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

Qing Xu

Tsinghua University

Xuewu Lin

Horizon Information Technology Co Ltd

Mengchi CAI (✉ cmc18@mails.tsinghua.edu.cn)

Tsinghua University <https://orcid.org/0000-0002-8681-7067>

Yu-ang Guo

Tsinghua University

Chuang Zhang

Tsinghua University

Kai Li

Dongdeng USharing Technology Co Ltd

Keqiang Li

Tsinghua University

Jianqiang Wang

Tsinghua University

Dongpu Cao

University of Waterloo

Original Article

Keywords: Intelligent Transportation Systems, Joint Detection and Tracking, Global Correlation Network, End-to-End Tracking

Posted Date: November 30th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-1107274/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

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, 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

I. INTRODUCTION

Environment perception is one of the most critical technology of ITS, since 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 MOT from the view of both vehicles and road side [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 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 vectors 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 perfor-

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

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

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

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

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

*Corresponding author: Mengchi Cai (cmc18@mails.tsinghua.edu.cn)

Fig. 1. Examples of tracking results on UA-DETRAC dataset. Eight different scenarios are shown here, and each scene shows three frames of images. The bounding boxes with different colors represent different vehicle IDs, and the black areas indicate areas that are ignored, as defined in UA-DETRAC.

mance with 74.04% average precision (AP) and 36 FPS in detection, 19.10% PR-MOTA and 34 FPS in tracking. Figure 1 shows some examples of tracking results. To summarize, the main contributions of this paper are as follows:

- 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.
- 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.
- 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.

The following of this paper is organized as follows. Section II introduces the existing research that is related to this paper. Section III provides the methodology of this paper, including network components and implementation details. Section IV conducts experiments and section V gives the conclusion.

II. RELATED WORKS

A. 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]–[12] and anchor-free [13]–[15]. Anchor-based algorithms set a series of anchor boxes and regress offsets between the anchor boxes and ground truth using local features. 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. DETR [15] 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.

B. Tracking-by-detection

With the improvement in detection accuracy, tracking-by-detection methods [16]–[22] 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, shape, and appearance, are used to describe the correlation between detections and tracks, and thus, establish a correlation matrix. 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.

Fig. 2. Detection module architecture. The Conv block denotes a convolution layer, *BN* denotes batch normalization, *FC* is a full-connected layer. This module inputs image $I \in R^{h \times w \times 3}$, and outputs a center detection confidence map $Y_d \in R^{h' \times w' \times n}$. n is the number of categories and several bounding boxes $B_d \in R^{h' \times w' \times 4}$. During training, it produces all potential bounding boxes in a tensor $B_d \in R^{h' \times w' \times 4}$. Conclusively, during the test, it is only required to calculate the bounding boxes at the peaks of the confidence map.

C. Joint detection and tracking

In the field of MOT, it is an important research direction to combine detection and tracking. [25]–[28] begins with multi-task learning, and adds ReID feature extraction to existing object detection networks. [29] adopts Faster-RCNN framework, and accomplishes tracking by RoI pooling and bounding box regression without data association. [30] considers 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. [31] converts the MOT problem into a pair-wise object detection problem, and realizes end-to-end joint object detection and tracking. [30] and [31] are two innovative approaches to joint detection and tracking. Similarly, this study also provides a new idea for joint detection and tracking.

III. METHODOLOGY

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_{ij} are the bounding box of object i at time j . Considering an image of the current frame $I_t \in R^{h \times w \times 3}$, the network assigns the bounding boxes $B_{x,t}$ of objects in the current frame to historical trajectories, or generated new trajectories. The following section introduces the proposed algorithm in detail.

A. 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. For 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 W , the correlation vectors $C_{ij} \in R^{c'}$ is obtained.

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

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.

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 confidences of the object centers in the current frame, while the tracking heatmap Y_t denotes the

Fig. 3. Illustration of gate step. This step is similar to the sSE block in [32]. The difference between the two is that the inputs of the gate step are two tensors. Tensor X contains the values, and tensor Y denotes spatial attention. In the proposed detection module, the confidence map from the classification branch can be directly regarded as spatial attention.

tracking confidence between the current frame 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).

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

where $\maxpool(H, a, b)$ 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 an 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 .

$$\begin{aligned} Q &= BN_Q(Conv_Q(F, 1, 1, c) + P) \\ K &= Gate[BN_K(Conv_K(F, 1, 1, c) + P), Y_d] \\ V &= Conv_V(F, 1, 1, c) \end{aligned} \quad (4)$$

where $Conv(F, a, b)$ denotes a convolution layer with kernel size a , strides b , and kernel number c , and BN denotes batch normalization layer. $Gate(X, Y)$ 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

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

The two embedding vectors that are close in 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 is calculated using equation (2). The final bounding boxes $B_{d,ij} = [x_{ij}, y_{ij}, h_{ij}, w_{ij}]$ can be obtained using equation (6). Here, the absolute coordinates and size of the bounding box are directly regressed, which differs from most existing methods, especially anchor-based methods.

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

Tracking module: Tracking is the process of assigning objects in the current frame to historical tracks, or of generating new tracks. The architecture of the tracking module is depicted in Figure 4. The inputs of the tracking module are the 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. As 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.

Fig. 4. Tracking module architecture. This architecture is similar to the regression branch in the detection module. The green portion indicates that it does not share parameters with the detection module. At the time of training, the module outputs all potential tracking bounding boxes $B_t \in R^{h' \times w' \times 4}$ and a tracking confidence map $Y_t \in R^{h' \times w' \times 1}$ between frames $t-i$ and t ; here, t lies between 1 and 5. Hence, only tracking the bounding boxes and confidence of the historical tracks are calculated during testing.

B. 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:

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

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

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 equation (8), while 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 $\sum_n G_{ijn} - \max_n G_{ijn} < 0.3$.

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

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 CIOU loss and classification focal 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

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

C. 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 $I_1 \sim I_t$. Trajectories T_i , confidence Y_i , and vectors $[V_i, Q_i]$ 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 tracks $\mathcal{T}$ and candidates $\mathcal{C}$. 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.

Algorithm 1: Inference pipeline of GCNet

Input: continuous frame images $I_1 \sim I_t$.
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 vector set  $\mathcal{O} = \emptyset$ ; candidate set  $\mathcal{C} = \emptyset$ ; and hyperparameters  $p_1, p_2$ , and  $p_3$ .
2 for  $I$  in  $I_2 \sim I_t$  do
3    $Q, K, V, B_d = \text{DetectionModule}(I)$ ;
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  $\exists j, IoU(B_i, T_j) > p_3$  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

```

IV. EXPERIMENTS

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.

A. 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 [33] for further details on the metrics.

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.

B. 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 I 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 the role of spatial attention and is conducive to the training of the regression branch. The concatenated feature vectors of V for regression introduces 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.

TABLE I
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

The results of the ablation study can thus be extended to the tracking module.

C. Benchmark evaluation

Table II shows the results obtained using the UA-DETRAC detection benchmark. The GCNet demonstrates promising performance, and outperforms 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. When only the detection module of the GCNet is used, it can run at 36 FPS 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 III. The MOT metrics with “PR-” can evaluate the overall effect of detection and tracking. EB and KIOU are the UA-DETRAC challenge winners. 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 FPS.

V. CONCLUSION

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 evaluation results demonstrate the effectiveness of the proposed approach in both detection and tracking. This approach is also equipped to run 36 FPS for detection and 34 FPS 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.

VI. DECLARATION

Acknowledgements

Not applicable

Funding

Supported by the National Key Research and Development Program of China 2021YFB1600402, the National Natural Science Foundation of China 52072212, China Intelligent and Connected Vehicles (Beijing) Research Institute Co., Ltd., and Dongfeng USharing Technology Co., Ltd.

Availability of data and materials

The datasets supporting the conclusions of this article are included within the article.

Authors' contributions

The author's contributions are as follows: Qing Xu, Kai Li, Keqiang Li, Jianqiang Wang, and Dongpu Cao are in charge of the whole trial; Xuewu Lin and Mengchi Cai write the manuscript; Yu-ang Guo and Chuang Zhang assist with experiments and analysis.

Competing interests

The authors declare no competing financial interests.

Consent for publication

Not applicable

Ethics approval and consent to participate

Not applicable

REFERENCES

- [1] Y. Liu, X. Guan, P. Lu, and R. Guo, “Research on key issues of consistency analysis of vehicle steering characteristics,” *Chinese Journal of Mechanical Engineering*, vol. 34, no. 1, pp. 1–12, 2021.
- [2] Q. Xu, M. Cai, K. Li, B. Xu, J. Wang, and X. Wu, “Coordinated formation control for intelligent and connected vehicles in multiple traffic scenarios,” *IET Intelligent Transport Systems*, 2021.
- [3] Y. Luo, D. Yang, M. Li, K. Li, and X. Lian, “Hardware-in-the-loop simulation on dynamical coordinated control method in parallel hybrid electric vehicle (phev),” *Chinese Journal of Mechanical Engineering*, vol. 5, 2008.
- [4] M. Cai, C. Chen, J. Wang, Q. Xu, K. Li, J. Wang, and X. Wu, “Formation control with lane preference for connected and automated vehicles in multi-lane scenarios,” *arXiv preprint arXiv:2106.11763*, 2021.
- [5] C. Chen, M. Cai, J. Wang, K. Li, Q. Xu, J. Wang, and K. Li, “Cooperation method of connected and automated vehicles at unsignalized intersections: Lane changing and arrival scheduling,” *arXiv preprint arXiv:2109.14175*, 2021.
- [6] M. Cai, Q. Xu, C. Chen, J. Wang, K. Li, J. Wang, and Q. Zhu, “Formation control for connected and automated vehicles on multi-lane roads: Relative motion planning and conflict resolution,” *arXiv preprint arXiv:2103.10287*, 2021.
- [7] S. Ren, K. He, R. Girshick, and J. Sun, “Faster r-CNN: Towards real-time object detection with region proposal networks,” vol. 39, no. 6, pp. 1137–1149.
- [8] T. Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in *2017 IEEE International Conference on Computer Vision (ICCV)*.
- [9] R. Girshick, “Fast r-CNN,” in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*.
- [10] J. Redmon and A. Farhadi, “YOLOv3: An incremental improvement.” [Online]. Available: <http://arxiv.org/abs/1804.02767>
- [11] C.-Y. Fu, W. Liu, A. Ranga, A. Tyagi, and A. C. Berg, “DSSD : Deconvolutional single shot detector.” [Online]. Available: <http://arxiv.org/abs/1701.06659>
- [12] He K , Gkioxari G , Piotr Dollár, et al., “Mask r-CNN.”
- [13] X. Zhou, D. Wang, and P. Krhenbühl, “Objects as points.”
- [14] H. Law and J. Deng, “CornerNet: Detecting objects as paired keypoints,” in *European Conference on Computer Vision*.

Fig. 5. Precision vs. recall curves of the detection algorithms. The scores in the legend are the AP scores for evaluating the performance of the object detection algorithms.

TABLE II
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

TABLE III
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

- [15] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers." [Online]. Available: <http://arxiv.org/abs/2005.12872>
- [16] S. H. Rezatofighi, A. Milan, Z. Zhang, Q. Shi, A. Dick, and I. Reid, "Joint probabilistic data association revisited," in *2015 IEEE International Conference on Computer Vision (ICCV)*. IEEE, pp. 3047–3055. [Online]. Available: <http://ieeexplore.ieee.org/document/7410706/>
- [17] A. Bewley, Z. Ge, L. Ott, F. Ramos, and B. Upcroft, "Simple online and realtime tracking," pp. 3464–3468. [Online]. Available: <http://arxiv.org/abs/1602.00763>
- [18] N. Wojke, A. Bewley, and D. Paulus, "Simple online and realtime tracking with a deep association metric," pp. 3645–3649.
- [19] E. Bochinski, V. Eiselein, and T. Sikora, "High-speed tracking-by-detection without using image information [challenge winner IWOT4s]," in *International Workshop on Traffic and Street Surveillance for Safety and Security at IEEE AVSS 2017*.
- [20] M. Ullah, F. A. Cheikh, and A. S. Imran, "HoG based real-time multi-target tracking in bayesian framework," in *IEEE International Conference on Advanced Video & Signal Based Surveillance*.
- [21] E. Ristani and C. Tomasi, "Features for multi-target multi-camera tracking and re-identification."
- [22] X. Shi, H. Ling, Y. Pang, W. Hu, P. Chu, and J. Xing, "Rank-1 tensor approximation for high-order association in multi-target tracking," vol. 127, no. 8, pp. 1063–1083. [Online]. Available: <http://link.springer.com/10.1007/s11263-018-01147-z>
- [23] A. Sadeghian, A. Alahi, and S. Savarese, "Tracking the untrackable: Learning to track multiple cues with long-term dependencies," in *2017 IEEE International Conference on Computer Vision (ICCV)*. IEEE, pp. 300–311. [Online]. Available: <http://ieeexplore.ieee.org/document/8237303/>
- [24] C. Kim, F. Li, A. Ciptadi, and J. M. Rehg, "Multiple hypothesis tracking revisited," p. 9.
- [25] Y. Zhang, C. Wang, X. Wang, W. Zeng, and W. Liu, "FairMOT: On the fairness of detection and re-identification in multiple object tracking." [Online]. Available: <http://arxiv.org/abs/2004.01888>
- [26] Z. Lu, V. Rathod, R. Votel, and J. Huang, "RetinaTrack: Online single stage joint detection and tracking," in *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [27] P. Voigtlaender, M. Krause, A. Osep, J. Luiten, and B. Leibe, "MOTS: Multi-object tracking and segmentation," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.
- [28] Z. Wang, L. Zheng, Y. Liu, Y. Li, and S. Wang, "Towards real-time multi-object tracking," in *Computer Vision – ECCV 2020*, A. Vedaldi, H. Bischof, T. Brox, and J.-M. Frahm, Eds. Springer International Publishing, pp. 107–122.
- [29] P. Bergmann, T. Meinhardt, and L. Leal-Taixe, "Tracking without bells and whistles," in *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*. IEEE, pp. 941–951. [Online]. Available: <https://ieeexplore.ieee.org/document/9010033/>
- [30] X. Zhou, V. Koltun, and P. Krähenbühl, "Tracking objects as points." [Online]. Available: <http://arxiv.org/abs/2004.01177>
- [31] J. Peng, C. Wang, F. Wan, Y. Wu, and Y. Fu, "Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking."
- [32] A. G. Roy, N. Navab, and C. Wachinger, "Concurrent spatial and channel squeeze & excitation in fully convolutional networks." [Online]. Available: <http://arxiv.org/abs/1803.02579>
- [33] L. Wen, D. Du, Z. Cai, Z. Lei, M. C. Chang, H. Qi, J. Lim, M. H. Yang, and S. Lyu, "UA-DETRAC: A new benchmark and protocol for multi-object detection and tracking."

BIOGRAPHICAL NOTES

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

Xuewu Lin received the B.E. degree from Beijing University of Technology, Beijing, China, in 2018, and the M.S. from Tsinghua University, Beijing, China, in 2021. He is currently an Engineer with the Horizon Information Technology Co.,

Ltd. His research interests include object detection, multi-object tracking and trajectory prediction.

Mengchi Cai received the B.E. degree from Tsinghua University, Beijing, China, in 2018. He is currently a Ph.D. candidate in mechanical engineering with the School of Vehicle and Mobility, Tsinghua University. 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, Beijing, China, in 2014. He received the M.S. degree in software engineering from Northwest University, Shaanxi, China, in 2018. He is currently a Ph.D. candidate in the School of Transportation Science and Engineering, Beihang University, Beijing, China. His current research interests include road detection, image segmentation and multi-sensor fusion.

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

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

Keqiang Li received the B.Tech. degree from Tsinghua University of China, Beijing, China, in 1985, and the M.S. and Ph.D. degrees in mechanical engineering from the Chongqing University of China, Chongqing, China, in 1988 and 1995, respectively. He is currently a Professor with the School of Vehicle and Mobility, Tsinghua University. His main research areas include automotive control system, driver assistance system, and networked dynamics and control. He is leading the national key project on CAVs (Intelligent and Connected Vehicles) in China. Dr. Li has authored more than 200 papers and is a co-inventor of over 80 patents in China and Japan.

Dr. Li has served as Fellow Member of Society of Automotive Engineers of China, editorial boards of the *International Journal of Vehicle Autonomous Systems*, Chairperson of Expert Committee of the China Industrial Technology Innovation Strategic Alliance for CAVs (CACAV), and CTO of China CAV Research Institute Company Ltd. (CCAV). He has been a recipient of Changjiang Scholar Program Professor, National Award for Technological Invention in China, etc.

Jianqiang Wang received the B. Tech. and M.S. degrees from Jilin University of Technology, Changchun, China, in 1994 and 1997, respectively, and the Ph.D. degree from Jilin University, Changchun, in 2002. He is currently a Professor with the School of Vehicle and Mobility, Tsinghua University, Beijing, China. He has authored over 150 papers and is a co-inventor of 99 patent applications. He was involved in over 10 sponsored projects. His active research interests include intelligent vehicles, driving assistance systems, and driver behavior. He was a recipient of the Best Paper Award in the 2014 IEEE Intelligent Vehicle Symposium, the Best Paper Award in the 14th ITS Asia Pacific Forum, the Best Paper Award in the 2017 IEEE Intelligent Vehicle Symposium, the Changjiang Scholar Program Professor in 2017, the Distinguished Young Scientists of NSF China in 2016, and the New Century Excellent Talents in 2008.

Dongpu Cao received the Ph.D. degree from Concordia University, Canada, in 2008. He is the Canada Research Chair in Driver Cognition and Automated Driving, and an Associate Professor at University of Waterloo. His current research focuses on driver cognition, automated driving and cognitive autonomous driving. He has contributed more than 200 papers and 3 books. He received the SAE Arch T. Colwell Merit Award in 2012, IEEE VTS 2020 Best Vehicular Electronics Paper Award, 2021 IEEE ITSS Outstanding Application Award, and 7 Best Paper Awards from international conferences. Prof. Cao has served as Deputy Editor-in-Chief for IET INTELLIGENT TRANSPORT SYSTEMS JOURNAL, and an Associate Editor for IEEE TRANSACTIONS ON VEHICULAR TECHNOLOGY, IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS and other journals. Prof. Cao is an IEEE VTS Distinguished Lecturer.