End-to-End Multi-Object Tracking with Global Response Map

Xingyu Wan, Jiakai Cao, Sanping Zhou, and Jinjun Wang

Institute of Artificial Intelligence and Robotics,
Xi'an Jiaotong University, Xi'an, Shaanxi, 710049, China
xingyuwan@stu.xjtu.edu.cn

Abstract. Most existing Multi-Object Tracking (MOT) approaches follow the Tracking-by-Detection paradigm and the data association framework where objects are firstly detected and then associated. Although deep-learning based method can noticeably improve the object detection performance and also provide good appearance features for cross-frame association, the framework is not completely end-to-end, and therefore the computation is huge while the performance is limited. To address the problem, we present a completely end-to-end approach that takes image-sequence/video as input and outputs directly the located and tracked objects of learned types. Specifically, with our introduced multi-object representation strategy, a global response map can be accurately generated over frames, from which the trajectory of each tracked object can be easily picked up, just like how a detector inputs an image and outputs the bounding boxes of each detected object. The proposed model is fast and accurate. Experimental results based on the MOT16 and MOT17 benchmarks show that our proposed on-line tracker achieved state-of-the-art performance on several tracking metrics.

Keywords: Multiple Object Tracking, Global Response Map, End-to-End.

1 Introduction

Multi-Object Tracking (MOT) aims to use image measurements and predictive dynamic models to consistently estimate the states of multiple objects over discrete time steps corresponding to video frames. The major challenges of MOT are to continuously and effectively model the vast variety of objects with high uncertainty in arbitrary scenarios, caused by occlusions, illumination variations, motion blur, false alarm, etc [55]. There are three key issues that a MOT framework should handle: 1) Modeling the dynamic motion of multiple objects; 2) Handling the entering/exiting of objects into/from the scene; 3) Robustness against occlusion and appearance/background variations. Single object tracking [16] focus on 1) and 3) but simply applying multiple single object trackers for the MOT task usually gives very limited performance due to 2).

With the significant progress in object detection, tracking-by-detection framework [36] has become a leading paradigm whereby the detection results of objects are represented as bounding boxes and available in a video sequence as prior information. MOT is then casted as a problem of data association where the objective is to connect detection outputs into trajectories across video frames using suitable measurements. The performance of these approaches largely depends on two key factors: Firstly the quality of detection results, where if the detection is missing or inaccurate at a single frame, or when occlusion occurs, the target state is then hard to estimate, and the target identity is prone to be lost; Secondly the data association model, where to achieve robust association across frames given the dynamic of objects, many works [34,39,51,52] conduct MOT in an off-line fashion with iterative solver [72] in order to make use of detection from both past and future, but is therefore time consuming and sensitive to the quality of appearance feature for association, not to mention scenarios where on-line processing is required.

More recently, many works have been proposed to utilize deep learning techniques to train Convolutional Neural Networks (CNNs) [58,19] from large scale datasets to obtain rich feature representations. These models have significantly improved the object detection performance and the quality of appearance feature. Many MOT approaches [65,46,60] have adopted Deep Neural Networks (DNNs) for feature representation learning and feature metric learning for data association. They usually establish a robust motion model to predict the motion variations of targets and introduce a well trained appearance model to extract deep feature from region of interest (ROI) for image patches, and finally some similarity distances are adopted to measure the affinity of two ROIs for pair-wise association. Furthermore, aiming to learn a robust metric for feature representation, several works [37,47,45] take multiple features of objects in the scene by incorporating a myriad of components such as motion, appearance, interaction, etc. Some works [56,24] even consider to combine temporal components to analyze long-term variation by using Long Short-Term Memory (LSTM). Since these methods are still based on disjoint detection/association steps, the computation is huge, and the performance is limited without end-to-end (i.e., from image-sequence/video to trajectory) capacity. There are works that attempt end-to-end training for the tracking-by-detection and association framework [69,70], but these approaches do not change the non-end-to-end nature of the MOT framework.

In this paper, we introduce a true end-to-end framework for MOT. The challenge is finding a suitable representation that is capable to handle both issues 1), 2) and 3) in an on-line manner. Our idea is to employ a modified object salience model to generate a global response map to locate the presence of multiple objects, such that for issue 1), the motion of each object is implicitly modeled, and for 2) and 3), minor occlusions/entering/exting within the window of frames can be robustly handled. The global response map has multiple channels where each channel models the response for different attributes to define the state space of all trajectories, such as "presence of object", "x/y", "$\Delta x/\Delta y$", as well as any

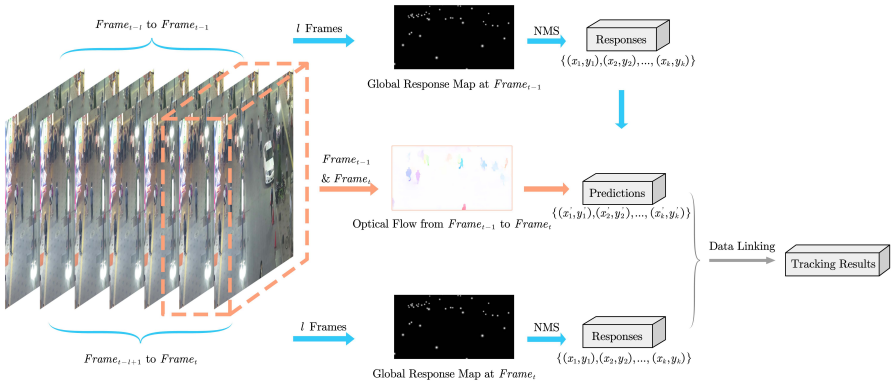

Fig. 1. The framework of our proposed MOT paradigm. This end-to-end framework is composed of several modules, which are object locating using global response map, motion displacement regression, and data linking using global assignment. The global response map is designed to extract the attributes of "presence of object" and "x/y", and motion displacement regressor is designed to retrieve the attribute of "$\Delta x/\Delta y$"

additional attributes in the future, such that 1) the object detection step is only implicitly modeled in the tracking process, and the spatial information of a target is no longer a bounding box region of interest, but a Gaussian-like distribution from 0 to 1; and 2) the inference process does not need complicated assignment process but just a simple linking step to extract the multiple trajectories. In this paper, we applied a logical inference approach to estimate the actual state of target response based on the sequence of global response maps. Conceptually speaking, the sub-module for extracting each attribute of a state is similar to an ad-hoc network. For example, the attributes of "presence of object" and "x/y" are from a sub-module similar to an object locating network, the attribute of "$\Delta x/\Delta y$" is from a sub-module similar to an optical flow extraction network, etc. This can be further extended to include *width*, *height*, *orientation*, *depth*, etc by adding suitable sub-modules in the future. The most important part is that, we are able to integrate all these sub-modules into one end-to-end network for MOT in one feed-forward step without any exhaustive cropping and iterating. The overall framework is illustrated in Fig. 1, and the main contributions of this work can be summarized as follows:

1. We propose a novel representation schema and network structure to perform end-to-end MOT of learned object types. Significantly different from existing tracking-by-detection and data association based approaches, our proposed method takes image-sequence/video as input and generates the trajectories of multiple objects in a true end-to-end fashion, where multiple attributes that define the state space of each trajectory are obtainable from the global response map generated by our model.

2. The proposed network includes a sub-network that operates like an image-sequence/video-based object locator and is capable of handling the occlusion

problem. From within the defined time window, the module can still maintain a positive response even when target is occluded, and thus significantly reducing the false negatives.

3. The proposed network also includes a sub-network that operates like an optical flow extraction network with a motion displacement regressor for estimating the motion dynamics. The module also helps solving the uncertain assignment problem in one single forward propagation.

4. The proposed multi-object tracking network is complete end-to-end without any detection/appearance priors. The experimental results show that our tracker achieves superior performance over the state-of-the-art approaches on public benchmarks.

2 Related Works

Detector-based Tracking. Owe to the galloping progresses of object detection techniques such as Faster R-CNN [42] and SDP [67], given these detection results as intialization/priors, MOT task can be conducted within tracking-by-detection paradigm [36] where the objective is to connect detection outputs into trajectories across video frames using reasonable measurements, which therefore casts the MOT problem as global data association. Traditional data association techniques including the Multiple Hypothesis Tracker (MHT) [41] and the Joint Probabilistic Data Association Filter (JPDAF) [13] aim to establish sophisticated models to capture the combinatorial complexity on a frame-by-frame basis. Both methods got improved later in conjunction with better appearance model [23] or more efficient approximation [15]. Aiming at global optimization with simplified models, the flow network formulations [71,39,59,8] and probabilistic graphical models [66,65,1,35] are considered, along with shortest-path, min-cost algorithms or even graph multi-cut formulations [53]. Most existing detector-based trackers highly rely on the quality of detection results, and to handle imperfect detections, several works [34,39,51,52] conduct MOT in off-line fashion to handle ambiguous tracking results for a robust tracking performance. Due to their off-line nature with low processing speed, they are not applicable to real-time vision tasks. Compare to these methods, our proposed approach runs in an on-line manner without being bounded to specific object detection techniques and does not require complicated data association step.

Deep Metric Learning for MOT. Learning effective feature representation with corresponding similarity measure plays a central role in data association. Metric learning based on DNNs for object appearance representation and computation of the affinity between measurements has become a popular trend. Various trackers [37,47,45,38,63,44] model different features of objects by incorporating a myriad of components such as motion, appearance, interaction, social behavior, etc. Leal-Taixe et al. [27] adopt a Siamese CNN to learn local features from both RGB images and optical flow maps. Robicquet et al.[44] introduced social sensitivity to describe the interaction between two targets and use this

definition to help the data association step. Later on, inspired by the success of Recurrent Neural Networks (RNNs) and their application to language modeling [54], several works have been trying to learn an end-to-end representation for state estimation utilizing RNNs [45,33]. Sadeghian et al. [45] proposed an offline metric learning framework using a hierarchical RNN to encode long-term temporal dependencies across multiple cues, i.e., appearance, motion and interaction. Milan et al. [33] presented an on-line RNN-based approach for multiple people tracking which is capable of performing prediction, data association and state update within a unified network structure. Followed by these works, [56] extended the research of RNN-based methods and leveraged the power of Long Short-Term Memory (LSTM) for learning a discriminative model of object trajectory by integrating dynamic features both in temporal and spatial. For the on-line MOT task, these methods may not perform well when heavy occlusion or mis-detection downgrade the robustness of appearance model. Differently, in our work, the occlusion problem is well handled by considering a window of frames to locate objects with an introduced logical inference methodology, without explicit appearance feature for metric learning.

3 Our Proposed MOT Algorithm

Traditional tracking-by-detection algorithm takes object detection and tracking as two separate tasks and adopts different CNNs respectively or applies cascade one. Within this procedure, different models and loss functions for different tasks are always needed, and therefore hard to achieve end-to-end training/inference. Aiming to integrating better DNNs-based detector into the visual tracking task, here we introduce an end-to-end MOT framework. Fig. 1 illustrates the proposed MOT framework, where we take consecutive image frames as network inputs, after learning a reasonable response map to locate interested targets globally, the target location is retrieved from response map using local NMS, and then we regress the motion displacements for these targets from a frame-wise optical-flow-like offset, after that we conduct global assignment between predictions and observations.

Our proposed MOT algorithm is organized as follows. The first section talks about the object locating sub-network, the next section is the motion displacement regression sub-network, and the final section is data linking strategy using global assignment.

3.1 Object Locating using Global Response Map

The goal of tracking is to consistently maintain the estimation of object states over discrete time step. In this specific computer vision task, using a well trained class-specific detector to filter out all the regions of interest over the image frame may not be necessary. Here we propose a simpler and more efficient way to locate objects for MOT. For all the targets to be tracked at each time step, we take them all as foreground objects and represent them as Gaussian-like distributions

from 0 to 1 with peak value at their center points on a saliency map. As shown in Fig. 2(a), each Gaussian-like distribution represents a foreground object to be tracked, the x, y coordinates correspond to the object spatial location, and z is a value from 0 to 1 corresponds to the actual status of object at current time step. The radius r and sigma σ of each distribution are defined as follow,

$$r = \min_{i=1}^{3} |\frac{a_i + \sqrt{a_i^2 - b_i}}{2}|, \qquad \sigma = \frac{r}{3}. \tag{1}$$

where

$$a_i = \begin{cases} h + w, \ i = 1 \\ 2 \times (h + w), \ i = 2 \\ -2 \times (h + w), \ i = 3. \end{cases} \tag{2}$$

$$b_i = \begin{cases} 4 \times \frac{h \times w \times (1-\alpha)}{1+\alpha}, \ i = 1 \\ 16 \times h \times w \times (1 - \alpha), \ i = 2 \\ 16 \times h \times w \times \alpha \times (\alpha - 1), \ i = 3. \end{cases} \tag{3}$$

Here h and w denote the height and width of target bounding box obtained from the ground-truth of training data, and α is an invariant parameter we set to 0.7 in this work. Given a bounding box scale w and h, we first compute three different radius $\{r_i, i = 1, 2, 3\}$, and we adopt the minimum value as the radius r of our Gaussian kernel, and the sigma σ is set to $\frac{r}{3}$ accordingly. In this way, our Gaussian-like distribution has a positive correlation with target size w and h.

Response Map Learning Network. The above representation as global response map is able to describe the object spatial location and actual state at the same time. To learn such representation, our tracking algorithm employs a HED-based [61] saliency detection network modified from [20]. Specifically, our object locating sub-network is an Auto-Encoder, which takes a time window of frames with a length l as inputs, and outputs a single channel response map after a *sigmoid* function. We adopt the short connection strategy as [20] but remove the fusion layer, and we compute the average value of $1, 2, 3, 6$ side outputs as our network output before activation. Given a training sequence $X_l = \{I^{t-l}, ..., I^t, I^i \in \mathbb{R}^{3*h*w}, i = t - l, ..., t\}$, and label response map $Y \in \mathbb{R}^{1*h*w} = \{y_j, j = 1, ..., |Y|\}$, the standard cross entropy loss function for our network is given by

$$L(X_l, Y) = -\sum_{j=1}^{|Y|} (y_j log\mathbf{P}(y_j = 1|X_l) + (1 - y_j) log\mathbf{P}(y_j = 0|X_l)), \tag{4}$$

where $\mathbf{P}(y_j = 1|X_l)$ denotes the probability of the activation value at location j, and label Y is obtained using the following logical inference methodology from the ground-truth of training data.

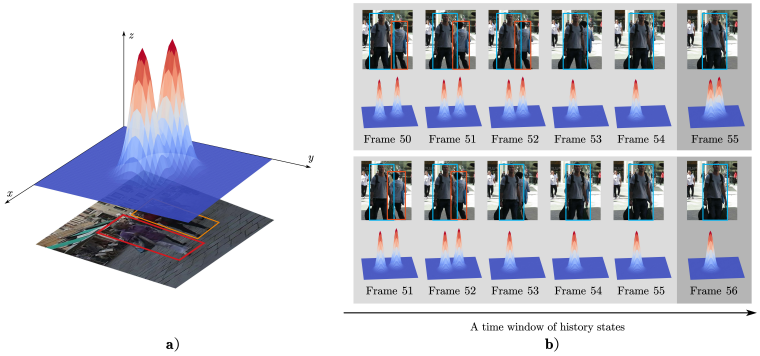

Fig. 2. Ground truth response maps, in which: **(a)** An illustration of local response map viewed in 3D. **(b)** An exemplary of logical inference methodology for target response estimation, row 1 and row 2 indicate the case 1 while row 3 and row 4 indicate the case 2. The actual states of two observed objects(annotated by blue and orange bounding boxes) at current frame (frame 55 for case 1, and frame 56 for case2) are estimated by a time window of history states

Logical Inference Methodology for Handling Occlusion. When a target being tracked is unseen at a particular time step, that does not mean this target actually leaves the surveillance scene. In this work, we argue that the actual state of object presence in visual tracking scenes should be distinguished from image-based detection results, and this estimation of actual state can be learned using history priors. As described above, we use a 0/1 response value to represent the target actual state at each time step. This representation should be estimated using not only image at current frame, but also images from the past. Here we introduce a logical inference methodology for estimating response value of each target. For target trajectories $\{T_j, j = 1, 2, ...m\}$ from ground-truth, the response value z_j^t of target actual state at frame t is estimated upon a time window of past states $\{z_j^i, i = t-l, t-l+1, ..., t-1\}$ with a length l. The specific estimation method is described as follow,

$$z_j^t = \begin{cases} 1, & \text{if } z_j^{t-1} = 1 \text{ or } \frac{\sum_{i=t-l}^{i=t-1} z_j^i}{l} >= \beta \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

Here β is a constant describing the proportion of positive states within the time window. In general, if a target appeared to be existing during most of the past time, we take it still exist at current time step even when we get a negative result from image-based detector. Conversely, if a target remained negative during most of the past time, we consider it actually leave the scene and take the positive detection of current time step as a false alarm. In addition, if a target has a positive response at last time step, i.e., $z_j^{t-1} = 1$, we set $z_j^t = 1$ accordingly no matter what the history states are. At training phase, this estimation strategy is employed to generate positive/negative samples from training data. By using

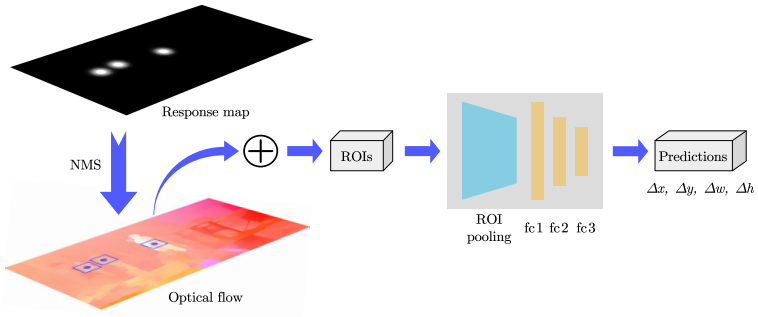

Fig. 3. The network structure of motion displacement regression. The inputs of this network are the center point distributions of located objects retrieved from global response map after NMS. The outputs are predicted motion displacements of all positive responses

this strategy, we can consistently maintain the correct positive responses for targets even when suffering occlusions.

3.2 Motion Displacement Regression

To obtain the predicted motion displacements of all located targets, most popular approaches are to iteratively crop the image patch of each region of interest(ROI) to learn a regressed motion prediction. These approaches are time consuming and hard to achieve end-to-end training/inference. In our proposed MOT algorithm, the motion dynamics are estimated from a motion displacement regression sub-network using a frame-wise optical-flow-like offset. As proposed in [21], we adopt FlowNet2 for frame-wise optical flow estimation. Given two adjacent frames I^{t-1} and I^t, the optical flow estimation from frame $t-1$ to frame t can be derived as $W^{t-1} = \sum_i (u_i, v_i), W^{t-1} \in \mathbb{R}^{2*h*w}$, where i denotes each pixel on the flow. After deriving the pixel-wise displacements from optical flow, we introduce a regression network to learn the predicted motion displacements globally for all responses with Gaussian-like distributions. As shown in Fig. 3, we first conduct local NMS with a kernel size s on response map Z^t at frame t with a threshold value of response $Score$ to filter out the top k positive responses and retrieve their center point locations. For each retrieved response distribution with a center point (cx, cy) and fixed kernel size r_z, we take it as a region of interest (ROI), and sample all the displacements of ROIs from optical flow at the same time to obtain a concatenated featuremap $F^{t-1} \in \mathbb{R}^{k*2*r_z*r_z}$ for regression. The regression network is composed of a ROI pooling [14] layer and several fully-connected layers. This network structure is designed to learn one accurate displacement value of response point from a ROI. The network output $D^t = \{d_j = (\Delta cx, \Delta cy, \Delta w, \Delta h), j = 1, ..., k\}$ is a movement displacement vector of all response points from frame $t-1$ to frame t. Given the ground-truth G^t and network output D^t, the loss function of our regression network is defined

as

$$smooth_{L_1}(G^t, D^t) = \begin{cases} 0.5(x)^2, & \text{if } |x| < 1 \\ |x| - 0.5, & \text{otherwise.} \end{cases} \tag{6}$$

where x denotes the L1 loss between G^t and D^t. Incorporating ROI pooling into our regression network enables our tracker to estimate frame-wise motion displacements of all the observed targets in one single forward propagation without any cropping and network iteration.

3.3 Data Linking Strategy

At tracking phase, the global response map Z^t and motion displacement estimation D^t are already obtained using the proposed framework, then our tracker conduct global assignment between predictions and observations. We first obtain the predicted location $(cx^{'}, cy^{'}, w^{'}, h^{'})$ by adding the regressed motion displacement $(\Delta cx, \Delta cy, \Delta w, \Delta h)$ to the previous target location. Followed by this, we compute the assignment cost matrix between predictions D^t and observations Z^t with the intersection-over-union (IOU) distance defined as Eq. 7, then we solve the assignment problem optimally using the Hungarian algorithm [26].

$$IOU(a, b) = \frac{Area(a) \cap Area(b)}{Area(a) \cup Area(b)} \tag{7}$$

Specifically, we first compute the IOU distance $IOU(D^t, Z^t)$ between each predicted location $d_k^t \in D^t$ with its nearest neighbourhood response $z_k^t \in Z^t$ at frame t. The nearest neighbourhood response is picked by solving the shortest path using center points distance of two distributions. After that, we pick the responses from observation space with the max IOU value higher than IOU_{min} as the candidate, and compute the IOU distance between the response locations at frame $t-1$ and the candidate locations at frame t. Followed by this, the assignment problem leads to an optimal association between detections and candidates which can be solved by applying the Hungarian algorithm [26] to maximize the sum of all IOUs at frame t. After this global assignment approach, we then adopt a matching cascade strategy for all unmatched responses and tracks at current frame t similar as [60]. Specifically, we set a constant parameter A_{max} denotes the maximum age. For each response z_k^t not assigned to an existing track at current frame, we compute the IOU distance between this response with each terminated track whose last frame is within a time window from $t-1$ to $t-A_{max}$. This computation is iteratively conducted frame-by-frame until the IOU distance is higher than IOU_{min}, and we take this response z_k^t to update the target state of corresponding terminated track as a match. After this matching cascade, all responses not assigned to an existing track will be initialized as a new track, and all tracks without an assigned response will be terminated.

Fig. 4. Qualitative results of our object locating method (column 1) and two popular detectors (Faster-RCNN [42](column 2) and SDP [67](column 3) respectively) on some challenging cases, best viewed in color. Our object response can still be located on the object even when occluded, which significantly helps the following data linking module to obtain a complete trajectory for the object

4 Experiment

4.1 Experimental Details

We report the performance of the proposed MOT algorithm on the MOT16 [32] and MOT17 [32] benchmark datasets. The provided public detection results of MOT16 dataset is DPM [11]. The MOT17 dataset contains the same video sequences(7 fully annotated training sequences and 7 testing sequences) as MOT16 but with two more sets of public detection results from Faster-RCNN [42] and SDP [67] respectively. We implemented our framework in Python3.6 using PyTorch, with six cores of 2.4GHz Inter Core E5-2680 and a NIVIDIA GTX 1080 GPU.

For object locating, here we set the length l of time window to 5. The parameter β of logical estimation methodology is set to 0.6. At the training phase, we use the VGGNet [50] pre-trained on the ImageNet dataset [9] as the shared base convolution blocks. The input of our network is a sequence of image frames which are resized to 512×960 for height and width. The total training epochs are 300 and the learning rate is initialized to $1e-3$ and divided by 10 every 100 epochs.

For motion displacement regression, we trained our model on 7 MOT training sequences with provided ground-truth tracking results. Here we set local NMS kernel size s to 3 and the threshold value of response $Score$ to 0.05 , the parameter k for maximum number of positives to 60, and the fixed kernel size r_z

Table 1. Run-time comparison among our proposed object locater and several popular detectors

Method	ms/frame	FPS
Fastest-DPM [64]	66	15
Fast-RCNN [14]	320	3
Faster-RCNN [42]	198	5
YOLO [40]	22	45
SSD300 [29]	17	58
Response Map (**Ours**)	**5**	**200**

of ROI to 20. At the training phase, we use the FlowNet2 [21] pre-trained on MPI-Sintel [4] for extracting frame-wise optical flow. The input of our network for motion variation regression is a sequence of adjacent image frames which are resized to 1024×1920 for height and width. We set the initial learning rate to $1e-4$, the total training epoches to 500 and divide the learning rate by 2 at epoch 166, 250, 333 and 416 respectively. We use Adam [25] optimizer for both sub-networks. The parameters IOU_{min} and A_{max} for data linking are set to 0.7 and 30 respectively.

4.2 Object Locating Performance

In order to exam the validity of our proposed representation schema, we visualized the results of object locating sub-network as well as the provided detection results on some challenging sequences from MOT benchmark [32]. As shown in Fig.4, compared with the official detection results from Faster-RCNN (column 2) and SDP (column 3) on challenging scenarios like small scales (row 1), crowded (row 2) and occluded (row 3), our object locating method (column 1) gives more accurate estimations centered on the foreground objects. Furthermore, at extremely occluded scenario (row 3), benefit from logical inference on history states, our method can still give positive and accurate responses for those objects completely occluded at this frame, while detector-based trackers have to make a further analyze on such case by introducing more complicate tricks.

In addition to the accuracy term, our object locating method runs much faster than DNN-based detectors. In Table 1, we compared the running time of our proposed object locating method with several popular detectors. Here we report the running times of classic DPM detector with a speeding up version [64], two-stage CNN-based detectors Fast-RCNN of VGG-16 version [14] and Faster-RCNN with VGG-16 for both proposal and detection [42], one-stage CNN-based detectors YOLO using VGG-16 [40] and SSD with 300×300 image size [29]. Our method were measured by computing the average run-time on 7 test sequences from MOT benchmark [32]. The FPS of our method for one feed-forward propagation is 40 times faster than classic Faster-RCNN [42] and about $3-4$ times faster than real-time detectors YOLO [40] and SSD [29]. These evaluation results confirm that object locating from the global response map is faster and more accurate. But more importantly, different from tracking-by-detection

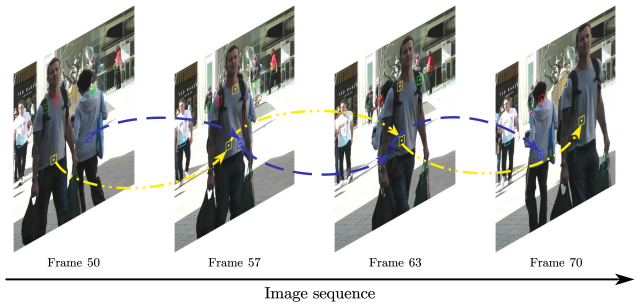

Fig. 5. An illustration of tracking results visualization using tracking as response points method on an extremely occluded scene.

and association based methods, this object locater is utilized as a sub-network within our MOT algorithm in an end-to-end fashion.

4.3 Tracking Performance on MOT Benchmark

Tracking as Response Points. Different from traditional tracking-by-detection paradigm, our proposed MOT method operates without using any detection results. Hence, instead of using the provided bounding box coordinates, our tracker only takes raw image-sequence/video as inputs, and tracks on each foreground response distribution. The representation of this tracking response points includes a center point coordinates (cx, cy) and a fixed kernel size r_z for each target. The exemplary visualization of our tracking as response points method on the MOT challenge dataset is shown in Fig. 5. Owe to the accurate center point locations retrieved from global response map and effective motion model, our tracker is rather robust for maintaining the identities of tracked targets during short-term occlusions.

Comparison to State-of-the-Art Trackers. Since the existing evaluation metrics for MOT are designed for traditional tracking-by-detection paradigm where the tracking results are provided and expressed as bounding box coordinates, in order to compare the performance of our tracker with other tracking methods, we ran one additional step that maps our generated response points to the provided detection bounding boxes from MOT challenge benchmark [32] so that the required metrics can then be calculated. Specifically, we compute the cost matrix between response distributions and bounding box detections using center point distance. Then we make the global optimal assignment by using Hungarian algorithm [26] same as section 3.3. We take the retrieved positive responses from object locating network as priors. For those detections from MOT dataset but not assigned to any response, we take them as false alarms. For the responses not assigned to any detection, we maintain the object location with

Table 2. Evaluation results on the MOT16 dataset

Mode	Method	MOTA ↑	MOTP ↑	IDF1 ↑	IDP ↑	IDR ↑	MT ↑	ML ↓	FP ↓	FN ↓	ID Sw. ↓	Frag ↓
×	STRN16 [62]	48.5	73.7	53.9	72.8	42.8	17.0%	34.9%	9,038	84,178	747	2,919
○	LMP [53]	48.8	**79.0**	51.3	71.1	40.1	18.2%	40.1%	6,654	86,245	461	595
○	KCF16 [6]	48.8	75.7	47.2	65.9	36.7	15.8%	38.1%	5,875	86,567	906	1,116
×	AFN [48]	49.0	78.0	48.2	64.3	38.6	19.1%	35.7%	9,508	82,506	899	1,383
×	eTC [57]	49.2	75.5	56.1	**75.9**	44.5	17.3%	40.3%	8,400	83,702	606	882
○	LSST16 [12]	49.2	74.0	56.5	77.5	44.5	13.4%	41.4%	7,187	84,875	606	2,497
○	HCC [30]	49.3	**79.0**	50.7	71.1	39.4	17.8%	39.9%	5,333	86,795	**391**	**535**
○	NOTA [5]	49.8	74.5	55.3	75.3	43.7	17.9%	37.7%	7,248	83,614	614	1,372
○	Tracktor16 [2]	54.4	78.2	52.5	71.3	41.6	19.0%	36.9%	**3,280**	79,149	682	1,480
○	**Ours**	**62.0**	73.6	**63.8**	70.5	**58.3**	**37.7%**	**20.7%**	18,308	**50,039**	909	2,009

Table 3. Evaluation results on the MOT17 dataset

Mode	Method	MOTA ↑	MOTP ↑	IDF1 ↑	IDP ↑	IDR ↑	MT ↑	ML ↓	FP ↓	FN ↓	ID Sw. ↓	Frag ↓
×	STRN17 [62]	50.9	75.6	56.0	74.4	44.9	18.9%	33.8%	25,295	249,365	2,397	9,363
×	jCC [22]	51.2	75.9	54.5	72.2	43.8	20.9%	37.0%	25,937	247,822	1,802	2,984
○	NOTA [5]	51.3	76.7	54.5	73.5	43.2	17.1%	35.4%	20,148	252,531	2,285	5,798
○	FWT [17]	51.3	77.0	47.6	63.2	38.1	21.4%	35.2%	24,101	247,921	2,648	4,279
×	AFN17 [48]	51.5	77.6	46.9	62.6	37.5	20.6%	35.5%	22,391	248,420	2,593	4,308
×	eHAF17 [49]	51.8	77.0	54.7	70.2	44.8	23.4%	37.9%	33,212	236,772	1,834	**2,739**
×	eTC17 [57]	51.9	76.3	58.1	73.7	48.0	23.1%	35.5%	36,164	232,783	2,288	3,071
×	FAMNet [7]	52.0	76.5	48.7	66.7	38.4	19.1%	33.3%	14,138	253,616	3,072	5,318
○	JBNOT [18]	52.6	77.1	50.8	64.8	41.7	19.7%	35.8%	31,572	232,659	3,050	3,792
○	LSST17 [12]	54.7	75.9	62.3	**79.7**	51.1	20.4%	40.1%	26,091	228,434	**1,243**	3,726
○	Tracktorv2 [2]	56.3	**78.8**	55.1	73.6	44.1	21.1%	35.3%	**8,866**	235,449	1,987	3,763
○	**Ours**	**61.2**	74.8	**63.2**	70.4	**57.3**	**36.7%**	**22.0%**	55,168	**159,986**	3,589	7,640

an initialized bounding box scale, and this scale is adjusted by motion variation regression network during tracking process.

We evaluate our tracker on the test sets of both MOT16 and MOT17 benchmark. The evaluation is carried out according to the metrics used by the MOT benchmarks [3,43,28], which includes Multiple Object Tracking Accuracy (MOTA), Multiple Object Tracking Precision (MOTP), ID F1 Score (IDF1, the ratio of correctly identified detections over the average number of ground-truth and computed detections), ID Precision (IDP), ID Recall (IDR), Mostly tracked targets (MT), Mostly lost targets (ML), the total number of false positives (FP), the total number of false negatives (FN), the total number of identity switches (ID Sw.), the total number of times a trajectory is fragmented (Frag).

Table 2 and Table 3 present the quantitative performance respectively, in comparison with some of existing best performing published trackers both online and offline. Evaluation metrics with ↑ means that higher scores denote better performance, while ↓ means the lower scores denote better performance. Mode with mark ○ means an online method, while mark × means an offline method. From Table 2 and Table 3 we can see that the main evaluation metric MOTA of our proposed method surpasses all the other state-of-the-art trackers. Moreover, our tracker also achieves the best performance in IDF1, IDR, MT, ML and FN. Among these metrics, the highest MT and the lowest ML indicate the robustness of our tracker for maintaining the identity of tracked targets. Futhermore, the lowest FN confirms introducing object locating network to estimate the actual targets does make an effort for reducing the false negatives caused by image-based detector, and the occlusion problem is handled rather well.

Table 4. Timing of each components of our proposed algorithm

Component	ms/frame	FPS
Object locating (including NMS)	108	9
Motion displacement regression	7.5	133
Data association	47	21
Total	162.5	6

Table 5. Run-time(ms/frame) comparison on MOT challenge benchmark

Method	Detection	Tracking	Total	FPS
RAR16wVGG [10]	198	>600	>798	<1.5
POI [68]	198	>160	>358	<3
CNNMTT [31]	198	>150	>348	<3
TAP [73]	198	>120	>318	<3.5
Deep SORT [60]	198	>25	>223	<4.5
ours	**0**	**162.5**	**162.5**	**6**

Run-Time Efficiency. We investigate the run-time efficiencies of each module in our proposed MOT method. At online inference phase, the run-times of each component tested on MOT benchmark dataset [32] is listed in Table 4. Our tracking as response points method outputs trajectories for all responses in one feed-forward propagation, and the run-time (ms per image) of our tracker is $162.5ms$(6 FPS). This is noticeably faster than many "real-time" and online methods [10,68,31,73,60] where "tracking" is conducted after a non-negligible detection step, such as those from the MOT benchmark [32] based on Faster-RCNN [42] detectors (as shown in Table 5).

5 Conclusions

In this work, we introduce a novel object representation schema and a new network model to support end-to-end on-line MOT. The proposed approach is significantly different from existing tracking-by-detection and association based methods where the object detection step is only implicitly conducted by a more efficient object locating network, while other attributes of a tracked object, such as x/y and $\Delta x/\Delta y$, can be extracted by the corresponding sub-networks (the object locating sub-network and the motion displacement regression sub-network respectively) in one feed-forward propagation. The complete model is capable of describing the dynamic motions of multiple objects and can handle the entering/exiting/occlusion of objects robustly. The network generates a global response map as the intermediate output from which the trajectory of each object can be obtained. The proposed method is fast and accurate, and our evaluation based on the MOT benchmark show that the proposed tracker significantly outperforms many other state-of-the-art methods. We believe that such simple and effective algorithm can provide a new inspiration for the MOT community. Our extension to extract other attributes in a more complete motion tracking state

space, such as width w and height h of the object, Δw and Δh, as well as other attributes including *orientation*, *depth*, etc, is on-going.

References

1. Andriyenko, A., Schindler, K., Roth, S.: Discrete-continuous optimization for multi-target tracking. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1926–1933. IEEE (2012)
2. Bergmann, P., Meinhardt, T., Leal-Taixe, L.: Tracking without bells and whistles. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 941–951 (2019)
3. Bernardin, K., Stiefelhagen, R.: Evaluating multiple object tracking performance: the clear mot metrics. EURASIP Journal on Image and Video Processing **2008**, 1–10 (2008)
4. Butler, D.J., Wulff, J., Stanley, G.B., Black, M.J.: A naturalistic open source movie for optical flow evaluation. In: European conference on computer vision. pp. 611–625. Springer (2012)
5. Chen, L., Ai, H., Chen, R., Zhuang, Z.: Aggregate tracklet appearance features for multi-object tracking. IEEE Signal Processing Letters **26**(11), 1613–1617 (2019)
6. Chu, P., Fan, H., Tan, C.C., Ling, H.: Online multi-object tracking with instance-aware tracker and dynamic model refreshment. In: 2019 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 161–170. IEEE (2019)
7. Chu, P., Ling, H.: 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. pp. 6172–6181 (2019)
8. Dehghan, A., Tian, Y., Torr, P.H., Shah, M.: Target identity-aware network flow for online multiple target tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1146–1154 (2015)
9. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: Imagenet: A large-scale hierarchical image database. In: 2009 IEEE conference on computer vision and pattern recognition. pp. 248–255. Ieee (2009)
10. Fang, K., Xiang, Y., Li, X., Savarese, S.: Recurrent autoregressive networks for online multi-object tracking. In: 2018 IEEE Winter Conference on Applications of Computer Vision (WACV). pp. 466–475. IEEE (2018)
11. Felzenszwalb, P.F., Girshick, R.B., McAllester, D., Ramanan, D.: Object detection with discriminatively trained part-based models. IEEE transactions on pattern analysis and machine intelligence **32**(9), 1627–1645 (2009)
12. Feng, W., Hu, Z., Wu, W., Yan, J., Ouyang, W.: Multi-object tracking with multiple cues and switcher-aware classification. arXiv preprint arXiv:1901.06129 (2019)
13. Fortmann, T., Bar-Shalom, Y., Scheffe, M.: Sonar tracking of multiple targets using joint probabilistic data association. IEEE journal of Oceanic Engineering **8**(3), 173–184 (1983)
14. Girshick, R.: Fast r-cnn. In: Proceedings of the IEEE international conference on computer vision. pp. 1440–1448 (2015)
15. Hamid Rezatofighi, S., Milan, A., Zhang, Z., Shi, Q., Dick, A., Reid, I.: Joint probabilistic data association revisited. In: Proceedings of the IEEE international conference on computer vision. pp. 3047–3055 (2015)
16. Held, D., Thrun, S., Savarese, S.: Learning to track at 100 fps with deep regression networks. In: European Conference on Computer Vision. pp. 749–765. Springer (2016)

17. Henschel, R., Leal-Taixé, L., Cremers, D., Rosenhahn, B.: Fusion of head and full-body detectors for multi-object tracking. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 1428–1437 (2018)
18. Henschel, R., Zou, Y., Rosenhahn, B.: Multiple people tracking using body and joint detections. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 0–0 (2019)
19. Hong, S., You, T., Kwak, S., Han, B.: Online tracking by learning discriminative saliency map with convolutional neural network. In: International conference on machine learning. pp. 597–606 (2015)
20. Hou, Q., Cheng, M.M., Hu, X., Borji, A., Tu, Z., Torr, P.H.: Deeply supervised salient object detection with short connections. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3203–3212 (2017)
21. Ilg, E., Mayer, N., Saikia, T., Keuper, M., Dosovitskiy, A., Brox, T.: Flownet 2.0: Evolution of optical flow estimation with deep networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2462–2470 (2017)
22. Keuper, M., Tang, S., Andres, B., Brox, T., Schiele, B.: Motion segmentation & multiple object tracking by correlation co-clustering. IEEE transactions on pattern analysis and machine intelligence **42**(1), 140–153 (2018)
23. Kim, C., Li, F., Ciptadi, A., Rehg, J.M.: Multiple hypothesis tracking revisited. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4696–4704 (2015)
24. Kim, C., Li, F., Rehg, J.M.: Multi-object tracking with neural gating using bilinear lstm. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 200–215 (2018)
25. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
26. Kuhn, H.W.: The hungarian method for the assignment problem. Naval research logistics quarterly **2**(1-2), 83–97 (1955)
27. Leal-Taixé, L., Canton-Ferrer, C., Schindler, K.: Learning by tracking: Siamese cnn for robust target association. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 33–40 (2016)
28. Li, Y., Huang, C., Nevatia, R.: Learning to associate: Hybridboosted multi-target tracker for crowded scene. In: 2009 IEEE Conference on Computer Vision and Pattern Recognition. pp. 2953–2960. IEEE (2009)
29. Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C.Y., Berg, A.C.: Ssd: Single shot multibox detector. In: European conference on computer vision. pp. 21–37. Springer (2016)
30. Ma, L., Tang, S., Black, M.J., Van Gool, L.: Customized multi-person tracker. In: Asian Conference on Computer Vision. pp. 612–628. Springer (2018)
31. Mahmoudi, N., Ahadi, S.M., Rahmati, M.: Multi-target tracking using cnn-based features: Cnnmtt. Multimedia Tools and Applications **78**(6), 7077–7096 (2019)
32. Milan, A., Leal-Taixé, L., Reid, I., Roth, S., Schindler, K.: Mot16: A benchmark for multi-object tracking. arXiv preprint arXiv:1603.00831 (2016)
33. Milan, A., Rezatofighi, S.H., Dick, A., Reid, I., Schindler, K.: Online multi-target tracking using recurrent neural networks. In: Thirty-First AAAI Conference on Artificial Intelligence (2017)
34. Milan, A., Roth, S., Schindler, K.: Continuous energy minimization for multitarget tracking. IEEE transactions on pattern analysis and machine intelligence **36**(1), 58–72 (2013)

35. Milan, A., Schindler, K., Roth, S.: Detection-and trajectory-level exclusion in multiple object tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3682–3689 (2013)
36. Nam, H., Han, B.: Learning multi-domain convolutional neural networks for visual tracking. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4293–4302 (2016)
37. Pellegrini, S., Ess, A., Schindler, K., Van Gool, L.: You'll never walk alone: Modeling social behavior for multi-target tracking. In: 2009 IEEE 12th International Conference on Computer Vision. pp. 261–268. IEEE (2009)
38. Pellegrini, S., Ess, A., Van Gool, L.: Improving data association by joint modeling of pedestrian trajectories and groupings. In: European conference on computer vision. pp. 452–465. Springer (2010)
39. Pirsiavash, H., Ramanan, D., Fowlkes, C.C.: Globally-optimal greedy algorithms for tracking a variable number of objects. In: CVPR 2011. pp. 1201–1208. IEEE (2011)
40. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)
41. Reid, D.: An algorithm for tracking multiple targets. IEEE transactions on Automatic Control **24**(6), 843–854 (1979)
42. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. In: Advances in neural information processing systems. pp. 91–99 (2015)
43. Ristani, E., Solera, F., Zou, R., Cucchiara, R., Tomasi, C.: Performance measures and a data set for multi-target, multi-camera tracking. In: European Conference on Computer Vision. pp. 17–35. Springer (2016)
44. Robicquet, A., Sadeghian, A., Alahi, A., Savarese, S.: Learning social etiquette: Human trajectory understanding in crowded scenes. In: European conference on computer vision. pp. 549–565. Springer (2016)
45. Sadeghian, A., Alahi, A., Savarese, S.: Tracking the untrackable: Learning to track multiple cues with long-term dependencies. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 300–311 (2017)
46. Sanchez-Matilla, R., Poiesi, F., Cavallaro, A.: Online multi-target tracking with strong and weak detections. In: European Conference on Computer Vision. pp. 84–99. Springer (2016)
47. Scovanner, P., Tappen, M.F.: Learning pedestrian dynamics from the real world. In: 2009 IEEE 12th International Conference on Computer Vision. pp. 381–388. IEEE (2009)
48. Shen, H., Huang, L., Huang, C., Xu, W.: Tracklet association tracker: An end-to-end learning-based association approach for multi-object tracking. arXiv preprint arXiv:1808.01562 (2018)
49. Sheng, H., Zhang, Y., Chen, J., Xiong, Z., Zhang, J.: Heterogeneous association graph fusion for target association in multiple object tracking. IEEE Transactions on Circuits and Systems for Video Technology **29**(11), 3269–3280 (2018)
50. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
51. Son, J., Baek, M., Cho, M., Han, B.: Multi-object tracking with quadruplet convolutional neural networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 5620–5629 (2017)

52. Tang, S., Andres, B., Andriluka, M., Schiele, B.: Multi-person tracking by multicut and deep matching. In: European Conference on Computer Vision. pp. 100–111. Springer (2016)
53. Tang, S., Andriluka, M., Andres, B., Schiele, B.: Multiple people tracking by lifted multicut and person re-identification. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3539–3548 (2017)
54. Vinyals, O., Toshev, A., Bengio, S., Erhan, D.: Show and tell: A neural image caption generator. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3156–3164 (2015)
55. Wan, X., Wang, J., Kong, Z., Zhao, Q., Deng, S.: Multi-object tracking using online metric learning with long short-term memory. In: 2018 25th IEEE International Conference on Image Processing (ICIP). pp. 788–792. IEEE (2018)
56. Wan, X., Wang, J., Zhou, S.: An online and flexible multi-object tracking framework using long short-term memory. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 1230–1238 (2018)
57. Wang, G., Wang, Y., Zhang, H., Gu, R., Hwang, J.N.: Exploit the connectivity: Multi-object tracking with trackletnet. In: Proceedings of the 27th ACM International Conference on Multimedia. pp. 482–490 (2019)
58. Wang, N., Li, S., Gupta, A., Yeung, D.Y.: Transferring rich feature hierarchies for robust visual tracking. arXiv preprint arXiv:1501.04587 (2015)
59. Wang, X., Türetken, E., Fleuret, F., Fua, P.: Tracking interacting objects using intertwined flows. IEEE transactions on pattern analysis and machine intelligence **38**(11), 2312–2326 (2015)
60. Wojke, N., Bewley, A., Paulus, D.: Simple online and realtime tracking with a deep association metric. In: 2017 IEEE international conference on image processing (ICIP). pp. 3645–3649. IEEE (2017)
61. Xie, S., Tu, Z.: Holistically-nested edge detection. In: Proceedings of the IEEE international conference on computer vision. pp. 1395–1403 (2015)
62. Xu, J., Cao, Y., Zhang, Z., Hu, H.: Spatial-temporal relation networks for multi-object tracking. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 3988–3998 (2019)
63. Yamaguchi, K., Berg, A.C., Ortiz, L.E., Berg, T.L.: Who are you with and where are you going? In: CVPR 2011. pp. 1345–1352. IEEE (2011)
64. Yan, J., Lei, Z., Wen, L., Li, S.Z.: The fastest deformable part model for object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2497–2504 (2014)
65. Yang, B., Nevatia, R.: Multi-target tracking by online learning of non-linear motion patterns and robust appearance models. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1918–1925. IEEE (2012)
66. Yang, B., Nevatia, R.: An online learned crf model for multi-target tracking. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition. pp. 2034–2041. IEEE (2012)
67. Yang, F., Choi, W., Lin, Y.: Exploit all the layers: Fast and accurate cnn object detector with scale dependent pooling and cascaded rejection classifiers. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2129–2137 (2016)
68. Yu, F., Li, W., Li, Q., Liu, Y., Shi, X., Yan, J.: Poi: Multiple object tracking with high performance detection and appearance feature. In: European Conference on Computer Vision. pp. 36–42. Springer (2016)
69. Zhang, J., Zhou, S., Chang, X., Wan, F., Wang, J., Wu, Y., Huang, D.: Multiple object tracking by flowing and fusing. arXiv preprint arXiv:2001.11180 (2020)

70. Zhang, J., Zhou, S., Wang, J., Huang, D.: Frame-wise motion and appearance for real-time multiple object tracking. arXiv preprint arXiv:1905.02292 (2019)
71. Zhang, L., Li, Y., Nevatia, R.: Global data association for multi-object tracking using network flows. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–8. IEEE (2008)
72. Zhang, S., Wang, J., Wang, Z., Gong, Y., Liu, Y.: Multi-target tracking by learning local-to-global trajectory models. Pattern Recognition **48**(2), 580–590 (2015)
73. Zhou, Z., Xing, J., Zhang, M., Hu, W.: Online multi-target tracking with tensor-based high-order graph matching. In: 2018 24th International Conference on Pattern Recognition (ICPR). pp. 1809–1814. IEEE (2018)