map F . The size of the feature map F is $h' \times w' \times c$, which is equivalent to $\frac{h}{8} \times \frac{w}{8} \times c$; here, h and w are the height and width of the original image, respectively. This resolution is 4 times that of DETR. The classification branch is a full convolution network, and outputs a confidence map $Y_d \in R^{h' \times w' \times n}$ with values between 0 and 1. The peaks of the i th channel of Y_d correspond to the centers of the objects belonging to the i th category. The regression branch is used to calculate bounding boxes $\{[x, y, h, w]_i | 1 \leq i \leq N\}$. First, this paper considers F and Y_d as inputs, and generates three feature maps K , Q , and V .

$$\begin{aligned} Q &= BN_Q(\text{Conv}_Q(F, 1, 1, c) + P), \\ K &= \text{Gate}[BN_K(\text{Conv}_K(F, 1, 1, c) + P), Y_d], \\ V &= \text{Conv}_V(F, 1, 1, c), \end{aligned} \quad (4)$$

where $\text{Conv}(F, a, b)$ denotes a convolution layer with kernel size a , strides b and kernel number c , and BN denotes batch normalization layer. $\text{Gate}(X, Y)$ is depicted in Figure 3, which is a form of spatial attention. P is the position embedding with the same shape as F , and is expressed as:

$$P_{ijk} = \begin{cases} \cos\left(\frac{4\pi k}{c} + \frac{\pi i}{h}\right), & 0 \leq k < \frac{c}{2}, \\ \cos\left(\frac{4\pi k}{c} + \frac{\pi j}{w}\right), & \frac{c}{2} \leq k < c, \\ 0 \leq i < h', 0 \leq j < w'. \end{cases} \quad (5)$$

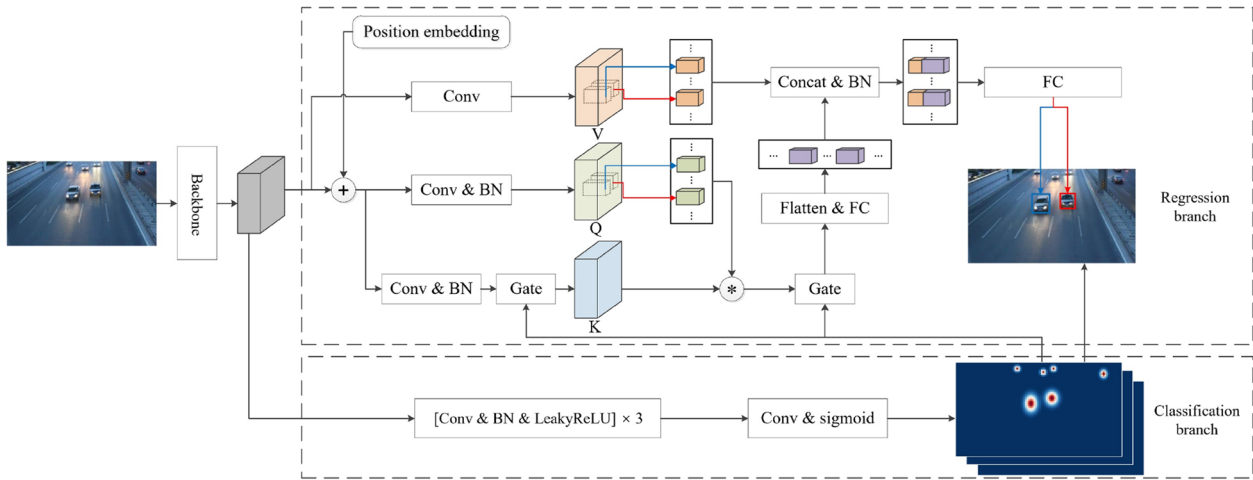


Figure 2 Detection module architecture

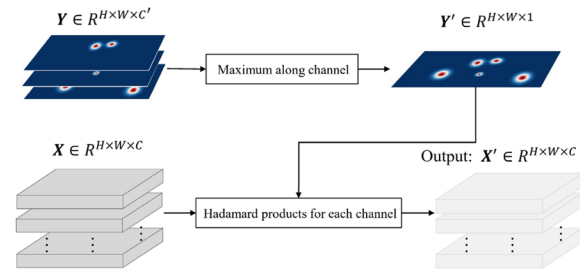


Figure 3 Illustration of gate step

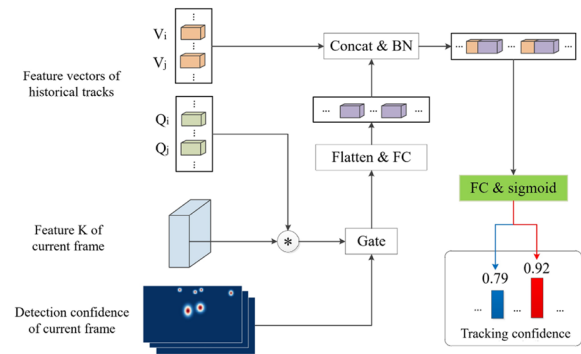


Figure 4 Tracking module architecture

The two embedding vectors that are close in the position have a large cosine similarity, while the two that are farther away have a smaller cosine similarity. This attribute reduces the negative influence of similar objects while tracking. Further, the correlation vectors C_{ij} between Q_{ij} and K are calculated using Eq. (2). The final bounding boxes $B_{d,ij} = [x_{ij}, y_{ij}, h_{ij}, w_{ij}]$ can be obtained using Eq. (6). Here, the absolute coordinates and size of the bounding box are directly regressed, which differs from most existing methods, especially anchor-based methods.

$$B_{d,ij} = W \cdot BN([C_{ij} \ V_{ij}]). \tag{6}$$

Tracking module: Tracking is the process of assigning objects in the current frame to historical tracks, or generating new tracks. The architecture of the tracking module is depicted in Figure 4. The inputs of the tracking module are: (1) Feature map K of the current frame, (2) detection confidence map of the current frame, and (3) feature vectors of historical tracks. Additionally, the tracking module outputs a tracking confidence and bounding box for each historical track. It can be observed, this architecture is almost identical to that of the detection

module. Most of its network parameters are shared with the detection module, except for the fully connected layer used for calculating tracking confidence (the green block in Figure 4). The tracked bounding boxes are consistent with the detected target boxes in terms of expression, which is $B_i = [x_i, y_i, h_i, w_i]$, with absolute coordinates and size. The tracking confidences indicate whether the objects are still present in the image of the current frame. The tracking module functions in an object-wise manner, such that it can naturally pass the ID of each object to the next frame, which is similar to parallel single-object tracking.

3.2 Training

Although the proposed model can be trained end-to-end, the GCNet is trained in two stages in this study. First, the detection module is trained, and then, the entire network is fine-tuned. The training strategy of the classification branch is consistent with CornerNet. A heatmap $Y_{gt} \in R^{h' \times w' \times n}$ with 2D Gaussian kernel is defined as follows:

$$Y_{gt,ijk} = \max_{1 \leq n \leq N_k} (G_{ijn}),$$

$$G_{ijn} = \exp \left[-\frac{(i-x_n)^2}{2\sigma_x^2} - \frac{(j-y_n)^2}{2\sigma_y^2} \right], \quad (7)$$

where N_k is the number of objects of class k , $[x_n, y_n]$ is the center of object n , and variance σ^2 is relative to the object size. σ_x and σ_y are expressed as shown in Eq. (8), and η_{IoU} is set to 0.3.

$$\sigma_x = \frac{h(1-\eta_{IoU})}{3(1+\eta_{IoU})},$$

$$\sigma_y = \frac{w(1-\eta_{IoU})}{3(1+\eta_{IoU})}. \quad (8)$$

The classification loss is a penalty-reduced pixel-wise focal loss.

$$L_{d,cla} = -\frac{1}{h'w'n} \cdot \sum_{ijk} \begin{cases} (1 - Y_{d,ijk})^2 \log(Y_{d,ijk}), & Y_{gt,ijk} = 1, \\ (1 - Y_{gt,ijk})^2 Y_{d,ijk}^2 \log(1 - Y_{d,ijk}), & Y_{gt,ijk} \neq 1. \end{cases} \quad (9)$$

The regression branch is trained using CIoU loss, as follows:

$$L_{d,reg} = \sum_{[ij]=1} \beta_{ij} \cdot L_{CIoU}(B_{gt,ij}, B_{d,ij}), \quad (10)$$

where $[ij] = 1$ indicates the corresponding $B_{d,ij}$, and is assigned to a ground truth. A bounding box $B_{d,ij}$ is assigned to a ground truth if $G_{ijn} > 0.3$ and $\sum_n G_{ijn} - \max_n G_{ijn} < 0.3$.

$$[ij] = \begin{cases} 1, & \sum_n G_{ijn} - \max_n G_{ijn} < 0.3, \\ 0, & \text{otherwise.} \end{cases} \quad (11)$$

Furthermore, for B_{ij} with $\max_n G_{ijn} = 1$, the weight of their regression loss w_{ij} is set to 2, and the other weights to 1. This is done to enhance the precision of the bounding boxes at the center points.

The entire network is fine-tuned using a pretrained detection module. At this training step, two images I_{t-i} and I_t are treated as inputs simultaneously, where i lies between 1 and 5. The loss contains two parts, i.e., detection loss of I_{t-i} and tracking loss between the two images. The tracking loss also comprises two terms, i.e., regression loss and classification loss. The tracking ground truth is determined by object ID. $B_{t,ij}$ and $Y_{t,ij}$ are positive if $[ij]$ in I_{t-i} is equal to 1, and the corresponding objects exist in I_t . The total train loss is expressed as:

$$Loss = L_{d,cla} + L_{t,cla} + 0.1 \times (L_{d,reg} + L_{t,reg}). \quad (12)$$

3.3 Inference Pipeline

The inference pipeline for joint MOD and MOT is described in Algorithm 1. The inputs of the algorithm are consecutive frames of images $I_1 - I_t$. Trajectory T_i , confidence Y_i , and vector $[V_i, Q_i]$ of all tracks and candidates are recorded in four collections: \mathcal{T} , \mathcal{O} , \mathcal{Y} , and \mathcal{C} . At each time step, object detection is performed on the current frame of image I , and tracked the existing track \mathcal{T} and candidate \mathcal{C} . Tracking confidences are used to update all confidences in sets \mathcal{Y} and \mathcal{C} , and obtained $Y_i = \min(2 \times Y_i \times Y_{t,i}, 1.5)$. The tracks and candidates with a confidence lower than p_2 are deleted, and other trajectories, candidates, and corresponding features are updated. This update strategy,

$Y_i = \min(2 \times Y_i \times Y_{t,i}, 1.5)$, provides these tracks with a higher tracking confidence, certain trust margin, and confidence possibly greater than 1. The detections with an IoU greater than p_3 , or confidence less than p_2 , are ignored. For the remaining detections, those with a detection confidence greater than p_1 are used to generate new tracks, and the rest are added to the candidate set \mathcal{C} . As observed, the entire detection and tracking process can be performed in sparse mode, such that the overall computational complexity of the algorithm is extremely low.

4 Experiments of the Algorithm

In this section, experiments are carried out to validate the performance of GCNet. Comparison and ablation study are carried out and the results indicate the advantages of the proposed method.

4.1 Benchmark and Implementation Details

Experiments of this study are conducted using the vehicle detection and tracking dataset, UA-DETRAC, which is captured from a roadside view, and can be seen as a typical application of environment perception in ITS. This dataset contains 100 sequences; 60 were used for training, and the remaining 40 were used for testing. The data in the training and test sets, which are derived from different traffic scenarios, make the test more difficult. The UA-DETRAC benchmark employs AP to rank the performance of the detectors as well as PR-MOTA, PR-MOTP, PR-MT, PR-ML, PR-IDS, PR-FM, PR-FP, and

PR-FN scores for tracking evaluation. This paper refers to Ref. [32] for further details on the metrics.

Algorithm 1: Inference pipeline of GCNet

```

Input: continuous frame images  $I_1 - I_t$ 
Output: object trajectories
 $\mathcal{T} = [T_1, T_2, \dots, T_n]$ ,
 $T_i = [B_{i,1}, B_{i,2}, \dots, B_{i,t-1}]$ ,  $B$  denotes
the bounding box
1 Initialize: Trajectory set  $\mathcal{T} = \emptyset$ ;
confidence set  $\mathcal{Y} = \emptyset$ ; feature
set  $\mathcal{O} = \emptyset$ ; candidate set  $\mathcal{C} = \emptyset$ ;
and hyperparameters  $p_1, p_2$ , and  $p_3$ 
2 for  $I$  in  $I_2 - I_t$  do
3  $Q, K, V, B = \text{DetectionModule}(I)$ ;
4 for  $T_i$  in  $\mathcal{T}$  do
5  $B_{t,i}, Y_{t,i} = \text{TrackingModule}(Q_i, K, V_i)$ ;
6 Update
 $Y_i = \min(2 \times Y_i \times Y_{t,i}, 1.5)$ ;
7 if  $Y_i < p_2$  then
8 Delete  $T_i$  from  $\mathcal{T}$ ;
9 else
10 Add  $B_{t,i}$  to  $T_i$ ;
11 Update  $Q_i = K_{mn}, V_i = V_{mn}$ ,
12 where  $(m, n)$  is the center of  $B_{t,i}$ ;
13 end
14 end
15 for  $C_i = [Y_i, Q_i, V_i, B_i]$  in  $\mathcal{C}$  do
16  $B_{t,i}, Y_{t,i} = \text{TrackingModule}(Q_i, K, V_i)$ ;
17 Update
 $Y_i = \min(2 \times Y_i \times Y_{t,i}, 1.5)$ ;
18 if  $Y_i < p_1$  then
19 Delete  $C_i$  from  $\mathcal{C}$ ;
20 else
21 Add  $B_{t,i}$  to  $T_i$ ;
22 Update  $Q_i = K_{mn}, V_i = V_{mn}$ ,
23 where  $(m, n)$  is the center of  $B_{t,i}$ ;
24 end
25 end
26 for  $B_i$  in  $B_d$  do
27 if  $\exists j, \text{IoU}(B_i, T_j) > p_3$  then
28 continue;
29 else if  $Y_i > p_1$  then
30 Add  $T_{new} = [B_i]$  to  $\mathcal{T}$ ;
31 Add  $[Q_i, V_i]$  to  $\mathcal{O}$ ;
32 Add  $Y_i$  to  $\mathcal{Y}$ ;
33 else if  $Y_i > p_2$  then
34 Add  $[Y_i, Q_i, V_i, B_i]$  to  $\mathcal{C}$ ;
35 end
36 end
37 end

```

All the experiments are performed using TensorFlow 2.0. The proposed model is trained with Adam on the complete training dataset of UA-DETRAC. The size of

the input images is 512×896 . Three commonly used data augmentation methods are employed: Random horizontal flip, random brightness adjustment, and scale adjustment. Hyperparameters p_1, p_2 , and p_3 for the inference are set to 0.5, 0.3, and 0.5 respectively.

4.2 Ablation Study

In the proposed joint detection and tracking framework, three main components influence the performance: 1) Gate by confidence map Y_d ; 2) concatenated feature vector in V for bounding box regression; and 3) specially designed position embedding P . The detection effects of the three models are compared with the GCNet to demonstrate the effectiveness of these components. Table 1 shows the results of the comparison. The full version of the GCNet exhibited the best performance, with 74.04% AP on UA-DETRAC. The gate and feature vector of V both yielded 2% AP. The gate step explicitly merges the classification result into the regression branch, which plays a role of spatial attention and is conducive to the training of the regression branch. The concatenated feature vectors of V for regression introduce more texture and local information, which is not included in the correlation vectors. This information is beneficial for inferring the size of the objects. To demonstrate the role of the position embedding, it is replaced with a normal explicit position embedding, where P_{ijk} equals i when $0 \leq k < c/2$, and equals j when $c/2 \leq k < c$. Notably, the self-designed position embedding attains a 5.80% increase in AP.

The ablation study is conducted only on the detection benchmark. This is because the tracking module shares most of its parameters with the detection module, and the tracking performance is highly correlated with the detection performance. The results of the ablation study can thus be extended to the tracking module.

4.3 Benchmark Evaluation

Table 2 shows the results obtained using the UA-DETRAC detection benchmark. The GCNet demonstrates promising performance, and outperforms

Table 1 Ablation study results

Model	AP			
	Full	Easy	Medium	Hard
GCNet	74.04	91.57	81.45	59.43
Without gate	71.62	88.49	78.99	57.56
Without V	71.71	90.29	78.13	57.65
Explicit position embedding	68.24	85.28	75.59	54.61

Bold values indicate the best scores of each single item

Table 2 Results on the UA-DETRAC detection benchmark

Model	AP						
	Full	Easy	Medium	Hard	Sunny	Night	Rainy
DPM	25.70	34.42	30.29	17.62	31.77	30.91	25.55
ACF	46.35	54.27	51.52	38.07	66.58	39.32	39.06
R-CNN	48.95	59.31	54.06	39.47	67.52	39.32	39.06
CompACT	53.23	64.84	58.70	43.16	71.16	46.37	44.21
Faster R-CNN	62.13	86.14	66.77	47.29	73.83	69.28	49.03
EB	67.96	89.65	73.12	53.64	83.73	73.93	53.40
R-FCN	69.87	93.32	75.67	54.31	84.08	75.09	56.21
CenterNet-Res50	63.85	83.35	70.19	49.56	80.09	62.54	50.91
GCNet	74.04	91.57	81.45	59.43	83.53	78.50	65.38

Bold values indicate the best scores of each single item

most detection algorithms on this benchmark. It attains a high AP on full and medium difficulty as well as on night and rainy images of the test set. Figure 5 shows the PR curves of the GCNet and other algorithms, exposed by the UA-DETRAC dataset. It can be observed that the proposed model is far more effective than the baselines in each scenario. Notably, the proposed model does not employ any other components for better precision, and the backbone network is the original version of ResNet50. Compared with other methods, the performance improvement of GCNet benefits from the global correlation mechanism in the model. In the complex traffic scenarios, there are many non-critical areas such as trees and buildings, as well as many traffic participants with similar appearances. When using correlation convolution for object detection, the correlation between different objects will decrease with the increase of the distance, which can effectively reduce the false and missed detection. When only the detection module of the GCNet is used, it can run at 36 frame/s on a single Nvidia 2080Ti.

The aim of designing GCNet considers both MOD and MOT. This is the real purpose of introducing the global correlation layer to regress the absolute coordinates. The tracking results are shown in Table 3. The MOT metrics with “PR-” can evaluate the overall effect of detection and tracking. EB and KIoU are the UA-DETRAC challenge winners. In the process of multi-objects tracking, the pixel coordinate distance of the same target among continuous frame images is generally close. Benefiting from the position embedding and global correlation, our

method can encode spatiotemporal motion of tracking target implicitly, which can improve the matching accuracy between trajectory in the precious frames and detection results in the current frame. Additionally, a significant PR-MOTA score and an excellent PR-MOTP score are obtained, approximately twice as high as that of EB and KIoU combined. Moreover, the leading scores are obtained in PR-ML and PR-FN on the UA-DETRAC tracking benchmark. Because the detection and tracking modules share most of the features, calculating the entire joint detection and tracking pipeline is approximately the same as calculating detection alone, and it can achieve a speed of approximately 34 frame/s.

5 Conclusions

This paper proposes a novel joint MOD and MOT network, called GCNet. A global correlation layer is introduced to achieve absolute coordinate and size regression, which performs object detection on a single image, and naturally propagates the ID of objects to the subsequent consecutive frames. Compared to existing tracking-by-detection methods, the GCNet calculates end-to-end object trajectories without a bounding box NMS, data association, and other complex tracking strategies. The proposed method is evaluated on the UA-DETRAC, a vehicle detection and tracking dataset. The results of the experiments indicate that:

- (1) The evaluation results demonstrate the effectiveness of the proposed approach outperforms the existing methods in both detection and tracking.

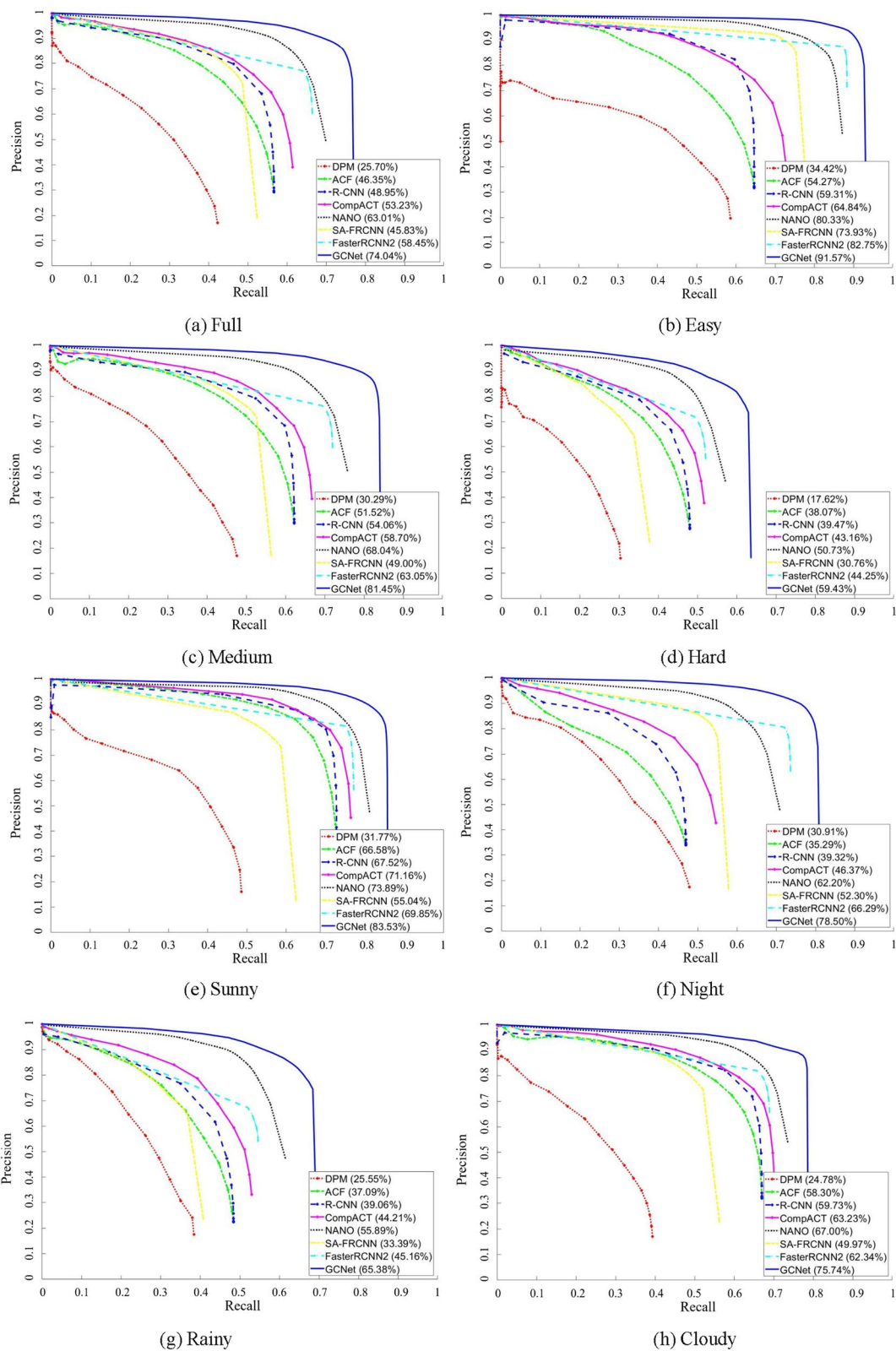


Figure 5 Precision and recall curves of the detection algorithms

Table 3 Results on the UA-DETRAC tracking benchmark

Model	PR-MOTA	PR-MOTP	PR-MT	PR-ML	PR-IDS	PR-FM	PR-FP	PR-FN
DPM+GOG	5.5	28.2	4.1	27.7	1873.9	1988.5	38957.6	230126.6
ACF+GOG	10.8	37.6	12.2	22.3	3850.8	3987.3	45201.5	197094.2
R-CNN+DCT	11.7	38.0	10.1	22.8	758.7	742.9	36561.2	210855.6
CompACT+TBD	13.6	37.3	15.3	19.3	2026.9	2467.3	43247.8	173837.3
CompACT+GOG	14.2	37.0	13.9	19.9	3334.6	3172.4	32092.9	180183.8
Faster R-CNN+MHT	14.5	32.5	15.9	19.1	492.3	576.7	18141.4	156227.8
EB+IoU	19.4	28.9	17.7	18.4	2311.3	2445.9	14796.5	171806.8
EB+KIoU	21.1	28.6	21.9	17.6	462.2	712.1	19046.9	159178.3
GCNet	19.1	57.3	20.9	9.6	755.8	994.5	17660.9	148517.5

Bold values indicate the best scores of each single item

(2) This approach is also equipped to run 36 frame/s for detection and 34 frame/s for joint detection and tracking, thereby meeting the real-time requirements of most application scenarios, such as onboard environment perception of autonomous vehicles, and roadside perception of ITS.

Acknowledgements

Not applicable.

Authors' Contributions

QX, KL, KQL, JW, and DC are in charge of the whole trial; XL and MC write the manuscript; YG and CZ assist with experiments and analysis. All authors read and approved the final manuscript.

Authors' Information

Qing Xu, received his B.S. and M.S. degrees in automotive engineering from *Beihang University, China*, in 2006 and 2008 respectively, and the Ph.D. degree in automotive engineering from *Beihang University, China*, in 2014. During his Ph.D. research, he worked as a visiting scholar with *Department of Mechanical Science and Engineering, University of Illinois, Urbana–Champaign, USA*. From 2014 to 2016, he had his postdoctoral research at *Tsinghua University, China*. He is currently working as an assistant research professor with *School of Vehicle and Mobility, Tsinghua University, China*. His main research interests include decision and control of intelligent vehicles.

Xuewu Lin, received the B.E. degree from *Beijing University of Technology, China*, in 2018, and the M.S. degree from *Tsinghua University, China*, in 2021. He is currently an engineer at *Horizon Information Technology Co., Ltd., China*. His research interests include object detection, multi-object tracking and trajectory prediction.

Mengchi Cai, received his B.E. degree and Ph. D. degree from *School of Vehicle and Mobility, Tsinghua University, China*, in 2018 and 2023, respectively. He is currently a postdoctoral researcher at *Intelligent and Connected Vehicles Lab, School of Vehicle and Mobility, Tsinghua University, China*. His research interests include connected and automated vehicles, multi-vehicle formation control, and unsignalized intersection cooperation.

Yu-ang Guo, received the B.E. degree in computer science from *Beijing Institute of Technology, China*, in 2014. He received the M.S. degree in software engineering from *Northwest University, China*, in 2018. He is currently a Ph.D. candidate at *School of Transportation Science and Engineering, Beihang University, China*. His current research interests include road detection, image segmentation and multi-sensor fusion.

Chuang Zhang, received the B.E. degree from *Southeast University, China*, in 2017, and the M.S. degree from *Beihang University, Beijing, China*, in 2020. He is currently a Ph.D. candidate in mechanical engineering at *School of Vehicle and Mobility, Tsinghua University, China*. His research interests include object detection, multi-object tracking and multi-sensor fusion.

Kai Li, is the CEO of *Dongfeng USharing Technology Co., Ltd., China*. He is now a senior engineer. His research interests include autonomous vehicles, intelligent information interaction, and automotive electronics architecture.

Keqiang Li, received the B.E. degree from *Tsinghua University, Beijing, China*, in 1985, and the M.S. and Ph.D. degrees in mechanical engineering from *Chongqing University, China*, in 1988 and 1995, respectively. He is currently a professor at *School of Vehicle and Mobility, Tsinghua University, China*. His main research areas include automotive control system, driver assistance system, and networked dynamics and control. He is leading National Key Project on CAVs (Intelligent and Connected Vehicles) in China.

Jianqiang Wang, received the B.E. and M.S. degrees from *Jilin University of Technology, China*, in 1994 and 1997, respectively, and the Ph.D. degree from *Jilin University, China*, in 2002. He is currently a professor at *School of Vehicle and Mobility, Tsinghua University, China*. His active research interests include intelligent vehicles, driving assistance systems, and driver behavior.

Dongpu Cao, received the Ph.D. degree from *Concordia University, Canada*, in 2008. He is the Canada research chair of *Driver Cognition and Automated Driving*, and an associate professor at *University of Waterloo, Canada*. His current research focuses on driver cognition, automated driving and cognitive autonomous driving.

Funding

Supported by National Key Research and Development Program of China (Grant No. 2021YFB1600402), National Natural Science Foundation of China (Grant No. 52072212), Dongfeng USharing Technology Co., Ltd., China Intelligent and Connected Vehicles (Beijing) Research Institute Co., Ltd., and "Shuimu Tsinghua Scholarship" of Tsinghua University of China.

Declarations

Competing Interests

The authors declare no competing financial interests.

Received: 23 November 2021 Revised: 11 October 2023 Accepted: 19 October 2023

Published online: 20 November 2023

References

- [1] Y Liu, X Guan, P Lu, et al. Research on key issues of consistency analysis of vehicle steering characteristics. *Chinese Journal of Mechanical Engineering*, 2021, 34: 11.
- [2] Q Xu, M Cai, K Li, et al. Coordinated formation control for intelligent and connected vehicles in multiple traffic scenarios. *IET Intelligent Transport Systems*, 2021, 15(1): 159-173.
- [3] Y Luo, D Yang, M Li, et al. Hardware-in-the-loop simulation on dynamical coordinated control method in parallel hybrid electric

- vehicle (PHEV). *Chinese Journal of Mechanical Engineering*, 2008, 44(5): 80-85.
- [4] M Cai, Q Xu, C Chen, et al. Formation control with lane preference for connected and automated vehicles in multi-lane scenarios. *Transportation Research Part C: Emerging Technologies*, 2022, 136: 103513.
- [5] C Chen, M Cai, J Wang, et al. Cooperation method of connected and automated vehicles at unsignalized intersections: Lane changing and arrival scheduling. *IEEE Transactions on Vehicular Technology*, 2022, 71(11): 11351-11366.
- [6] M Cai, Q Xu, C Chen, et al. Formation control for connected and automated vehicles on multi-lane roads: Relative motion planning and conflict resolution. *IET Intelligent Transport Systems*, 2023, 17(1): 211-226.
- [7] S Ren, K He, R Girshick, et al. Faster R-CNN: Towards real-time object detection with region proposal networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2016, 39(6): 1137-1149.
- [8] T Y Lin, P Goyal, R Girshick, et al. Focal loss for dense object detection. *Proceedings of the IEEE International Conference on Computer Vision*, 2017: 2980-2988.
- [9] R Girshick. Fast R-CNN. *Proceedings of the IEEE International Conference on Computer Vision*, 2015: 1440-1448.
- [10] X Zhou, D Wang, P Krähenbühl. Tracking objects as points. *Computer Vision—ECCV 2020: 16th European Conference*, Glasgow, UK, August 23–28, 2020: 474-490.
- [11] H Law, J Deng. Cornernet: Detecting objects as paired keypoints. *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018: 734-750.
- [12] N Carion, F Massa, G Synnaeve, et al. End-to-end object detection with transformers. *Computer Vision—ECCV 2020: 16th European Conference*, Glasgow, UK, August 23–28, 2020: 213-229.
- [13] A Farhadi, J Redmon. Yolov3: An incremental improvement. *Computer Vision and Pattern Recognition*, Berlin/Heidelberg, Germany, 2018, 1804: 1-6.
- [14] C Y Fu, W Liu, A Ranga, et al. Dssd: Deconvolutional single shot detector. arXiv preprint arXiv: 1701.06659, 2017.
- [15] K He, G Gkioxari, P Dollár, et al. Mask R-CNN. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018, 42(2): 386-397.
- [16] S H Rezaatofghi, A Milan, Z Zhang, et al. Joint probabilistic data association revisited. *Proceedings of the IEEE International Conference on Computer Vision*, 2015: 3047-3055.
- [17] A Bewley, Z Ge, L Ott, et al. Simple online and realtime tracking. *2016 IEEE International Conference on Image Processing (ICIP)*, 2016: 3464-3468.
- [18] N Wojke, A Bewley, D Paulus. Simple online and realtime tracking with a deep association metric. *2017 IEEE International Conference on Image Processing (ICIP)*, 2017: 3645-3649.
- [19] E Bochinski, V Eiselein, T Sikora. High-speed tracking-by-detection without using image information. *14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2017: 1-6.
- [20] M Ullah, F A Cheikh, A S Imran. Hog based real-time multi-target tracking in bayesian framework. *13th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2016: 416-422.
- [21] E Ristani, C Tomasi. Features for multi-target multi-camera tracking and re-identification. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018: 6036-6046.
- [22] X Shi, H Ling, Y Pang, et al. Rank-1 tensor approximation for high-order association in multi-target tracking. *International Journal of Computer Vision*, 2019, 127(8): 1063-1083.
- [23] A Sadeghian, A Alahi, S Savarese. Tracking the untrackable: Learning to track multiple cues with long-term dependencies. *Proceedings of the IEEE International Conference on Computer Vision*, 2017: 300-311.
- [24] C Kim, F Li, A Ciptadi, et al. Multiple hypothesis tracking revisited. *Proceedings of the IEEE International Conference on Computer Vision*, 2015: 4696-4704.
- [25] Y Zhang, C Wang, X Wang, et al. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *International Journal of Computer Vision*, 2021, 129(11): 3069-3087.
- [26] Z Lu, V Rathod, R Votel, et al. Retinatrack: Online single stage joint detection and tracking. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020: 14668-14678.
- [27] P Voigtlaender, M Krause, A Osep, et al. Mots: Multi-object tracking and segmentation. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2019: 7942-7951.
- [28] Z Wang, L Zheng, Y Liu, et al. Towards real-time multi-object tracking. *Computer Vision—ECCV 2020: 16th European Conference*, Glasgow, UK, August 23–28, 2020: 107-122.
- [29] P Bergmann, T Meinhardt, L Leal-Taixe. Tracking without bells and whistles. *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2019: 941-951.
- [30] J Peng, C Wang, F Wan, et al. Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking. *Computer Vision—ECCV 2020: 16th European Conference*, Glasgow, UK, August 23–28, 2020: 145-161.
- [31] T Meinhardt, A Kirillov, L Leal-Taixe, et al. Trackformer: Multi-object tracking with transformers. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022: 8844-8854.
- [32] L Wen, D Du, Z Cai, et al. UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking. *Computer Vision and Image Understanding*, 2020, 193: 102907.

Submit your manuscript to a SpringerOpen[®] journal and benefit from:

- Convenient online submission
- Rigorous peer review
- Open access: articles freely available online
- High visibility within the field
- Retaining the copyright to your article

Submit your next manuscript at ► [springeropen.com](https://www.springeropen.com)
