Multiple Object Tracking with Correlation Learning

Qiang Wang, Yun Zheng, Pan Pan, Yinghui Xu

Machine Intelligence Technology Lab, Alibaba Group

{qishi.wq, zhengyun.zy, panpan.pp, renji.xyh}@@alibaba-inc.com

Abstract

Recent works have shown that convolutional networks have substantially improved the performance of multiple object tracking by simultaneously learning detection and appearance features. However, due to the local perception of the convolutional network structure itself, the long-range dependencies in both the spatial and temporal cannot be obtained efficiently. To incorporate the spatial layout, we propose to exploit the local correlation module to model the topological relationship between targets and their surrounding environment, which can enhance the discriminative power of our model in crowded scenes. Specifically, we establish dense correspondences of each spatial location and its context, and explicitly constrain the correlation volumes through self-supervised learning. To exploit the temporal context, existing approaches generally utilize two or more adjacent frames to construct an enhanced feature representation, but the dynamic motion scene is inherently difficult to depict via CNNs. Instead, our paper proposes a learnable correlation operator to establish frame-to-frame matches over convolutional feature maps in the different layers to align and propagate temporal context. With extensive experimental results on the MOT datasets, our approach demonstrates the effectiveness of correlation learning with the superior performance and obtains state-of-the-art MOTA of 76.5% and IDF1 of 73.6% on MOT17.

1. Introduction

Multi-Object Tracking (MOT) is an essential component for computer vision with many applications, such as video surveillance [31] and modern autonomous driving [19, 41]. It aims to continuously locate trajectories of multiple targets in video frames. Decades of research efforts have led to impressive performance on challenging benchmarks [24, 30, 8].

MOT has traditionally adopted the tracking-by-detection paradigm [3, 5, 1, 58], which capitalizes on the natural division of detection and data association tasks for the problem. These algorithms extract appearance features within
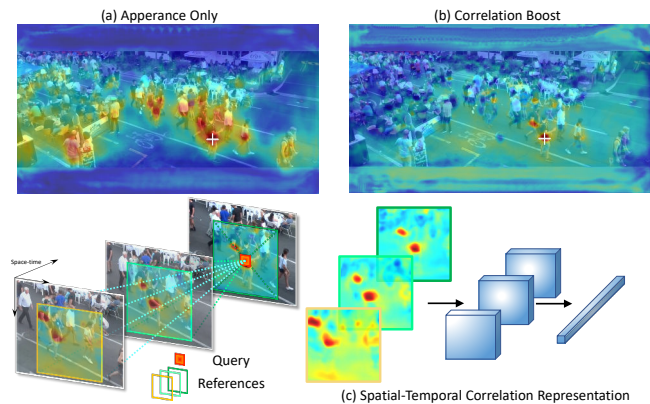

Figure 1. Visualization of the matching confidences (a)-(b) computed between the indicated target (white cross) in the reference image and all locations of the query image. The appearance feature based tracker [58] (a) generates undistinctive and inaccurate confidences due to the existing of similar distractors. In contrast, our Correlation Tracker (b) predicts a distinct high-confidence value at the correct location, with correlation learning (c).

each detection patches and record object location information for subsequent data association [51, 5]. This tracking paradigm makes researchers mainly focus on optimizing detection [56, 16], feature representation [27, 15], or data association [3, 5, 17]. With the rapid progress of detection algorithms [13, 14, 36, 35], the detection based tracking has achieved great performance improvement [56, 16]. Although tremendous strides have been made in MOT, there still exists tough challenges in determining distractors and frequent occlusions, especially in complex interactive scenes [8]. Additionally, the above cascaded structure is inefficient and prevents the joint optimization between stages.

One promising approach is to extend the end-to-end trainable detection framework [36, 35, 64] to jointly learn detection and appearance feature, which has largely advanced the state-of-the-art in MOT [44, 50, 29, 58]. However as illustrated in Fig. 1, in the case of existing similar distractors, the appearance feature generates undistinctive and inaccurate matching confidences (Fig. 1a), severely affecting the performance of association. These methods are limited in local descriptors, and it is difficult to distinguish

similar objects. While as shown in Fig. 1c, the context relation map can help to easily distinguish different targets.

Based on those observations, we propose a correlation network to learn the topological information of the object and context. Specifically, we use a spatial correlation layer to record the relationship between targets and relative spatial positions. While constructing a full correlation (*e.g.*, non-local [48]) for all locations is computationally prohibitive for real-time MOT, this work constructs a local correlation volume by limiting the search range at each feature pyramid. Besides, our correlation learning is not limited for targets of interest category [53, 49]. Background contexts, such as vehicles, are also modeled to help target recognition and relational reasoning (Fig. 1c). We establish dense correspondences of each spatial location and its context, and explicitly constrain the correlation volumes through self-supervised learning.

Further, the detector in MOT usually uses independent frames as input and therefore does not make full use of temporal information. This detection method makes the algorithm suffer from missing detection in crowded scenes, and further increases the difficulty of subsequent data association. Recently, adjacent frames [64, 34] or three frames [32] are adopted to enhance the temporal consistence. The performance of the algorithm in occlusion scenes has been improved to a certain extent, but these methods are still limited with fewer frames. CenterTrack [64] attempt to use an aggressive data augmentation to increase the ability of target alignment, but convolution networks itself are inherently limited in local receptive fields. To solve the above problem, we extend the spatial correlation module to the temporal dimension and incorporate the historical information to reduce ambiguities in object detections.

To summarize, we make the following contributions:

- We propose CorrTracker, a unified correlation tracker to intensively model associations between objects and transmit information through associations.
- We propose a local structure-aware network and enhance the discriminability of similar objects with self-supervised learning.
- We extend the local correlation network to model temporal information efficiently.
- CorrTracker shows significant improvements over existing state-of-the-art results in four MOT benchmarks. In particular, we achieve 76.5% MOTA and IDF1 of 73.6% on MOT17.

2. Related Work

Real-time Tracking. As MOT has strong practical merit, the tracking speed attracts much attention. The researchers start from the simplest IOUTracker [4], which only uses the intersection-over-union of bounding boxes for tracking,

to add the motion model of Kalman Filter [3] to predict the position of the rectangular boxes for matching. Although they have achieved amazing speed, stable tracking cannot be achieved under challenges such as target interleaving. Researchers [27, 51] introduce Person Re-Identification (ReID) features as an appearance model to increase the discriminative power of the tracker. However, the individual calculation for patches makes the object classification and ReID feature extraction as a computational bottleneck. MOTDT [27] achieves real-time tracking by using RoI-pooling [14] on a shared feature map. In order to further decrease the computational cost of ReID feature extraction, JDE [50] adds a ReID branch in a single-stage detector YOLOv3 [35] to achieve efficient ReID feature calculation. FairMOT [58] explores the importance of detection and recognition tasks and uses anchor-free method [65] to reduce the ambiguity of anchors. We are mainly based on FairMOT, which achieves the state-of-the-art performance with a more balanced ReID and detection.

Other researchers [1, 64, 34, 32] explore new tracking paradigms to remove ReID recognition. Tracktor [1] uses the the bounding boxes in the previous frame to directly regress the current locations. CenterTrack [64], Chained-Tracker [34], and TubeTK [32] use multiple frames to simultaneously predict the bounding boxes for adjacent frames to achieve short association, thereby merging to long-term tracks. However, these methods usually have many identity switches because they cannot model long-term dependencies.

Tracking with Graph Model. MOT has traditionally been approached as a hand-crafted graph optimization problem [61, 17], where the cropped targets are treated as nodes. Recently, graph neural network based methods [53, 49, 5] have been shown as a promising alternative to traditional optimization methods. State-of-the-art approach [53] utilizes graph convolutional network to propagate features in the spatial-temporal space. MPN [5] introduces the message passing network to dissect the information and associate detections through the edge classification. Different from these methods, feature propagation is carried out at the frame feature level, which can absorb the information of both the foreground and background and reduce the loss of contextual information.

Tracking with Optical Flow. FlowTrack [60] introduces optical flow to predict the target location. But explicitly using optical flow is not only time-consuming, but also only encodes the pixel-level motion. CenterTrack [64] borrows the method of optical flow to directly predict the movement of the target center between two frames, which is called *instance flow*. However, directly predicting the offset on the concatenated feature map needs to provide training samples with all displacement, which requires excessive data augmentation. Our correlation method predicts a dense set of
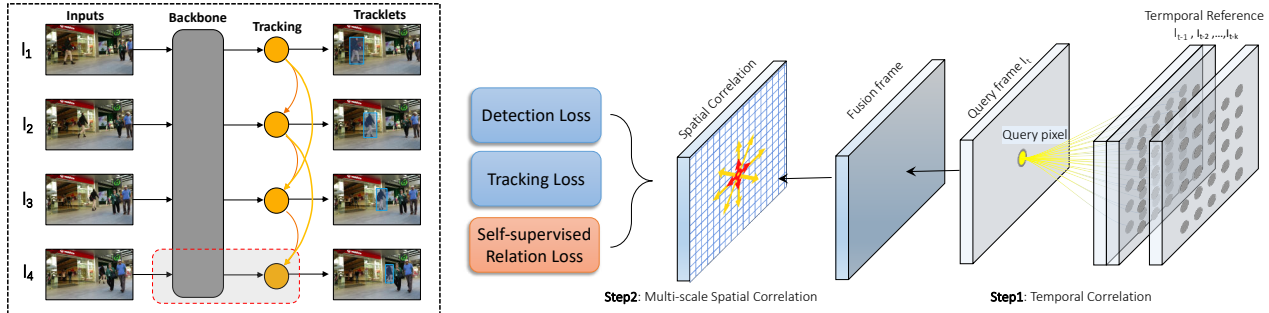
Figure 2. **Overview**. We enhance the appearance features with correlation layer, which densely encoding pairwise relation of object and their spatial-temporal neighbourhoods. The local correlation volumes are optimized in a self-supervised manner.

matching confidence for each target, which is intrinsically invariant to translation of the paired frames. Our correlation operation is similar to the correlation volume in optical flow [10, 40, 42] and correlation filter [7]. We both predict dense local correlation, and regard it as a part of the feature description. However, optical flow does not calculate the internal correlation of the image, nor does it have propagated the feature from multiple frames. D&T [12] also utilizes correlation layer to predict candidate motion between pair of consecutive frames. Compared with it, our anchor-free framework is more compact and efficient.

Attention Mechanism. Our modeling of local correlation is similar to the self-attention mechanism and Transformer. Transformer has been a huge success in the NLP field [43] and has also been adapted to the computer vision [48, 18, 28] to capture long-range dependencies. In order to reduce the quadratic complexity of the non-local operation, the researchers propose to shrink the attention span with local region [33], or only along individual axes [47]. Different from these methods, we mainly encode the context identity through local correlation weighting, and use this cues to increase the model robustness.

3. Methodology

Figure 2 shows the overall pipeline of the proposed CorrTracker. Our method can be distilled down to three stages: (1) general feature extraction, (2) simultaneous learning correlation from spatial-temporal dependencies and predicting the detection, and (3) performing data association to assign detections into their most likely trajectories, where stage (1) and stage (2) are differentiable and composed into an end-to-end trainable architecture. We adopt a compact association technique that is similar to the one used by DeepSORT [51] to control the initialization and the termination of tracks. The main contribution is the highly efficient modeling for the correlation between dense location and their context on feature maps, which helps suppressing distractors in complex scenes.

3.1. Motivation

For each input video frame $\mathbf{I}_t \in \mathbb{R}^{H \times W \times 3}$, an object detector is applied to find all candidate detections $\mathcal{D}_t = \{\mathbf{d}_t^i\}_{i=1}^N$, $\mathbf{d}_t^i = (x_t^i, y_t^i, w_t^i, h_t^i)$ appearing in this frame and we have existing trajectories $\mathcal{T}_{t-1} = \{\mathbf{T}_{t-1}^j\}_{j=1}^M$, $\mathbf{T}_{t-1}^j = \{\mathbf{d}_1^j, ..., \mathbf{d}_{t-2}^j, \mathbf{d}_{t-1}^j\}$. Then the affinity matrix $\mathbf{A} \in \mathbb{R}^{N \times M}$ is estimated by pair-wise comparisons of cropped patches and existing trajectories. The metric jointly considers both the appearance features $\mathbf{f}(\cdot) \in \mathbb{R}^d$ and geometric representations.

$$\mathbf{A}_{ij} = \mathrm{dist}(\mathbf{f}(\mathbf{d}_t^i), \hat{\mathbf{f}}(\mathbf{T}_{t-1}^j)) + \alpha \mathrm{IoU}(\mathbf{d}_t^i, \hat{\mathbf{d}}_t^j), \quad (1)$$

The discriminative feature $\hat{\mathbf{f}}(\mathbf{T}_{t-1}^j)$ of a trajectory is usually updated with a constant-weighting strategy to follow the appearance changes. Each confidence value for appearance feature is obtained in a distance metric, *e.g.*, the inner product space. However, the sole reliance on person-to-person feature comparisons are often insufficient to disambiguate multiple similar regions in an image. As illustrated in Fig. 1, in the case of similar distractors, the feature extractor usually generates inaccurate and uninformative matching confidences (Fig. 1a), severely affecting the performance of data association. This is the key limitation of appearance feature matching, since co-occurring similar objects are all pervasive in MOT.

Patch based feature extraction is applied as a prevalent scheme in MOT owing to its intuition. However the correlation information between the cropped image patches is lost directly, and the adjacency spatial relationship is only retained in coordinates $\mathbf{d}_t^i$. Although the subsequent data association will be globally optimized, directly using ReID features without considering the context tends to introduce more identity switches, lagging the tracking robustness. To deal with this problem, we model the local structure of objects to distinguish it from distractors.

Inspired by correlation volume from optical flow [10], we observe that a confidence value in the correlation vol-

ume models the geometric structure of each target. We design a novel dense correlation module, aiming to explore the context information for MOT. The relative position is encoded in the correlation volumes, which can be used as an auxiliary discriminant information.

3.2. Spatial Local Correlation Layers

In this work, we use Spatial Local Correlation Layers to model the relational structure for associating a target with its neighbour. In our local correlation layer, the feature similarity is only evaluated in the neighbourhood of the target image coordinate. Formally, we let l denote the level in the feature pyramid and the correlation volume $\mathbf{C}^l$ between the query feature $\mathbf{F}_q^l \in \mathbb{R}^{H_l \times W_l \times d_l}$ and reference feature $\mathbf{F}_r^l \in \mathbb{R}^{H_l \times W_l \times d_l}$ is defined as,

$$\mathbf{C}^l(\mathbf{F}_q, \mathbf{F}_r, \mathbf{x}, \mathbf{d}) = \mathbf{F}_q^l(\mathbf{x})^T \mathbf{F}_r^l(\mathbf{x} + \mathbf{d}), \|\mathbf{d}\|_\infty \le R, \quad (2)$$

where $\mathbf{x} \in \mathbb{Z}^2$ is a coordinate in the query feature map and $\mathbf{d} \in \mathbb{Z}^2$ is the displacement from this location. The displacement is constrained to $\|\mathbf{d}\|_\infty \le R$, *i.e.* the maximum motion in any direction is R. While most naturally thought of as a 4-dimensional tensor, the two displacement dimensions are usually vectorized into one to simplify further processing in the CNN. The resulting 3-d correlation volume $\mathbf{C}^l$ thus is of size $H^l \times W^l \times (2R+1)^2$. We also introduce the dilation tricks [55], which can increase the receptive field without additional cost. We use element-wise addition to incorporate the correlation feature into a unified appearance representation. This context correlation features are encoded by a feed-forward Multilayer Perceptron (MLP) to match the number of channels d_l in appearance features $\mathbf{F}_t^l$.

$$\mathbf{F}_\mathbf{C}^l = \mathbf{F}_t^l + \mathrm{MLP}^l\left(\mathbf{C}^l\left(\mathbf{F}_t^l, \mathbf{F}_t^l\right)\right). \quad (3)$$

The non-local [48] module is to explicitly model all pairwise interactions between elements in a feature maps $\mathbf{F}_t^l \in \mathbb{R}^{H^l \times W^l \times d^l}$. The resulting four-dimensional correlation volume $NL(\mathbf{F}_t^l) \in \mathbb{R}^{H^l \times W^l \times H^l \times W^l}$ captures dense matching confidences between every pair of image locations. They build a full connection volume at a single scale, which is both computationally expensive and memory intensive. By contrast, our work shows that constructing a local correlation volume leads to both effective and efficient models. In comparison with the global correlation method, our local correlation model adds less overhead to the latency (see Table 1).

3.3. Correlation at Multiple Pyramid Levels

In order to achieve long-range correlation, we propose to learn correlation at feature pyramids, as shown in Figure 3. On the one hand, we hope that our correlation module can obtain long-distance dependencies as much as possible, but as the local region size R increases, both calculation
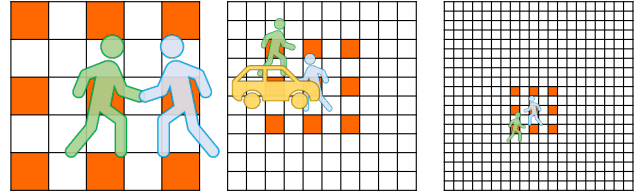

Figure 3. Correlation at Multiple Pyramid Levels. For a feature tensor in $\mathbf{F}^l$, we take the inner product with local region ($R = 1, D = 2$) to generate a 3-d correlation volume $W^l \times H^l \times (2R+1)^2$, D is the dilation rate.

and storage increase significantly, which hinders the application. On the other hand, MOT naturally needs to deal with multi-scale targets. The two-stage detection [36] uses RoI pooling [14] to eliminate the difference in target scales, but this type of method usually suffers from high processing latency. In order to solve the above problems, we utilize the general pyramid structure in the convolutional network and learn correlation on the feature pyramids. Our multi-scale pyramid correlation can also be regarded as a comparison of multi-granularity features, covering the spatial context in the range of $[0, R \times D \times 2^l]$, where D refers to the dilation rate. And, we pass this correlation from the top layer to the bottom layer,

$$\hat{\mathbf{F}}_\mathbf{C}^{l-1} = \mathbf{Conv}(\mathbf{Upsample}(\mathbf{F}_\mathbf{C}^l)) + \mathbf{F}_\mathbf{C}^{l-1}, \quad (4)$$

In this way, we can obtain an approximate correlation between the target and the entire global context, while keep the compactness and efficiency. Our pyramid correlation leverages the natural spatial-temporal coherence in videos. Multi-object tracking can be decomposed into multiple independent single-object tracking. Our method can be equivalent to a dense siamese network tracking [2] on the feature pyramid. On the other hand, from the perspective of set matching, global characteristics need to be considered. Our multi-scale correlation takes into account both aspects of information transmission.

3.4. Temporal Correlation Learning

The correlation between different frames are usually ignored by the MOT field, and trackers usually overcome occlusion through data association. Single frame detector is difficult to ensure a good temporal consistency [59]. This makes the algorithm's performance drop significantly in occlusion, motion blur and small target scenes, which becomes the bottleneck for MOT. We extend the spatial local correlation from Section 3.2 to the temporal dimension, and establish correlation for the targets in different frames. The correlation between two frames can be viewed as the establishment of motion information learning. We also use this correlation to enhance the feature representation, which can increase the detection accuracy.

Specifically, we establish multi-scale correlation between different frames, and use reference images as memory to enhance image features. This method helps tracker overcome target occlusion and motion blur, and increases the consistency of detection and identity features.

$$\hat{\mathbf{F}}_q(\mathbf{x}) = \sum_{\forall \|\mathbf{d}\|_\infty < R} \frac{\mathbf{C}^l(\mathbf{F}_q, \mathbf{F}_r, \mathbf{x}, \mathbf{d})}{(2R+1)^2} \mathbf{F}_r(\mathbf{x} + \mathbf{d}) \quad (5)$$

$$\mathbf{C}^l(\mathbf{F}_q, \mathbf{F}_r, \mathbf{x}, \mathbf{d}) = \mathbf{F}_q^l(\mathbf{x})^T \mathbf{F}_r^l(\mathbf{x} + \mathbf{d}), \|\mathbf{d}\|_\infty \le R \quad (6)$$

Similar to the multi-head attention [43], we adopt the embedded features and dot-product similarity. In our case, we set the normalization factor as $(2R+1)^2$ and locally aggregate features. This shrinked region design also comes from the motion prior of the MOT scene. For the minimal memory consumption and fastest run-time, we can only save the previous features $\mathbf{F}_{t-1}$ in the memory. For the maximum accuracy, our long-term model saves the latest 5 frame image features by default.

3.5. Self-supervised Feature Learning

In Section 3.2 and Section 3.4, we present how we model the correlation in spatial and temporal dimension. We can simply use the proposed correlation module as a plugin module without explicitly adding constraints, similar to the non-local module, which has shown significant improvement. Here we investigate a multi-task learning approach that imposes a semantic supervision from visual object tracking [2] and self-supervised training from correspondence flow [45] on correlation volumes.

Our correlation module is interpretable, measuring the similarity between different objects. Actually, our method intensively performs $M \times N$ siamese tracking operations [2] to increases the discrimination. In this view, we can explicitly impose a tracking supervision. Specifically, we set up the ground-truth label as

$$\tilde{C}^l(\mathbf{F}_q, \mathbf{F}_r, \mathbf{x}, \mathbf{d}) = \begin{cases} 1 \text{ if } \mathbf{y}_q(\mathbf{x}) = \mathbf{y}_r(\mathbf{x} + \mathbf{d}) \\ 0 \text{ if } \mathbf{y}_q(\mathbf{x})! = \mathbf{y}_r(\mathbf{x} + \mathbf{d}), \\ -1 \text{ if } \mathbf{y}_q(\mathbf{x}) < 0 \end{cases} \quad (7)$$

where $\mathbf{y}$ is the identity label of the corresponding position in feature maps. We ignore the position without objects $\mathbf{y}_q(\mathbf{x}) < 0$ and use a class-balanced cross-entropy loss [2]. Inspired by the recent advances of self-supervised tracking [45], we use *colorization* as a proxy task for training our local correlation.

$$\hat{\mathbf{I}}_q(\mathbf{x}) = \sum_{\forall \|\mathbf{d}\|_\infty < R} \frac{\mathbf{C}^l(\mathbf{F}_q, \mathbf{F}_r, \mathbf{x}, \mathbf{d})}{(2R+1)^2} \mathbf{I}_r(\mathbf{x} + \mathbf{d}), \quad (8)$$

we use the cross-entropy categorical loss after quantizing the color-space into discrete categories [45].

3.6. Tracking Framework

We modify the FairMOT [58] backbone by adding correlation module before the iterative deep aggregation module [57]. Our model retains the detection and ReID branches, and adding correlation loss in Sec. 3.5 for multi-task learning. For the tracking inference, our tracker first calculates the similarity between the detections of the current frame and the previous trajectories according to Eq. (2), and use the Hungarian algorithm [23] for finding the optimal matching. The unmatched detections are used to initialize new trajectories. In order to reduce false positives, we mark these new trajectories as "*inactive*" until the next frame is matched again and confirmed as "*active*". The unmatched trajectories are set to the "lost" state. When the continuous lost time t_{loss} of a trajectory exceeds the threshold τ_{loss}, we put it in the remove set. If there is a success matching before removing, we restore the trajectory to the active state. We use Kalman Filter [20] to model the pedestrian motion and keep the same settings in FairMOT [58].

4. Experiments

To demonstrate the advantages of the proposed correlation tracker, we first compare the correlation module with other relational reasoning methods [48, 46] and evaluate different settings to justify our design choices in Section 4.2. Then we show that our correlation tracker outperforms the state-of-the-art methods on four MOT benchmarks [24, 30, 8] in Section 4.3. Finally, we visualize the tracking trajectories in Section 4.4 and compare with other motion prediction based trackers [64, 34, 32].

4.1. Implementation Details

Network Setup. The implementation and hyperparameters mostly follow [58], we adopt CenterNet [65] detector with a variant of Deep Layer Aggregation (DLA-34) [57] as backbone and utilize the iterative deep aggregation module (IDA) to recover a high-resolution feature map with stride 4. We also add a 3×3 deformable convolution layer [66] before every upsampling stage. The backbone network is initialized with the parameters pre-trained on COCO [26] and then pre-trained on CrowdHuman [39] with self-supervised learning as FairMOT [58]. The proposed correlation module is augmented before IDA module to fuse multi-scale correlation. For the correlation module, we set local region size $R = 5$ and dilation rate $D = 2$.

Training and Validation Datasets. For a fair comparison, we also use the default training datasets as Fair-MOT [58]. There are six training datasets including the ETH [11], CityPerson [62], CalTech [9], CUHK-SYSU [52], PRW [63] and MOT17 [30]. ETH and CityPerson only provide box annotations, so we ignore the ReID losses from these datasets. CalTech, CUHK-SYSU, PRW

Table 1. Evaluation of correlation architecture on the MOT17 [30] validation set.

Method	Two frames	MOTA ↑	IDF1↑	ID Sw. ↓	Speed↑
baseline [58]	✗	69.1%	72.9%	**299**	25.6
non-local [48]	✗	67.7%	70.4%	311	16.60
CorrNet [46]	✗	70.0%	73.3%	303	22.93
SLC (ours)	✗	70.3%	**75.8**%	258	20.19
concat-raw [64]	✔	69.3%	74.1%	336	23.99
concat-feat [34]	✔	70.4%	74.0%	308	19.77
TLC (ours)	✔	70.9%	74.7%	326	19.26
STLC (ours)	✔	71.5%	76.1%	307	16.56

Table 2. Ablation studies on MOT17 validation set. "LT" and "Self" denote using the proposed long-term memory and self-supervised loss, respectively.

Method	MOTA ↑	IDF1↑	ID Sw. ↓	Speed↑
STLC	71.5%	**76.1**%	307	**16.56**
STLC+LT	**72.1**%	75.6%	311	**15.62**
STLC+LT+Track Loss	**72.1**%	**76.1**%	**299**	**15.62**
STLC+LT+Self Loss	**72.4**%	**77.6**%	301	**15.62**

and MOT17 provide both box and identity annotations which allows us to train both branches. We de-duplicate the overlaps between ETH datasets and MOT16 for fair comparison. For all validation experiments, we use the six datasets mentioned above and the first half frames of MOT17 as training, and the second half of MOT17 as the validation set.

We train on an input resolution of 1088×608, which yields an output resolution of 272×152. We use random flip, random scaling (between 0.5 to 1.2), cropping, and color jittering as data augmentation, and use Adam [22] to optimize the overall objective. The learning rate is initialized as $1e^{-4}$ and then decayed to $1e^{-5}$ in the last 10 epochs. We train with a batch-size of 12 (on 2 GPUs) for 30 epochs. In the training phase, we sample 5 temporally ordered frames within a random interval of less than 3.

Test Datasets and Evaluation Metrics. We evaluate the performance of our correlation tracker on the 2DMOT2015 [24], MOT16 [30], MOT17 [30], and MOT20 [8]. In particular, 2DMOT2015 [24] contains 11 test videos. MOT16 [30] and MOT17 [30] contain same 7 test videos with part different annotations. The MOT20 [8] consists of 4 test videos on extremely crowded scenes, which makes them really challenging.

We adopt the standard metrics of MOT Benchmarks for evaluation, including Multiple Object Tracking Accuracy (MOTA) [21], ID F1 Score [37], Mostly tracked targets (MT), Mostly lost targets (ML), the number of False Positives (FP), the number of False Negatives (FN), and the number of Identity Switch (ID Sw.) [25]. The run time is also provided and evaluated on a NVIDIA Tesla V100 GPU.

4.2. Ablation Studies

To elaborate on the effectiveness of the proposed approach, we conduct extensive ablation studies. First, we give detailed correlation analysis with different settings to justify our design choices, as presented in Table. 1. Next, the tracking accuracy and runtime for different region sizes of the correlation module is explored. Different building blocks are compared to illustrate the effectiveness and efficiency of the full correlation tracker.

Spatial correlation. In order to evaluate the effectiveness of our spatial local correlation module (SLC), we compared our baseline model [58] and two relation methods, the non-local module [48] and CorrNet [46]. Directly use of non-local brings performance degradation on MOT since non-local module does not record the relative positional relationship between the targets and usually brings performance drops on small object [54]. In addition, non-local method has a huge overhead in memory and computation. In video recognition, a learnable correlation filter network [46] is proposed, and the grouped convolution is used to reduce the amount of calculation. Our method has achieved similar MOTA compared to CorrNet, but has a large improvement on IDF1, which comes from our multi-scale correlation design. Compared with the baseline, our IDF1 has increased by 2.9%, and identity switches have been reduced by 15%, demonstrating the discrimination of our spatial correlation. The Re-ID embeddings cannot easily distinguish similar distractor, our correlation feature models geometric information and is better suited for MOT.

Temporal correlation. In the field of video object detection [38], temporal and global information are commonly used to improve performance [6]. The research of temporal detection for MOT is still preliminary. Recently, CenterTrack [64] concatenates the previous frame and the current frame, and Chained-Tracker [34] concatenates the high-level image features for two frames to fuse temporal information. We compare these two methods, dubbed *concat-raw* and *concat-feat*, with our temporal local correlation module (TLC). These two methods brought 0.2% and 1.3% MOTA improvements over the single frame baseline. Compared with these two methods, our temporal local correlation module achieves consistent improvements in both MOTA and IDF1. Our temporal correlation module helps for the temporal feature alignment around frames. At the same time, our method only adds a small overhead to feature-level concatenation, which proves the efficiency of our algorithm. Compared with the baseline FairMOT, employing both spatial and temporal local correlation (STLC) yields a IDF1 of 76.1% , which brings 3.2% improvement.

Long-term dependences. We also analyze the performance improvement with long-term correlation. Compared with the method using two frames as the source cues, our long-term method achieves a large improvement of 0.6% MOTA, due to the increased capacity for object detections. The improvement of MOTA also means an increase in the
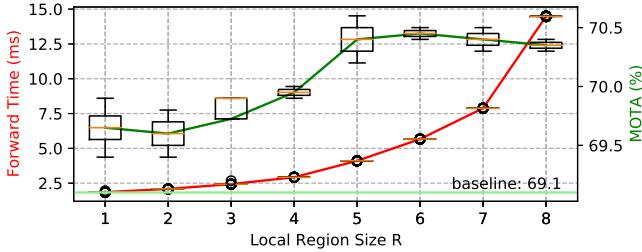
Figure 4. Effect of filter size R on speed and MOTA accuracy on MOT17-val.

upper bound of our tracker.

Self-supervised learning. For correlation learning, explicit supervision is usually not imposed [48, 46]. We have proposed two supervision methods in section 3.5. It can be seen that the siamese tracking supervision imposed to training has achieved a relatively good improvement in IDF1. There is no change in the run time of our algorithm, because the change in training loss does not change the inference processing. Self-supervised losses have also been appropriately improved on both MOTA and IDF1 due to more positive samples employed in correlation volume.

Choice of local region. Figure 4 shows the MOTA and run time of our correlation module for different region size $R \in \{1, 2, ..., 8\}$. As expected, a larger local size R can cover a larger neighborhood while matching pixels, thus yields a higher accuracy. But the improvements become marginal beyond $R = 5$, possibly due to the low resolution of the feature maps. Note that non-local module usually doubles the run time of the backbone and the cost of explicitly computing optical flow [10] can be very high as well. This shows that our correlation module is more efficient by learning motion information from features directly. Region size $R = 5$ yields a good trade-off between speed and accuracy. The computation overhead is relatively small, compared to the complexity of the whole detection networks.

4.3. Experiments on MOT Challenges

To extensively evaluate the proposed method, we compare it with 8 state-of-the-art trackers, which cover most of current representative methods. There are 2 joint detection and embedding methods (JDE [50] and FairMOT [58]), 2 multi-frame prediction methods (Tube_TK [32] and CTracker [34], 2 graph network based methods (MPN [5] and JDMOTGNN [49]), 2 offset prediction based methods (CenterTracker [64] and Tracktor++v2 [1]). The results are summarized in Table 3.

2DMOT2015 [24]. The evaluation on 2DMOT2015 is performed by the official toolkit. As shown in Table 3, our correlation tracker outperforms the top private method of 2DMOT2015, (*i.e.*, FairMOT [58]), by 1.7% in MOTA and 1.0% in IDF1. It is worth noting that the ID Switches are decreased by 13%, which shows the robustness of our cor-

Table 3. Comparisons of our method with state-of-the-arts on MOT benchmarks [24, 30, 8]. We set new state-of-the-art results by a significant margin in terms of MOTA and IDF1. Our correlation tracker is more accurate while running with high speed.

Method	MOTA ↑	IDF1 ↑	MT ↑	ML ↓	FP ↓	FN ↓	ID Sw. ↓	Hz ↑
2D MOT 2015 [24]								
CorrTracker (ours)	**62.3**	**65.7**	**49.0**	12.9	6909	**15728**	513	17.9
JDMOTGNN [49]	60.7	64.6	47.0	**10.5**	7334	16358	477	2.4
FairMOTv2 [58]	60.6	64.7	47.6	11.0	7854	15785	591	**30.5**
Tube_TK_POI [32]	58.4	53.1	39.3	18.0	5756	18961	854	5.8
MPN [5]	51.5	58.6	31.2	25.9	7620	21780	**375**	6.5
Tracktor++v2 [1]	46.6	47.6	18.2	27.9	**4624**	26896	1290	1.8
MOT16 [30]								
CorrTracker (ours)	**76.6**	**74.3**	**47.8**	13.3	10860	**30756**	979	14.8
FairMOTv2 [58]	74.9	72.8	44.7	15.9	10163	34484	1074	**25.4**
CTracker [34]	67.6	57.2	32.9	19.0	8934	48305	1897	6.8
Tube_TK_POI [32]	66.9	62.2	39.0	16.1	11544	47502	1236	1.0
POI [56]	66.1	65.1	34.0	20.8	5061	55914	805	9.9
Tracktor++v2 [1]	56.2	54.9	20.7	35.8	**2394**	76844	**617**	1.8
MOT17 [30]								
CorrTracker (ours)	**76.5**	**73.6**	**47.6**	12.7	29808	**99510**	3369	14.8
FairMOTv2 [58]	73.7	72.3	43.2	17.3	27507	117477	3303	**25.9**
CenterTrack [64]	67.8	64.7	34.6	24.6	18498	160,332	3039	3.8
CTracker [34]	66.6	57.4	32.2	24.2	22284	160491	5529	6.8
Tracktor++v2 [1]	56.3	55.1	21.1	35.3	**8866**	235449	**1987**	1.8
MOT20 [8]								
CorrTracker (ours)	65.2	**69.1**	66.4	8.9	79429	95855	5183	8.5
JDMOTGNN [49]	**67.1**	67.5	53.1	13.2	**31913**	135409	**3131**	0.9
FairMOTv2 [58]	61.8	67.3	**68.8**	**7.6**	103440	**88901**	5243	**13.2**

relation module. Moreover, our tracker is superior to the recent end-to-end graph trackers JDMOTGNN [49]. Our feature propagation approach can absorb both foreground and background information, which improves our tracker by 1.6% in terms of MOTA.

MOT16 [30] and MOT17 [30]. Table 3 reports the evaluation results with the comparisons to recent prevailing trackers on MOT16. The recent proposed FairMOTv2 [58] achieves the second performance in MOTA and IDF1, while our method ranks first with 76.6% MOTA, outperforming other private approaches by a significant margin. Our CorrTracker achieves the best performance on MOT17, surpassing FairMOTv2 by 2.8% MOTA and 1.3% IDF1. Moreover, the FN of our CorrTracker surpasses FairMOTv2 by 15%, which means nearly 20,000 new bounding boxes are added to the association process. In this case, our algorithm still maintains comparable or even superior ID Switches, which actually proves that our method significantly improves the tracking association. As reported in Table 3, our CorrTracker, CenterTrack [64] and CTrack [34] all use multi-frame cues to predict detections, our FN is largely decreased by 30%.

MOT20 [8]. To further evaluate the proposed models, we report the results on MOT20, which is more challenging than MOT17. The final results is presented in the bottom block of Table 3. Our CorrTracker achieves MOTA score of 65.2%, substantially outperforming FairMOTv2 [58] with MOTA of 61.8%. Although our approach is an order of magnitude faster than JDMOTGNN [49] in speed, our accuracy is slightly worse due to the anchor-free design.
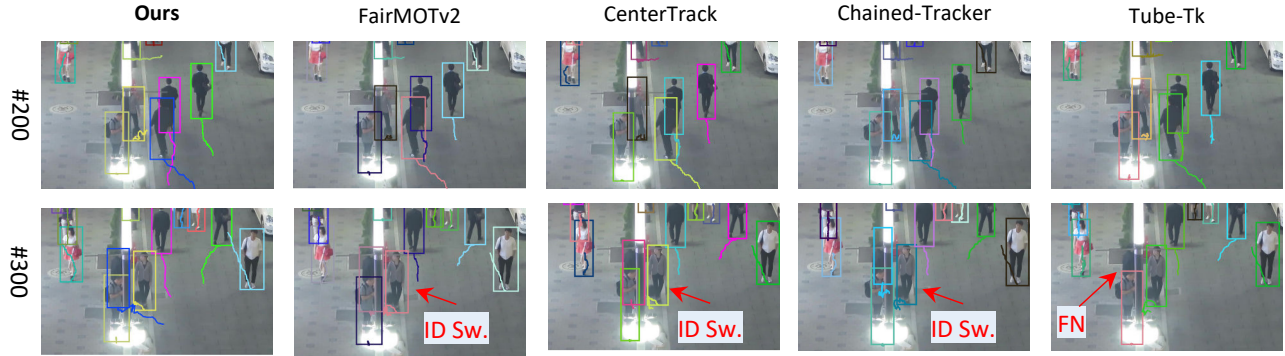
Figure 5. Qualitative comparisons against several prior methods [58, 64, 34, 32] in occlusion situations. Frames are sampled from MOT17-03. Our CorrTracker can identify objects via mining the context patterns around targets.

Figure 6. Qualitative results of our correlation tracker on MOT17 [30] and MOT20 [8]. The color of each bounding box indicates the target identity. The dotted line under each bounding box denotes the recent tracklet of each target. The proposed tracker predicts trajectories with substantially robust and temporally consistent.

4.4. Visualization

We visualize the tracking trajectories for prior methods, *i.e.*, the center offset [64] and multi-frame bounding boxes [34, 32], in Figure 5. We observe that our correlation map focuses on the entire context, while the regular appearance feature concentrates on the local region of the target. Our correlation module improves the reliability of recognition since it provides a global view of the target. Methods based on offset prediction, *e.g.*, CenterTrack [64] and CTracker [34], can easily generate id switches when encountered with complex object interactions. Figure 6 shows qualitative results of our Correlation Tracker on MOT17 and MOT20, the advantage over existing method is most pronounced on the robust to occlusion and tiny objects.

5. Conclusion

In this work, we propose a novel correlation tracking framework based upon the observation that the relational structure helps to distinguish similar objects. Our correlation module densely matches all targets with their local context and learn a discriminative embeddings from the correlation volumes. Furthermore, we show how to extend the correlation module from spatial layout to the adjacent frames for strengthening the temporal modeling ability. We explore that self-supervised learning to impose a discriminative constraint on the correlation volume, which explicitly predicts a instance flow. Extensive experiments on four MOT challenges demonstrate that our CorrTracker achieves state-of-the-art performance and is efficient in inference.

References

[1] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixe. Tracking without bells and whistles. In *Proceedings of the IEEE international conference on computer vision*, pages 941–951, 2019.

[2] Luca Bertinetto, Jack Valmadre, Joao F Henriques, Andrea Vedaldi, and Philip HS Torr. Fully-convolutional siamese networks for object tracking. In *European conference on computer vision*, pages 850–865. Springer, 2016.

[3] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In *2016 IEEE International Conference on Image Processing (ICIP)*, pages 3464–3468. IEEE, 2016.

[4] Erik Bochinski, Volker Eiselein, and Thomas Sikora. High-speed tracking-by-detection without using image information. In *2017 14th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, pages 1–6. IEEE.

[5] Guillem Brasó and Laura Leal-Taixé. Learning a neural solver for multiple object tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6247–6257, 2020.

[6] Yihong Chen, Yue Cao, Han Hu, and Liwei Wang. Memory enhanced global-local aggregation for video object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10337–10346, 2020.

[7] Peng Chu and Haibin Ling. Famnet: Joint learning of feature, affinity and multi-dimensional assignment for online multiple object tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6172–6181, 2019.

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

[9] Piotr Dollár, Christian Wojek, Bernt Schiele, and Pietro Perona. Pedestrian detection: A benchmark. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 304–311. IEEE, 2009.

[10] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766, 2015.

[11] Andreas Ess, Bastian Leibe, Konrad Schindler, and Luc Van Gool. A mobile vision system for robust multi-person tracking. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[12] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Detect to track and track to detect. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3038–3046, 2017.

[13] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *IEEE transactions on pattern analysis and machine intelligence*, 32(9):1627–1645, 2009.

[14] Ross Girshick. Fast r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 1440–1448, 2015.

[15] Roberto Henschel, Laura Leal-Taixé, Bodo Rosenhahn, and Konrad Schindler. Tracking with multi-level features. *arXiv preprint arXiv:1607.07304*, 2016.

[16] Roberto Henschel, Yunzhe Zou, and Bodo Rosenhahn. Multiple people tracking using body and joint detections. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.

[17] Andrea Hornakova, Roberto Henschel, Bodo Rosenhahn, and Paul Swoboda. Lifted disjoint paths with application in multiple object tracking. In *The 37th International Conference on Machine Learning (ICML)*, July 2020.

[18] Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. Relation networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3588–3597, 2018.

[19] Joel Janai, Fatma Güney, Aseem Behl, Andreas Geiger, et al. Computer vision for autonomous vehicles: Problems, datasets and state of the art. *Foundations and Trends® in Computer Graphics and Vision*, 12(1–3):1–308, 2020.

[20] R. E. Kalman. A new approach to linear filtering and prediction problems. *ASME Journal of Basic Engineering*, 1960.

[21] Rangachar Kasturi, Dmitry Goldgof, Padmanabhan Soundararajan, Vasant Manohar, John Garofolo, Rachel Bowers, Matthew Boonstra, Valentina Korzhova, and Jing Zhang. Framework for performance evaluation of face, text, and vehicle detection and tracking in video: Data, metrics, and protocol. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 31(2):319–336, 2008.

[22] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[23] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955.

[24] L. Leal-Taixé, A. Milan, I. Reid, S. Roth, and K. Schindler. MOTChallenge 2015: Towards a benchmark for multi-target tracking. *arXiv:1504.01942 [cs]*, Apr. 2015. arXiv: 1504.01942.

[25] Yuan Li, Chang Huang, and Ram Nevatia. Learning to associate: Hybridboosted multi-target tracker for crowded scene. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2953–2960. IEEE, 2009.

[26] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[27] Chen Long, Ai Haizhou, Zhuang Zijie, and Shang Chong. Real-time multiple people tracking with deeply learned candidate selection and person re-identification. In *ICME*, 2018.

[28] Xiankai Lu, Wenguan Wang, Chao Ma, Jianbing Shen, Ling Shao, and Fatih Porikli. See more, know more: Unsupervised video object segmentation with co-attention siamese

networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3623–3632, 2019.

[29] Zhichao Lu, Vivek Rathod, Ronny Votel, and Jonathan Huang. Retinatrack: Online single stage joint detection and tracking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14668–14678, 2020.

[30] A. Milan, L. Leal-Taixé, I. Reid, S. Roth, and K. Schindler. MOT16: A benchmark for multi-object tracking. *arXiv:1603.00831 [cs]*, Mar. 2016. arXiv: 1603.00831.

[31] Sangmin Oh, Anthony Hoogs, Amitha Perera, Naresh Cuntoor, Chia-Chih Chen, Jong Taek Lee, Saurajit Mukherjee, JK Aggarwal, Hyungtae Lee, Larry Davis, et al. A large-scale benchmark dataset for event recognition in surveillance video. In *CVPR 2011*, pages 3153–3160. IEEE, 2011.

[32] Bo Pang, Yizhuo Li, Yifan Zhang, Muchen Li, and Cewu Lu. 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*, pages 6308–6318, 2020.

[33] Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and Dustin Tran. Image transformer. In *Proceedings of the 35th International Conference on Machine Learning*, pages 4055–4064. PMLR, 2018.

[34] Jinlong Peng, Changan Wang, Fangbin Wan, Yang Wu, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yanwei Fu. Chained-tracker: Chaining paired attentive regression results for end-to-end joint multiple-object detection and tracking. In *Proceedings of the European Conference on Computer Vision*, 2020.

[35] Joseph Redmon and Ali Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[36] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[37] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *European Conference on Computer Vision*, pages 17–35. Springer, 2016.

[38] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[39] Shuai Shao, Zijian Zhao, Boxun Li, Tete Xiao, Gang Yu, Xiangyu Zhang, and Jian Sun. Crowdhuman: A benchmark for detecting human in a crowd. *arXiv preprint arXiv:1805.00123*, 2018.

[40] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 8934–8943, 2018.

[41] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, Vijay Vasudevan, Wei Han, Jiquan Ngiam, Hang Zhao, Aleksei Timofeev, Scott Ettinger, Maxim Krivokon, Amy Gao, Aditya Joshi, Yu Zhang, Jonathon Shlens, Zhifeng Chen, and Dragomir Anguelov. Scalability in perception for autonomous driving: Waymo open dataset. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[42] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. *arXiv preprint arXiv:2003.12039*, 2020.

[43] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[44] Paul Voigtlaender, Michael Krause, Aljosa Osep, Jonathon Luiten, Berin Balachandar Gnana Sekar, Andreas Geiger, and Bastian Leibe. Mots: Multi-object tracking and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7942–7951, 2019.

[45] Carl Vondrick, Abhinav Shrivastava, Alireza Fathi, Sergio Guadarrama, and Kevin Murphy. Tracking emerges by colorizing videos. In *Proceedings of the European conference on computer vision (ECCV)*, pages 391–408, 2018.

[46] Heng Wang, Du Tran, Lorenzo Torresani, and Matt Feiszli. Video modeling with correlation networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 352–361, 2020.

[47] Huiyu Wang, Yukun Zhu, Bradley Green, Hartwig Adam, Alan Yuille, and Liang-Chieh Chen. Axial-deeplab: Stand-alone axial-attention for panoptic segmentation. *arXiv preprint arXiv:2003.07853*, 2020.

[48] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7794–7803, 2018.

[49] Yongxin Wang, Xinshuo Weng, and Kris Kitani. Joint detection and multi-object tracking with graph neural networks. *arXiv preprint arXiv:2006.13164*, 2020.

[50] Zhongdao Wang, Liang Zheng, Yixuan Liu, and Shengjin Wang. Towards real-time multi-object tracking. In *European Conference on Computer Vision*, 2020.

[51] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE international conference on image processing (ICIP)*, pages 3645–3649. IEEE, 2017.

[52] Tong Xiao, Shuang Li, Bochao Wang, Liang Lin, and Xiaogang Wang. Joint detection and identification feature learning for person search. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3415–3424, 2017.

[53] Jiarui Xu, Yue Cao, Zheng Zhang, and Han Hu. Spatial-temporal relation networks for multi-object tracking. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3988–3998, 2019.

[54] Minghao Yin, Zhuliang Yao, Yue Cao, Xiu Li, Zheng Zhang, Stephen Lin, and Han Hu. Disentangled non-local neural networks. *arXiv preprint arXiv:2006.06668*, 2020.

[55] Fisher Yu, Vladlen Koltun, and Thomas Funkhouser. Dilated residual networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 472–480, 2017.

[56] Fengwei Yu, Wenbo Li, Quanquan Li, Yu Liu, Xiaohua Shi, and Junjie Yan. Poi: Multiple object tracking with high performance detection and appearance feature. In *European Conference on Computer Vision*, pages 36–42. Springer, 2016.

[57] Fisher Yu, Dequan Wang, Evan Shelhamer, and Trevor Darrell. Deep layer aggregation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2403–2412, 2018.

[58] Yifu Zhan, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. A simple baseline for multi-object tracking. *arXiv preprint arXiv:2004.01888*, 2020.

[59] Hong Zhang and Naiyan Wang. On the stability of video detection and tracking. *arXiv preprint arXiv:1611.06467*, 2016.

[60] Jimuyang Zhang, Sanping Zhou, Xin Chang, Fangbin Wan, Jinjun Wang, Yang Wu, and Dong Huang. Multiple object tracking by flowing and fusing. *arXiv preprint arXiv:2001.11180*, 2020.

[61] Li Zhang, Yuan Li, and Ramakant Nevatia. Global data association for multi-object tracking using network flows. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008.

[62] Shanshan Zhang, Rodrigo Benenson, and Bernt Schiele. Citypersons: A diverse dataset for pedestrian detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3213–3221, 2017.

[63] Liang Zheng, Hengheng Zhang, Shaoyan Sun, Manmohan Chandraker, Yi Yang, and Qi Tian. Person re-identification in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1367–1376, 2017.

[64] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Tracking objects as points. In *European Conference on Computer Vision*, 2020.

[65] Xingyi Zhou, Dequan Wang, and Philipp Krähenbühl. Objects as points. In *arXiv preprint arXiv:1904.07850*, 2019.

[66] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9308–9316, 2019.