Temporal Coherent Object Flow for End-to-End Multi-Object Tracking

Abstract

*Multi-object tracking is a challenging vision task that requires simultaneous reasoning about intra-frame object detection and cross-frame object association. Conventional solutions use frame as the basic unit and typically rely on a motion predictor that exploits the appearance features from the last frame to associate detected candidates, leading to insufficient adaptability to appearance change and long-term associations. In this study, we propose a section-based end-to-end multi-object tracking approach that integrates a temporal coherent **O**bject **F**low tracker (OFTrack), capable of jointly reasoning for both object detection and tracking, as well as achieving simultaneous multi-frame tracking by treating multiple consecutive frames as the basic processing unit, denoted as a "section". The object flow tracker boosts the optical flow to the object flow by employing object perception and section-based motion estimation strategies. Object perception adopts object-aware sampling and scale-aware correlation to enable precise target discrimination. Motion estimation models the correlation of different objects in multiple frames via specialized temporal-spatial attention to achieve robust association in very long videos. Comprehensive experiments on several widely used MOT benchmarks including MOT17, MOT20, KITTI and BDD100k, demonstrate the superior performance of our approach.*

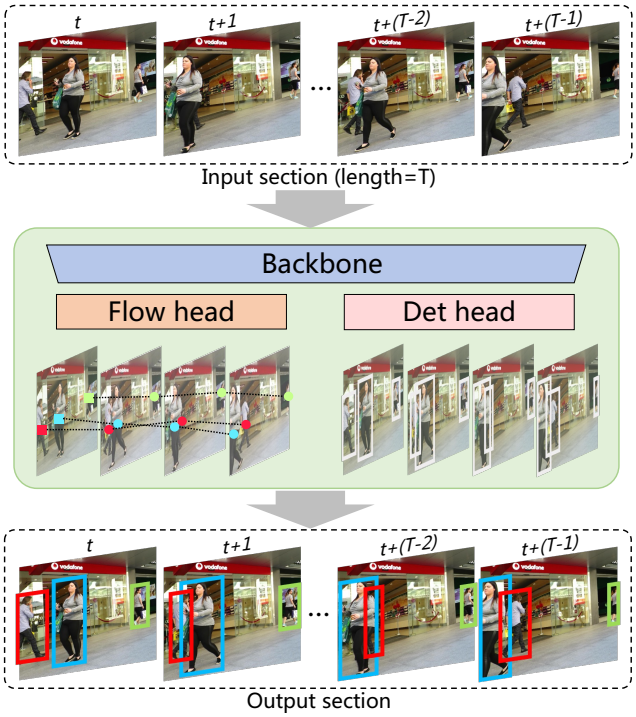


Figure 1. Our OFTrack jointly reasons detection and tracking through a detection head (det head) and a motion estimation head (flow head). The flow head shares the same backbone with the det head and enables simultaneous tracking of target objects across multiple frames within a section.

1. Introduction

Multi-object tracking (MOT) is a challenging vision problem and has many real-world applications [3, 13, 28], such as video surveillance, autonomous driving, robotics and etc. The primary objective of MOT is to identify and track numerous objects of interest in dynamic scenes and to maintain their identities across successive frames. The challenge stems from inherent complexities, including the intricate association in crowded scenes, the visual resemblance between targets, and complex motion patterns.

To address the problems, existing MOT approaches predominantly adhere to two distinct strategies: the two-stage tracking-by-detection methods and the one-stage object query network methods. **Two-stage** tracking-by-detection paradigm [1, 6, 10, 39, 46] first detects objects in the individual video frames using detectors adapted from the object detection research and then relies on a motion tracker to associate the detected objects between consecutive frames. Typically, the motion tracker is a Kalman filter [7] or its modifications, which predicts the trajectories based on previous states. It performs well for smooth and simple motion patterns, but can easily fail in the presence of complex motions and object scale change. In addition, the object tracking and detection processes are relatively independent, despite they have a lot in common. **One-stage** object query network paradigm [9, 12, 29, 35, 45] is a recent research trend that mainly extend the DETR [11, 50] for MOT. They adopt the query-based scheme, represent each target object

as a query, and force to regress the same instance across frames. The object query-based approaches perform implicit inter-frame target association, discard the motion process and can not inherit well from the skills of motion operations [36, 46], culminating in diminished association ability and inferior performances.

In this work, we introduce a new-design temporal coherent object flow tracker (OFTrack), that uses multiple consecutive frames as the basic processing unit, i.e., tracking objects in multiple frames within the section simultaneously. OFTrack exploits an object flow network to provide temporal coherent motion estimation, ensuring robust tracking in long sequences with large intervals. Additionally, our framework effectively associates the tracker and detector by attaching the flow head to the object detector, utilizing the backbone-sharing features of the detector for stronger object awareness, leading to better association.

Our flow head is inspired by the pixel-level optical flow. However, boosting the optical flow to object flow presents several challenges, object flow in tracking tasks requires semantic awareness of the objects rather than pixels, and needs to estimate the motion of objects over an extended period rather than only at an infinitesimal distance in the optical flow. To tackle these challenges, we propose object perception and motion estimation strategies that enhance semantic awareness and adaptability for long-term association. In object perception, we design the object-aware sampling strategy to accurately sample the features centered on tracked targets and their surrounding features. We also employ the scale-aware correlation which uses bidirectional multi-scale processing to generate correlation volumes. In motion estimation, we propose spatial-temporal attention to simultaneously predict the object's motion in multiple frames within a section. Spatial attention interacts the correlation of different objects within a frame, and temporal attention considers the same object across frames for the duration of a section. Our approach accounts for the multi-frame association of objects can significantly improve tracking accuracy. Extensive experiments on several challenging datasets such as MOT17 [30], MOT20 [14], KITTI [19] and BDD100k [44] exhibit state-of-the-art performance among the one-stage trackers, which is also comparable to two-stage approaches.

In summary, our main contributions include:

1. An end-to-end multi-object tracking framework, which attaches a learnable object flow network to the object detector, achieves the jointly reasoning for both object detection and tracking.
2. A temporal coherent object flow network, in which the object perception and motion estimation strategies are introduced to enhance semantic awareness and adaptability for long-term association by simultaneously predicting the object motion in multiple frames.

2. Related Work

Two-stage tracking by detection methods. One of the predominant schemes of multiple object trackers is the tracking-by-detection paradigm [6, 39, 46]. They first predict the object bounding boxes through object detectors [16, 18, 33] for each frame, and then associate the detected objects using a separate motion tracker between consecutive frames. SORT [6] first introduces the Kalman filter to track objects and associates each bounding box with its highest overlapping by the Hungarian algorithm [23]. DeepSORT [39] improves the association in the SORT with motion and deep appearance features. StrongSORT [15] upgrades DeepSORT from detection, embedding, association, and integrates it with lightweight appearance-free algorithms. ByteTrack [46] utilizes the similarities of low confidence detections to tackle the problem of non-negligible missing detection and fragmented trajectories. BoT-SORT [1] adopts camera-motion compensation and designs a more accurate Kalman filter state. P3AFormer [48] adopts pixel-wise distribution architecture and combines with the Kalman filter to enhance the object association. OC-SORT [10] improves the linear motion assumption in the Kalman filter for better adapting the occlusion and non-linear motion.

Unlike the two-stage approach of separately conducting object detection and association, our method achieves multi-object tracking more efficiently by utilizing a joint model for object flow motion and detection.

One-stage end-to-end methods. The one-stage paradigm has been made in a variety of explorations in recent years, which joint detection and association pipeline and aims to convert detectors into trackers to achieve detection and tracking simultaneously in a single stage.

Query-based methods are a recent research trend [9, 17, 27, 29, 45, 47] that mainly extends the DETR [11, 50] for multi-object tracking. These methods represent each target object as a query and force it to regress the same instance across different frames. TrackFormer [29] and MOTR [45] concatenate the object and auto-regressive track query to perform object detection and association simultaneously. TransTrack [35] passes track features cyclically to learn the aggregated embedding of each object. MeMOT [9] preserves a large spatio-temporal memory and uses an attention aggregator to encode past observations. Query-based approaches perform implicit inter-frame target association and offer an end-to-end integrated computational process for the entire model. However, due to the absence of explicit position changes, query-based approaches are unable to extend excellent works based on motion operations [36, 46]. Additionally, a fixed number of queries makes it challenging to detect objects in complex and crowded visual scenes.

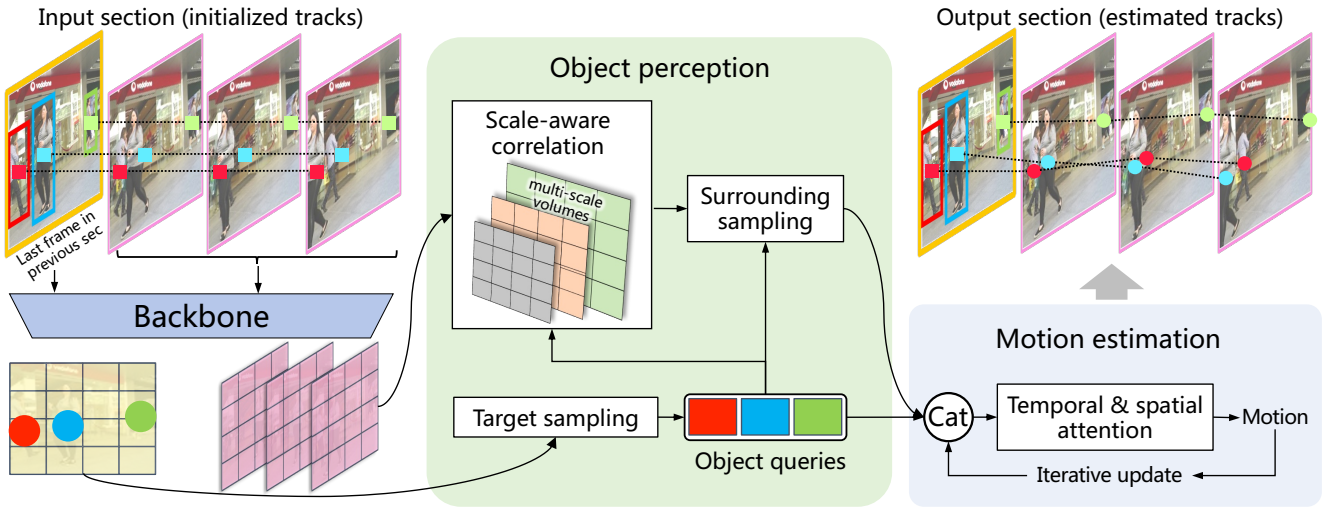


Figure 2. The architecture of our flow head. The flow head comprises two primary modules: object perception and motion estimation. Initially, we extract features from all frames within the current section and sample object queries from the first frame. Note that for continuous association, there is a frame overlap in adjacent sections. Subsequently, we perform scale-aware correlation to obtain the correlation volumes and sample surrounding features from them. Finally, our specialized spatial-temporal attention mechanism iteratively updates the object motion, with the final prediction derived from the last iteration.

Tracking-by-regression methods avoid the association of detections between frames, opting instead to achieve tracking by regressing past object locations to their new positions in the current frame. Tracktor++ [4] exploits the regression head of a detector to perform temporal realignment of object bounding boxes. CenterTrack [49] uses tracking-conditioned detection to localize objects and predict their offsets. In MPNTrack [8], a graph optimization framework based on message-passing networks is combined into a unified tracker. PermaTrack [37] uses ConvGRU[2] to fuse historical memory to reason about the location of the target and learn the occlusion problem. TransCenter [42] adopts dense representations with image-related dense detection queries and sparse tracking queries.

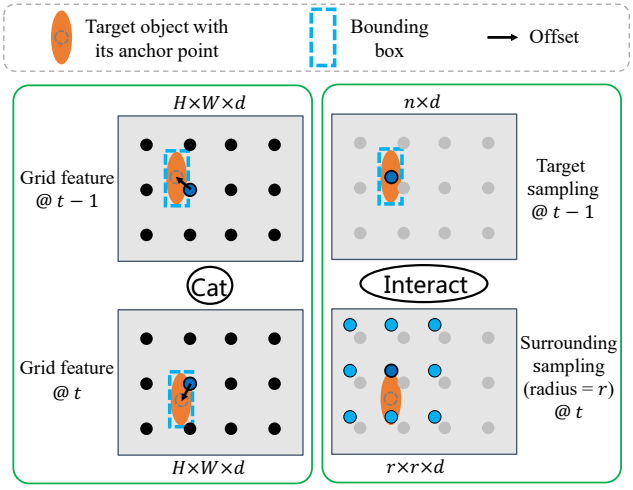
Our approach inherits the architecture from the tracking-by-regression, but we have refrained from using additional graphical optimization or complex motion appearance models. We have designed a temporal coherent object flow that can be integrated into the object detector to form a compact tracking framework. Furthermore, in contrast to the optical flow [36] which predicts the motion of pixels at an infinitesimal distance, we boost the optical flow to the object flow from two aspects: designing the object perception to improve pixel awareness to object awareness, proposing simultaneously motion estimation of multiple frames for long-term association.

3. Method

Our OFTrack, illustrated in Figure 1, consists of three components: a feature extraction backbone, a bounding box regression head (det head), and an object motion prediction head (flow head). We choose the YOLOX [18], a prominent object detector in MOT, as our backbone and det head. The det head detects objects in each frame, while the flow head simultaneously estimates the object motion in multiple frames. Flow head, as illustrated in Figure 2, consists of two parts: **object perception** is designed to improve the semantic awareness of objects. The target sampling generates the object queries of the first frame in the current section. The scale-aware correlation calculates the correlation volumes, and the surrounding sampling extracts the surrounding features from correlation volumes centered around target objects; **motion estimation module** is proposed for simultaneous motion estimation in multiple frames. It concatenates the object quires, surrounding features and object coordinates for attention layers from spatial and temporal dimensions, respectively. The object motion is updated in multiple iterations. Finally, we associate the identical objects based on the Intersection over Union(IoU) similarity.

3.1. Motivation

Our flow head is designed to estimate the object's motion. The goal is to adapt the optical flow to the object flow, which needs to solve two main challenges: (1) **Object awareness**. Our flow head treats the target as a whole for motion estimation, i.e., all pixels inside the object share the same motion pattern. This requires our flow head to be aware of the object semantics of the target and be adaptive to changes in the target scale; (2) **Long-term association**. The optical flow is designed to estimate the pixel motion at an infinitesimal



(a) Grid feature in existing trackers (*w* offset disturbance) (b) Object-aware sampling in ours (*w/o* offset disturbance)

Figure 3. Comparison of features used for motion estimation between traditional trackers and our approaches.

distance across a small number of frames, while our object flow in the tracking task requires a motion estimation over an extended period of video sequence.

To address the above challenges, we developed several strategies. **For object awareness**, we attach the flow head to the object detector and jointly train object tracking and detection in an end-to-end manner. This allows the flow head to obtain the object-aware capability from the detector. Additionally, we design the object perception to perform object-aware sampling that focuses more on the semantic information of objects, and multi-scale correlation to enhance the adaptability to object scale changes. **For long-term association**, we propose the spatial-temporal attention in the motion estimation module to simultaneously predict the object motion of multiple frames across long intervals, ensuring long-term tracking accuracy.

3.2. Object perception

Assuming that we have completed the tracking for the section $k-1$ with the frame length T (frame ids are within the range $[t-T+1, t]$), now our focus is the section k (frame ids are $[t, t+T-1]$). Note that for continuous association, there is an overlap of one frame between adjacent sections, meaning that the last frame of the previous section serves as the first frame of the current section.

Target sampling processes the feature of frame t, given the frame features $f^t \in \mathbb{R}^{d\times H\times W}$, $f^t \in \mathbb{R}^{d\times H\times W}$, and the tracking result in frame t with the number of objects N, the object center points $C_t = (x_i^t, y_i^t), i \in [1, N]$. We conduct bilinear interpolation from f^t at the locations C_t to get the object query $Q_t \in \mathbb{R}^{d\times N}$.

Scale-aware correlation calculates the correlation volumes between frame t and frames in the section k (take one frame $t_i \in [t, t+T-1]$ for clarity) in two steps: (i) expanding the frame feature f^{t_i} to 4 scales $\{f_1^{t_i}, f_2^{t_i}, f_3^{t_i}, f_4^{t_i}\}$, $f_1^{t_i}$ is the scaled up feature at size $2H \times 2W$ using bilinear interpolation, $f_2^{t_i}$ is in the original feature size f^{t_i}, $f_3^{t_i}$ and $f_4^{t_i}$ are with $H/2 \times W/2$ and $H/4 \times W/4$ resolutions using the average pooling; (ii) calculating the scale-aware correlation volumes between the object query in frame t and the all scaled features in frame t_i, the correlation volumes $\mathbf{V}^{t_i}$ are efficiently computed through the matrix multiplication as:

$$V_k^{t_i} = Q_t^T \cdot f_k^{t_i}, \quad V_k^{t_i} \in \mathbb{R}^{N\times H_k\times W_k}, \quad k \in [1, 4] \quad (1)$$

Surrounding sampling processes the correlation volumes of all frames in the section k (take one frame t_i for clarity), according to the location (x, y) centered on the targets in frame t, we extract the surrounding feature $B_k^{t_i} \in \mathbb{R}^{N\times r\times r}$ on the correlation volumes $V_k^{t_i}$ using bilinear interpolation on the meshgrid G(x,y) with the radius r:

$$G(x, y) = \{(x+dx, y+dy) | \, dx, dy \in \mathbb{Z}, \, dx, dy \le r\} \quad (2)$$

Compared to existing trackers [37, 49] that rely on features extracted at grid points for motion estimation, as shown in Figure 3a. These grid features are from the detector's backbone which adopts slide window or patch-based feature extraction schemes, hence the fixed grid points in features are not anchored on tracking targets but with offset instead, leading to inaccurate motion results due to offset disturbance. In contrast, our approach samples feature at the center of each target and then conduct motion estimation, shown in Figure 3b, which therefore resolves the offset disturbance issue.

3.3. Motion estimation

We design the temporal attention and spatial attention for motion estimation, as illustrated in Figure 4. The input tokens are the concatenation of object queries, surrounding features and the object motion, the initial object queries Q_t and object coordinates C_t in frame t are repeated to all frames in the section as $Q \in \mathbb{R}^{N\times T\times d}$ and $C \in \mathbb{R}^{N\times T\times 2}$, the object motion $\eta(C - C_t) \in \mathbb{R}^{N\times T\times d2}$ is obtained by the sinusoidal positional encoding η of coordinates,the surrounding features of radius r extract from the all 4 scaled correlation volumes are $B \in \mathbb{R}^{N\times T\times(r\times r\times 4)}$. The input tokens ($\mathbb{R}^{N\times T\times D}$) are fed into the temporal attention layers first, then reshaped to the $\mathbb{R}^{N\times T\times D}$ to the spatial attention layers, finally the object motion is predicted by a simple linear regression head.

We apply the motion estimation for M times in order to progressively improve the track estimates, that generates a sequence of motion estimation $\{C^1, ..., C^M\}$, the input tokens will be updated by re-sampling the object queries and surrounding features based on the C^i in each iteration. Additionally, we conduct experiments on the effect of iteration

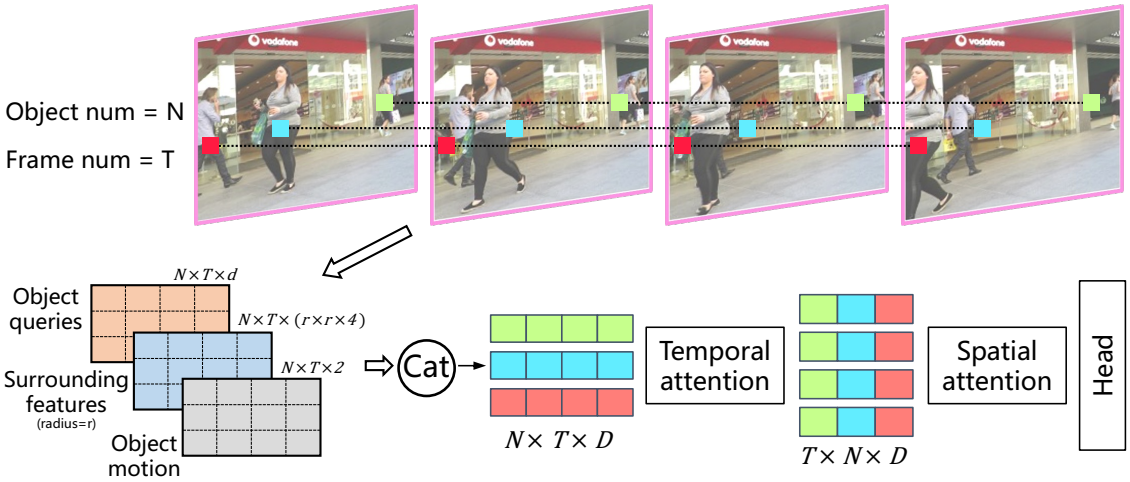


Figure 4. The architecture of the motion estimation module. We estimate the object's motion through temporal attention and spatial attention. Taking a section as the basic process unit, the tracks of all frames are initialized with the location of objects in the first frame. After obtaining the concatenation of object queries, surrounding features and object motion, the temporal attention models the same object across frames, and the spatial attention interacts with the different objects within one frame.

number on motion evaluation and select the appropriate iteration number, $N = 6$, considering both accuracy and efficiency under the MOT scenario.

3.4. Training and inference

In the training phase, we jointly train det head and flow head in an end-to-end manner. Due to the application of a pre-trained YOLOX model [18, 46], we first warm up the flow head for 20 epochs while freezing the parameters of the backbone and det head, to achieve consistent training for both object detection and motion estimation. Our approach takes a section of frames as input, we sample T consecutive frames at random intervals of $[1, 2, 3]$ from training videos, select target objects in the first frame as a reference, and conduct both object detection and motion estimation in the whole section.

For the warm-up of the flow head, we take the L1 distance between the estimation and ground-truth over the full iterations with exponentially increasing weights for each frame t_i:

$$\mathcal{L}_f = \sum_{i}^{M} \sum_{t_i}^{T} \gamma^{M-i} \left| \mathbf{C}_{t_i}^{i} - \mathbf{C}_{t_i}^{gt} \right| \tag{3}$$

where $\{\mathbf{C}^1, ..., \mathbf{C}^M\}$ represent the all iterations of motion estimation, C^{gt} is the ground-truth and we set $\gamma = 0.8$.

For training the whole network in an end-to-end manner, we combine the detector loss $\mathcal{L}_d$ in [18] for each frame with motion estimation loss $\mathcal{L}_f$:

$$\mathcal{L} = \lambda_d \sum_{t_i}^{T} \mathcal{L}_d^{t_i}(B_d^{t_i}, B_{gt}^{t_i}) + \lambda_f \mathcal{L}_f \tag{4}$$

where $\lambda_d = 1.0$ and $\lambda_f = 2.0$ are the weighting factors, B_d indicates estimated detection boxes and B_{gt} means the ground-truth bounding boxes.

In the inference phase, the frames within the section are detected independently and motion estimation is performed simultaneously on all frames. Considering the emergence of new target objects, the process of motion estimation within the flow head is initiated for the subsequent section under the fulfillment of either of the following conditions: reaching the last frame of the section, or detecting a new target in any frame.

4. Experiments

In this section, we verify the individual contributions in the ablation study and present the tracking evaluation on several challenging benchmarks, including MOT17 [30], MOT20 [14], KITTI [19] and BDD100K [44].

4.1. Setting

Implementation Details. For MOT17 [30] and MOT20 [14] that only consist of pedestrians, we adopt the pretrained YOLOX detector from ByteTrack[46] and jointly train the entire model including the flow head on the MOT17 training set and MOT20 training set respectively. For KITTI [19] and BDD100K [44] that are driving scenarios, we adopt the COCO-pretrained YOLOX[18] and use the KITTI training set and the BDD100K training set to train the detector first as [46] and then jointly train the entire model. The training samples are directly sampled from the same sequence within the interval length of 6 frames. The size of an input image is resized to 1440×800. The flow head parameters are initialized with Xavier Uniform. The

GFC	OAS		MOTA	IDF1	HOTA
	Target	Surround			
✓			71.2	32.9	38.5
	-	-	74.4	55.8	51.9
	✓	-	76.1	68.0	60.7
	✓	✓	78.6	71.7	63.3

(a) Comparisons of feature processing. **GFC** means the Grid Feature Concatenation, **OAS** represents the Object-Aware Sampling, **Target** is the Target Sampling, and the **Surround** stands for the Surrounding Sampling.

Scale strategy	No.feat	MOTA	IDF1	HOTA
$[1]$	1	72.6	35.0	40.1
$[1/2, 1]$	2	73.7	52.1	49.6
$[1/4, 1/2, 1]$	3	76.1	70.1	62.0
$[1/8, 1/4, 1/2, 1]$	4	77.8	69.8	62.1
$[1/4, 1/2, 1, 2]$	4	78.6	71.7	63.3

(b) Multiple scale comparison of scale-aware correlation. **No. feat.** represents the total number of feature layers.

T	S	MOTA	IDF1	HOTA
6	-	71.0	53.2	48.2
-	6	73.3	57.5	54.0
4	4	75.3	68.6	58.3
6	6	78.6	71.7	63.3
10	10	78.6	71.8	63.3

(c) Number of Temporal (T) and Spatial (S) attention layers in motion estimation module.

Len	MOTA	IDF1	HOTA
2	76.2	67.5	60.5
4	78.4	71.5	63.2
8	78.6	71.7	63.3
12	76.1	68.4	60.8
16	76.6	66.8	60.0

(d) Sequence length of one section.

iters	MOTA	IDF1	HOTA
1	72.6	66.8	59.9
2	73.7	68.8	60.5
4	77.5	70.3	62.5
6	78.6	71.7	63.3
8	78.6	71.7	63.3
12	78.8	71.8	63.3

(e) Number of iterations for motion estimation.

Table 1. Ablation experiments. The model is trained on the MOT17 train-half and tested on the MOT17 val-half. Default settings are marked in gray. See Section 4.2 for more details.

AdamW [25] optimizer is employed with an initial learning rate of 1e-4 and the learning rate decreases according to the cosine function with the final decrease factor of 0.1. We adopt a warm-up learning rate 1e-5 with a 0.2 warm-up factor on the first 5 epochs. We train our model on 4 Nvidia Tesla V100 GPUs for a total of 80 epochs. The mini-batch size is set to 16 with each GPU hosting 4 batches. Our approach is implemented in Python 3.8 with PyTorch 1.10.

Metrics. We mainly use the Multiple Object Tracking Accuracy (MOTA) [5], Identity F1 Score (IDF1) [34], and Higher Order Tracking Accuracy (HOTA)[26] for evaluation. MOTA provides an overall assessment with a greater emphasis on object detection, IDF1 evaluates the object association for tracking performance, HOTA maintains a balance between object detection and association. We also report other metrics, such as Association Accuracy(AssA), Detection Accuracy (DetA), and ID switch (IDs), for a more comprehensive comparison.

4.2. Ablation Study

We ablate our approach using the MOT17 dataset. MOT17 contains train set and test set, we split the MOT17 train set into train-half set and val-half set as in ByteTrack [46], all of the ablation experiments are trained on train-half and tested on val-half.

Object-aware sampling. We compared the traditional Grid Feature Concatenation (GFC) with our Object-Aware Sampling (OAS) strategy for motion estimation, these two approaches are thoroughly elucidated in Figure 3. From Table 1a we can observe that OAS has a significant superiority over GFC. Furthermore, we conducted a separate evaluation of the target sampling module and surrounding sampling module within the object-aware sampling. Adopting the closet grid feature points to the targets as the object queries and selecting surrounding features from the grid points (line 2) only yields the HOTA of 51.9. The target sampling module effectively improves the performance by increasing the HOTA to 60.7. The highest HOTA score of 63.3 is achieved by adopting both the target sampling module and the surrounding sampling module, which demonstrates the excellent effectiveness of our object-aware sampling strategy.

Scale-aware correlation. Table 1b shows the experimental results on the multi-scale features in the scale-aware correlation, our scaling strategy including up-scale (2x) using the bilinear interpolation and down-scale (1/2x, 1/4x, 1/8x) using the average pooling, we conduct the different numbers of progressive hierarchy, all the scaling layers are combined with the original features (1x) for calculation. From Table 1b we can easily find that increasing the number of scales can steadily improve the performance. Notably, with an equal number of features (4 layers), our object-aware strategy (line 5: HOTA 63.3) outperforms the pooling strategy in the original optical flow (line 4: HOTA 62.1), demonstrating the effectiveness of the proposed scale-aware correlation.

Temporal and spatial attention layers. Attention layers have a large effect on the results of motion estimation, we compare the temporal attention and spatial attention in Ta-

Methods	MOT17 [30]						MOT20 [14]					
	MOTA↑	IDF1↑	HOTA↑	AssA↑	DetA↑	IDs↓	MOTA↑	IDF1↑	HOTA↑	AssA↑	DetA↑	IDs↓
Two-Stage:												
QDTrack [31]	68.7	66.3	53.9	52.7	55.6	3378	/	/	/	/	/	/
TraDeS [40]	69.1	63.9	52.7	50.8	55.2	3555	/	/	/	/	/	/
StrongSORT [15]	79.6	79.5	64.4	64.4	64.6	**1194**	73.8	77.0	62.6	**64.0**	61.3	**770**
OC-SORT [10]	78.0	77.5	63.2	63.4	63.2	1950	75.7	76.3	62.4	62.5	62.4	942
BoT-SORT [1]	80.5	**80.2**	**65.0**	**65.5**	**64.9**	1212	77.8	**77.5**	**63.3**	62.9	**64.0**	1313
Bytetrack [46]	80.3	77.3	63.1	62.0	64.5	2196	77.8	75.2	61.3	59.6	63.4	1223
P3AFormer [48]	**81.2**	78.1	/	/	/	1893	**78.1**	76.4	/	/	/	1332
One-Stage:												
Tracktor++ [4]	56.5	55.1	/	/	/	3763	/	/	/	/	/	/
CenterTrack [49]	67.8	64.7	52.2	51.0	53.8	3039	/	/	/	/	/	/
TransTrack [42]	75.2	63.5	54.1	47.9	61.6	3603	65.0	59.4	48.9	45.2	53.3	3608
PermaTrack [37]	73.8	68.9	55.5	53.1	58.5	3699	/	/	/	/	/	/
TrackFormer [29]	74.1	68.0	57.3	54.1	60.9	2829	68.6	65.7	54.7	53.0	56.7	**1532**
MeMOT [9]	72.5	69.0	56.9	55.2	/	2724	63.7	66.1	54.1	55.0	/	1938
MOTR [45]	71.9	68.4	57.2	55.8	/	**2115**	/	/	/	/	/	/
TransCenter [42]	73.2	62.2	54.5	49.7	60.1	4614	67.7	58.7	/	/	/	3759
MeMOTR [17]	72.8	**71.5**	58.8	**58.4**	59.6	/	/	/	/	/	/	/
OFTrack (ours)	**80.2**	69.1	**59.8**	55.3	**65.2**	3519	**75.3**	**68.6**	**58.3**	**55.0**	**62.2**	2132

Table 2. Performance comparison to state-of-the-art approaches on the MOT17 [30] and MOT20 [14] test set under the private protocol.

ble 1c and observe a poor performance when considering spatial or temporal aspects individually. Although combining these aspects led to a notable performance enhancement, the impact diminished and computational cost substantially increased with deeper layers. Consequently, we opt for a final solution comprising 6 layers for both spatial and temporal attention, totaling 12 layers.

Methods	mMOTA	mIDF1	MOTA	IDF1
QDTrack[31]	36.6	50.8	63.5	**71.5**
TETer[24]	39.1	53.3	/	/
Unicorn[43]	41.2	54.0	66.6	71.3
MOTR[45]	32.3	44.8	56.2	65.8
OFTrack (ours)	**50.1**	**54.5**	**69.6**	67.4

Table 3. Performance comparison to state-of-the-art approaches on the BDD100K[44] MOT validation set.

Sequence length of one section. Simultaneous multi-frame tracking enhances the predictive ability of object flow over long sequences, but excessively long sequences may surpass the precise prediction range. In Table 1d, a length of 2 represents the traditional optical flow approach for predicting adjacent frames, while we adopt a length of 8 which yields the optimal performance.

Iterative motion estimation. We ablate the impact of the different number of iterations for motion estimation. From Table 1e we discover that the performance improvement becomes negligible when the number of iterations exceeds 4, and the impact remains constant after 6 iterations. For optimal accuracy and efficiency, we have chosen 6 iterations in our flow head.

	Methods	HOTA	MOTA	DetA	AssA	IDs↓
Car	IMMDP[41]	68.66	82.75	68.02	69.76	211
	SMAT[20]	71.88	83.64	72.13	72.13	198
	AB3D[38]	69.99	83.61	71.13	69.33	**113**
	CenterTrack[49]	73.02	**88.83**	**75.62**	71.20	254
	TrackMPNN[32]	72.30	87.33	74.69	70.63	481
	QDTrack[31]	68.45	84.93	72.44	65.49	313
	QD-3DT[21]	72.77	85.94	74.09	72.19	206
	Eager[22]	**74.39**	87.82	75.27	74.16	239
	OFTrack (ours)	73.75	87.73	72.62	**77.71**	162
Person	MPNTrack[8]	45.26	46.23	43.74	47.28	397
	AB3D[38]	37.81	38.13	32.37	44.33	181
	CenterTrack[49]	40.35	53.84	44.48	36.93	425
	TrackMPNN[32]	39.40	52.10	44.24	35.45	626
	QDTrack[31]	41.12	55.55	44.81	38.10	487
	QD-3DT[21]	41.08	51.77	44.01	38.82	717
	Eager[22]	39.38	49.82	40.60	38.72	496
	OFTrack (ours)	**47.97**	**58.95**	**44.93**	**53.11**	**141**

Table 4. Performance comparison to state-of-the-art approaches on the KITTI[19] MOT test set.

4.3. State-of-the-art Comparison

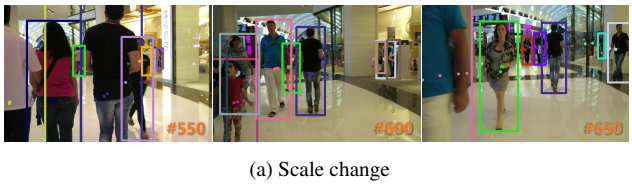
MOT17 and MOT20 are datasets for pedestrian tracking. We use the standard split and obtain the test set evaluation by submitting the results to the online MOTChallenge website. The MOT17 test set contains 2,355 trajectories distributed in 17,757 frames, with an average of 95 objects per frame. MOT20 test set contains 1,501 trajectories across a mere 4,479 frames, with an average of 479 objects per frame, resulting in a much more challenging crowded-scene. The performances are presented in Table 2 under the "private" protocol. As can be seen from the performance comparison, our OFTrack achieves a superior performance both in MOT17 and MOT20 for one-stage methods with the MOTA of 80.2 and 75.3, the HOTA of 59.8 and 58.3, respectively. For the performance gap in the two-stage approaches, one-stage approaches need training dataset in the form of sequence videos rather than images, while the number of training sequences in MOT17 and MOT20 is insufficient to train an end-to-end tracker.

BDD100k is a large driving dataset and the MOT split contains 1400 videos for training and 200 videos for validation. It needs to track objects of 8 classes and contains cases of large camera motion. Table 3 shows that our OFTrack performs better quality and ranks first in mMOTA at 50.1 and mIDF1 at 54.5. Getting an impressive improvement when compared with the previous best results.

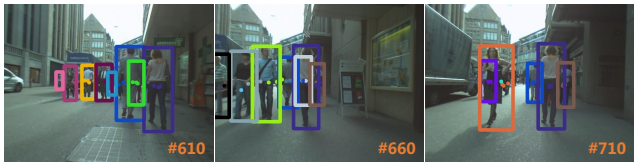
KITTI is a classic tracking benchmark for cars and pedestrians with 21 sequences in the training set and 29 sequences in the testing set. We show the evaluation in Table 4, In terms of car tracking, we achieve a HOTA score of 73.75 alongside a MOTA score of 87.73. Due to the limited number of cars within the training set, the end-to-end approach involving combined target detection and motion estimation necessitates a more extensive dataset. Unfortunately, the current dataset falls short of fulfilling the requisite training criteria. In the context of pedestrian tracking, our OFTrack demonstrates a notable performance advantage over other trackers, achieving a remarkable HOTA score of 47.97 and a MOTA score of 58.95.

4.4. Visualization

We offer visualizations of prototypical challenging scenarios in MOT, including the scale-changing scene (Figure 5a), occlusion scene (Figure 5b) and very crowded scene (Figure 5c), to demonstrate the tracking abilities of the proposed scale-aware and object-aware design. Figure 5 shows the object boxes of the shown frame (frame number in the lower right corner) and center trajectories of the previous three frames. We observe that our OFTrack has a strong discriminative ability for targets with severe scale variations and keeps high reliable associative ability in crowded scenes with dramatic occlusions.



(a) Scale change



(b) Occlusion

(c) Crowded scene

Figure 5. Tracking trajectories visualization of scale change and occlusion in MOT17 [30], crowded scene in MOT20 [14].

5. Conclusion

In this study, we propose a novel end-to-end MOT approach that exploits an temporal coherent object flow to provide motion estimation. We design the object perception and the motion estimation strategies, object perception adopts the object-aware sampling and scale-aware correlation to to enhance the awareness of the object and the adaptability to scale changes, motion estimation models the correlation of different objects in multiple frames via temporal-spatial attention to achieve robust association in very long videos. The scalability of our object flow allows it to be incorporated into most object detectors for object-level motion estimation. Extensive experiments demonstrate the effectiveness of our flow head, and our tracker exhibits impressive performance over previous trackers.

Limitations. Although our OFTrack can effectively estimate the object motion. We observe that our tracker does not adapt well to the non-rigid deformation of the object, e.g., for dancers whose posture and appearance change dramatically. The reason is that our approach treats the target as a rigid object and does not differentiate the information inside it. In future, we intend to integrate our flow head with a deformation estimation counterpart, enhancing its adaptability to a broader range of scenarios.

Temporal Coherent Object Flow for End-to-End Multi-Object Tracking

Supplementary Material

In this supplementary material, we describe more details in the inference phase of our method implementation in Section 6. We carry out a efficiency comparison and a deep analysis of our flow head with the other non-object-aware motion estimation methods in Section 7. In Section 8, we illustrate extensive visual results of the scale-change and very crowded scenarios in MOT17 [30] and MOT20 [14] sequences.

6. Implementation details

Stride size. Our OFTrack processes a simultaneously object motion across multiple frames within a defined section. Notably, the algorithm does not support the addition of new target trajectories within the section. Given the potential appearance of new targets during the inference process, it is not strictly the section length as the stride size, calculating the trajectory of an emerging target is contingent on both the section length and the detection of a new target in any frame within the section. This consideration is reflected in the algorithm outlined in Algorithm 1.

Trajectory initialization. During motion estimation, the input section undergoes initialization with a trajectory, ideally duplicating the object position from the first frame and extending it to all subsequent frames within the section. However, due to our non-strict adherence to the section length as the stride size, overlapping in trajectories may occur when the step length is smaller than the section length. In cases where new targets emerge in a frame, such as t_3 in section k with frames $t_1, t_2, t_3, t_4, t_5, t_6, t_7, t_8$, the subsequent section $k+1$ includes frames $t_3, t_4, t_5, t_6, t_7, t_8, t_9, t_{10}$, resulting in trajectory overlapping in frames $t_3 \sim t_8$. To address this, we utilize the trajectory obtained for frames $t_3 \sim t_8$ in section k as the initialization trajectory for section $k+1$, with the remaining trajectories for frames $t_9 \sim t_{10}$ replicated from the trajectory of frame t_8. A detailed outline of these steps is provided in Algorithm 1.

Algorithm 1: Section-based motion estimation

Input: video frames F, section length S
Output: tracks B

Initialization: $L \leftarrow len(F)$, $i = 0$;
while $i < L$ **do**
  $F_i \leftarrow F[i : i+S]$;
  /* detection at frame level */
  foreach *frame* f *of the* F_i **do**
    append$(D_i, \text{dethead}(f))$;
  if $i = 0$ **then**
    /* initial trajectories are the replication */
    $C_0 \leftarrow \text{boxcenter}(D_i[i])$;
    $T_i \leftarrow \text{repeat}(C_0, S)$;
    $B[0] \leftarrow D_i[i]$;
  else
    /* use the overlapping as the initialized trajectory and complement the remaining trajectories with replication */
    $r = i - last\_i$;
    $T_i[0 : S-r-1] \leftarrow M_i[r : S]$;
    $T_i[S-r : S] \leftarrow \text{repeat}(T_i[-1], r)$;
  /* motion estimation across multiple frames within a section */
  $M_i \leftarrow \text{flowhead}(F_i, T_i)$;
  for $j \leftarrow i+1$ **to** $i+S-1$ **do**
    $B[j] \leftarrow \text{IoUmatch}(M_i[j], D_i[j])$;
    /* assessing the presence of newly appeared objects. */
    if findnew$(B[j], B[j-1])$ **then**
      break;
  $last\_i \leftarrow i$;
  $i \leftarrow i + j$;
  clear (F_i);
  clear (D_i);
  clear (T_i);

7. Additional experiments

Efficiency comparison. Table 5 presents an efficiency comparison among various YOLOX models. Notably, the pretrained YOLOX models utilized in this comparison are trained on the complete MOT17 [30] train set. To mitigate the impact of pre-training data, both training and inference operations are conducted on the MOT17 full train set rather than the half train set. It is essential to recognize that the experimental results in this context serve as a reference for comparing model sizes and do not adhere to the same evaluation standards as other results. The efficiency findings reveal that the lightweight model achieves a higher tracking speed, whereas the larger model demonstrates superior performance.

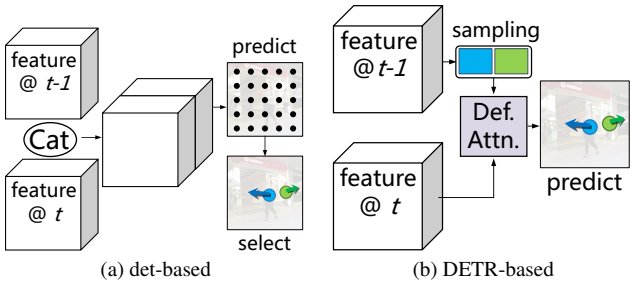
Model	X	L	M	S
MOTA	91.4	90.4	88.7	80.0
IDF1	82.3	79.1	75.0	68.3
HOTA	74.5	72.6	69.9	59.4
Params(M)	125.9	81.0	52.1	35.8
FLOPs(G)	425.5	247.6	132.2	66.0
Speed	11.3*fps*	14.4*fps*	19.8*fps*	25.8*fps*

Table 5. Efficiency comparison on various YOLOX models. The lightweight model achieves a higher tracking speed, the larger model demonstrates superior performance.

Comparison with non-object-aware methods. For a more comprehensive evaluation of our flow head and validate the effectiveness of our object perception module, we compared our object-aware design with two typical non-object-aware schemes, i.e., det-based and DETR-based head, as illustrated in Figure 6

The **det-based** motion estimation head is intuitively based on the object detection head. As shown in Figure 6a, the det-based motion estimation head concatenates the grid features from two consecutive frames and predicts the object motion of all points in grid features. It then identifies the corresponding motion prediction from the nearest grid location of the object. Notably, the location selection does not involve interpolated sampling; instead, it aligns directly with the position in the concurrent detection head. The **DETR-based** motion estimation head is inspired by the detection approach DETR [11], which constitutes the tracking branch of the TransCenter [42]. Depicted in Figure 6b, the DETR-based head samples the target queries from the previous feature and feeds them into the deformable attention [50] with the current frame feature. Then the motion of objects is then predicted from the queries.

The performance of the two motion estimation heads, det-based and DETR-based, is evaluated alongside our flow head on the MOT17 [30] val-half set and BDD100K [44] validation set. Table 6 shows that there is a significant gap in association performance between the det-based head and our flow head (32.9 to 71.7 on IDF1). The det-head is proficient in regressing the absolute location of the object, while the motion requires the relative location of the object between two frames. The concatenation of two features does not enhance the information about their location change but offsets it. The DETR-based head improves the motion estimation accuracy compared to the det-based head, while the performance gap with the flow head is still significant (55.8 to 71.7 on IDF1). Deformable attention interacts with the information between the target query and its surrounding features, and weights the surrounding feature to enhance the target query itself. However, since it doesn't handle the offset disturbance well and doesn't have a robust motion estimation network, it cannot explicitly represent location changes.



(a) det-based (b) DETR-based

Figure 6. The structural of two non-object-aware motion estimation head. The **det-based** head is intuitively inherited from the object detection, it concatenates the grid features from consecutive frames and predicts the object motion of all points in grid features. The **DETR-based** head is inspired by Deformable DETR decoder, it samples the target queries from the previous feature and feeds them into the deformable attention with the current frame feature.

In contrast, the object-aware sampling in our flow head accurately positions itself at the center of each target, effectively addressing offset disturbances in fixed grid points. Additionally, our scale-aware correlation involves multi-scale correlation maps, with each map precisely centered on the target to depict its similarity with the surroundings. Experimental results underscore the effectiveness of our object perception module, as our flow head attains more precise motion estimation for target objects when compared to alternative approaches.

Methods	MOT17			BDD100k			
	MOTA	IDF1	HOTA	mMOTA	mIDF1	MOTA	IDF1
det-based	71.2	32.9	38.5	38.7	45.9	55.2	65.7
DETR-based	74.4	55.8	51.9	42.6	50.6	62.4	64.9
ours	78.6	71.7	63.3	50.1	54.5	69.6	67.4

Table 6. Performance comparisons of motion estimation heads on MOT17 [30] val half set and BDD100K[44] MOT val set.

8. Additional visualizations

Extensive visual results for MOT17 and MOT20 sequences are presented in Figure 7. Figure 7a illustrates scale change scenarios in MOT17-08, while Figure 7b and Figure 7c depict crowded scenes in MOT17-03 and MOT20-6, with Figure 7c featuring a heavily occluded scene. The frame number is displayed in the bottom right corner of each figure, and sequences are presented at intervals of 50 frames. In addition to the bounding box of the current frame, each target in the image displays its tracking trajectory, represented by the center point for the previous three frames.

(a) MOT17-08

(b) MOT17-03

(c) MOT20-06

Figure 7. Tracking trajectories visualization in MOT17 [30] and MOT20 [14]. The frame number is displayed in the bottom right corner of each figure, each target displays its tracking trajectory represented by the center point for the previous 3 frames.

References

[1] Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. Bot-sort: Robust associations multi-pedestrian tracking. *arXiv preprint arXiv:2206.14651*, 2022. 1, 2, 7

[2] Nicolas Ballas, Li Yao, Chris Pal, and Aaron Courville. Delving deeper into convolutional networks for learning video representations. In *Proceedings of the ICLR*, 2016. 3

[3] Mk Bashar, Samia Islam, Kashifa Kawaakib Hussain, Md Hasan, ABM Rahman, Md Kabir, et al. Multiple object tracking in recent times: A literature review. *arXiv preprint arXiv:2209.04796*, 2022. 1

[4] Philipp Bergmann, Tim Meinhardt, and Laura Leal-Taixe. Tracking without bells and whistles. In *Proceedings of the ICCV*, pages 941–951, 2019. 3, 7

[5] Keni Bernardin and Rainer Stiefelhagen. Evaluating multiple object tracking performance: the clear mot metrics. *EURASIP Journal on Image and Video Processing*, 2008:1–10, 2008. 6

[6] 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. 1, 2

[7] Gary Bishop, Greg Welch, et al. An introduction to the kalman filter. *Proc of SIGGRAPH, Course*, 8(27599-23175): 41, 2001. 1

[8] Guillem Brasó and Laura Leal-Taixé. Learning a neural solver for multiple object tracking. In *Proceedings of the CVPR*, pages 6246–6256, 2020. 3, 7

[9] Jiarui Cai, Mingze Xu, Wei Li, Yuanjun Xiong, Wei Xia, Zhuowen Tu, and Stefano Soatto. Memot: multi-object

tracking with memory. In *Proceedings of the CVPR*, pages 8090–8100, 2022. 1, 2, 7

[10] Jinkun Cao, Xinshuo Weng, Rawal Khirodkar, Jiangmiao Pang, and Kris Kitani. Observation-centric sort: Rethinking sort for robust multi-object tracking. *arXiv preprint arXiv:2203.14360*, 2022. 1, 2, 7

[11] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *Proceedings of the ECCV*, pages 213–229. Springer, 2020. 1, 2

[12] Xin Chen, Bin Yan, Jiawen Zhu, Dong Wang, Xiaoyun Yang, and Huchuan Lu. Transformer tracking. In *Proceedings of the CVPR*, pages 8126–8135, 2021. 1

[13] Gioele Ciaparrone, Francisco Luque Sánchez, Siham Tabik, Luigi Troiano, Roberto Tagliaferri, and Francisco Herrera. Deep learning in video multi-object tracking: A survey. *Neurocomputing*, 381:61–88, 2020. 1

[14] Patrick Dendorfer, Hamid Rezatofighi, Anton Milan, Javen Shi, Daniel Cremers, Ian Reid, Stefan Roth, Konrad Schindler, and Laura Leal-Taixé. Mot20: A benchmark for multi object tracking in crowded scenes. *arXiv preprint arXiv:2003.09003*, 2020. 2, 5, 7, 8, 1, 3

[15] Yunhao Du, Yang Song, Bo Yang, and Yanyun Zhao. Strongsort: Make deepsort great again. *arXiv preprint arXiv:2202.13514*, 2022. 2, 7

[16] Kaiwen Duan, Song Bai, Lingxi Xie, Honggang Qi, Qingming Huang, and Qi Tian. Centernet: Keypoint triplets for object detection. In *Proceedings of the ICCV*, pages 6569–6578, 2019. 2

[17] Ruopeng Gao and Limin Wang. Memotr: Long-term memory-augmented transformer for multi-object tracking. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9901–9910, 2023. 2, 7

[18] Zheng Ge, Songtao Liu, Feng Wang, Zeming Li, and Jian Sun. Yolox: Exceeding yolo series in 2021. *arXiv preprint arXiv:2107.08430*, 2021. 2, 3, 5

[19] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. In *Proceedings of the CVPR*, pages 3354–3361. IEEE, 2012. 2, 5, 7

[20] Nicolas Franco Gonzalez, Andres Ospina, and Philippe Calvez. Smat: Smart multiple affinity metrics for multiple object tracking. In *Image Analysis and Recognition: 17th International Conference, ICIAR 2020, Póvoa de Varzim, Portugal, June 24–26, 2020, Proceedings, Part II 17*, pages 48–62. Springer, 2020. 7

[21] Hou-Ning Hu, Yung-Hsu Yang, Tobias Fischer, Trevor Darrell, Fisher Yu, and Min Sun. Monocular quasi-dense 3d object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(2):1992–2008, 2022. 7

[22] Aleksandr Kim, Aljoša Ošep, and Laura Leal-Taixé. Eagermot: 3d multi-object tracking via sensor fusion. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 11315–11321. IEEE, 2021. 7

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

[24] Siyuan Li, Martin Danelljan, Henghui Ding, Thomas E Huang, and Fisher Yu. Tracking every thing in the wild. In *Proceedings of the ECCV*, pages 498–515. Springer, 2022. 7

[25] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Proceedings of the ICLR*, 2018. 6

[26] Jonathon Luiten, Aljosa Osep, Patrick Dendorfer, Philip Torr, Andreas Geiger, Laura Leal-Taixé, and Bastian Leibe. Hota: A higher order metric for evaluating multi-object tracking. *International journal of computer vision*, 129:548–578, 2021. 6

[27] Run Luo, Zikai Song, Lintao Ma, Jinlin Wei, Wei Yang, and Min Yang. Diffusiontrack: Diffusion model for multi-object tracking. *arXiv preprint arXiv:2308.09905*, 2023. 2

[28] Wenhan Luo, Junliang Xing, Anton Milan, Xiaoqin Zhang, Wei Liu, and Tae-Kyun Kim. Multiple object tracking: A literature review. *Artificial intelligence*, 293:103448, 2021. 1

[29] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixe, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. In *Proceedings of the CVPR*, pages 8844–8854, 2022. 1, 2, 7

[30] Anton Milan, Laura Leal-Taixé, Ian Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *arXiv preprint arXiv:1603.00831*, 2016. 2, 5, 7, 8, 1, 3

[31] Jiangmiao Pang, Linlu Qiu, Xia Li, Haofeng Chen, Qi Li, Trevor Darrell, and Fisher Yu. Quasi-dense similarity learning for multiple object tracking. In *Proceedings of the CVPR*, pages 164–173, 2021. 7

[32] Akshay Rangesh, Pranav Maheshwari, Mez Gebre, Siddhesh Mhatre, Vahid Ramezani, and Mohan M Trivedi. Trackmpnn: A message passing graph neural architecture for multi-object tracking. *arXiv preprint arXiv:2101.04206*, 2021. 7

[33] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems*, 28, 2015. 2

[34] Ergys Ristani, Francesco Solera, Roger Zou, Rita Cucchiara, and Carlo Tomasi. Performance measures and a data set for multi-target, multi-camera tracking. In *Proceedings of the ECCV*, pages 17–35. Springer, 2016. 6

[35] Peize Sun, Jinkun Cao, Yi Jiang, Rufeng Zhang, Enze Xie, Zehuan Yuan, Changhu Wang, and Ping Luo. Transtrack: Multiple object tracking with transformer. *arXiv preprint arXiv:2012.15460*, 2020. 1, 2

[36] Zachary Teed and Jia Deng. Raft: Recurrent all-pairs field transforms for optical flow. In *Proceedings of the ECCV*, pages 402–419. Springer, 2020. 2, 3

[37] Pavel Tokmakov, Jie Li, Wolfram Burgard, and Adrien Gaidon. Learning to track with object permanence. In *Proceedings of the ICCV*, pages 10860–10869, 2021. 3, 4, 7

[38] Xinshuo Weng, Jianren Wang, David Held, and Kris Kitani. 3d multi-object tracking: A baseline and new evaluation metrics. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10359–10366. IEEE, 2020. 7

[39] 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. 1, 2

[40] Jialian Wu, Jiale Cao, Liangchen Song, Yu Wang, Ming Yang, and Junsong Yuan. Track to detect and segment: An online multi-object tracker. In *Proceedings of the CVPR*, pages 12352–12361, 2021. 7

[41] Yu Xiang, Alexandre Alahi, and Silvio Savarese. Learning to track: Online multi-object tracking by decision making. In *Proceedings of the ICCV*, pages 4705–4713, 2015. 7

[42] Yihong Xu, Yutong Ban, Guillaume Delorme, Chuang Gan, Daniela Rus, and Xavier Alameda-Pineda. Transcenter: Transformers with dense representations for multiple-object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022. 3, 7, 2

[43] Bin Yan, Yi Jiang, Peize Sun, Dong Wang, Zehuan Yuan, Ping Luo, and Huchuan Lu. Towards grand unification of object tracking. In *Proceedings of the ECCV*, pages 733–751. Springer, 2022. 7

[44] Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *Proceedings of the CVPR*, pages 2636–2645, 2020. 2, 5, 7

[45] Fangao Zeng, Bin Dong, Yuang Zhang, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Motr: End-to-end multiple-object tracking with transformer. In *Proceedings of the ECCV*, pages 659–675, 2022. 1, 2, 7

[46] Yifu Zhang, Peize Sun, Yi Jiang, Dongdong Yu, Fucheng Weng, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *Proceedings of the ECCV*, pages 1–21. Springer, 2022. 1, 2, 5, 6, 7

[47] Yuang Zhang, Tiancai Wang, and Xiangyu Zhang. Motrv2: Bootstrapping end-to-end multi-object tracking by pre-trained object detectors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 22056–22065, 2023. 2

[48] Zelin Zhao, Ze Wu, Yueqing Zhuang, Boxun Li, and Jiaya Jia. Tracking objects as pixel-wise distributions. In *Proceedings of the ECCV*, pages 76–94. Springer, 2022. 2, 7

[49] Xingyi Zhou, Vladlen Koltun, and Philipp Krähenbühl. Tracking objects as points. In *Proceedings of the ECCV*, pages 474–490. Springer, 2020. 3, 4, 7

[50] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020. 1, 2