

Tracking Objects as Pixel-wise Distributions

Zelin Zhao¹^{*}, Ze Wu², Yueqing Zhuang², Boxun Li², and Jiaya Jia^{1,3}^{**}

¹ The Chinese University of Hong Kong

² MEGVII Technology

³ SmartMore

Abstract. Multi-object tracking (MOT) requires detecting and associating objects through frames. Unlike tracking via detected bounding boxes or center points, we propose tracking objects as pixel-wise distributions. We instantiate this idea on a transformer-based architecture named P3AFormer, with pixel-wise propagation, prediction, and association. P3AFormer propagates pixel-wise features guided by flow information to pass messages between frames. Further, P3AFormer adopts a meta-architecture to produce multi-scale object feature maps. During inference, a pixel-wise association procedure is proposed to recover object connections through frames based on the pixel-wise prediction. P3AFormer yields 81.2% in terms of MOTA on the MOT17 benchmark – highest among all transformer networks to reach 80% MOTA in literature. P3AFormer also outperforms state-of-the-arts on the MOT20 and KITTI benchmarks. The code is at <https://github.com/dvlab-research/ECCV22-P3AFormer-Tracking-Objects-as-Pixel-wise-Distributions>.

Keywords: Multi-object Tracking, Transformer, Pixel-wise tracking

1 Introduction

Multi-Object Tracking (MOT) is a long-standing challenging problem in computer vision, which aims to predict the trajectories of objects in a video. Prior work investigates the tracking paradigms [76,2,63,80], optimizes the association procedures [25,75] and learns the motion models [54,52]. Recently, with the powerful transformers deployed in image classification [15,35] and object detection [4,82,35], concurrent work applies transformers to multi-object tracking [68,11,71,51]. Albeit the promising results, we note that the power of the transformer still has much room to explore.

As shown in Figure 1, current transformer-based MOT architectures MOTR [71] and TransCenter [68] still face challenges in detecting small objects and handling occlusions. Representing objects via pixel-wise distributions and conducting pixel-wise associations might alleviate these issues for the following reasons. First, pixel-wise information may help overcome occlusion based on low-level clues [57,44,45]. Moreover, recent transformer architecture demonstrates strong

^{*} The work was done when Zelin Zhao took internship at SmartMore.

^{**} Prof. Jiaya Jia is the corresponding author.



Fig. 1. Comparison of concurrent transformer-based MOT approaches on the MOT17-val dataset. Each row corresponds to one method. (a) MOTR [71] occasionally fails to detect the small objects. (b) TransCenter [68] has a lot of ID-switches (indicated by blue circles) when the small objects are occluded. (c) Our proposed P3AFormer can robustly track the small objects under occlusion.

performance in pixel-wise prediction [35,9,8]. From another perspective, pixel-wise prediction preserves more low-confident details, which can improve tracking robustness [75].

We propose a transformer approach, called P3AFormer, to conduct pixel-wise propagation, prediction, and association. P3AFormer propagates information between frames via the dense feature propagation technique [83], exploiting context information to resist occlusion. To produce robust pixel-level distribution for each object, P3AFormer adopts the meta-architecture [9,8], which generates object proposals and object-centric heatmaps. Masked attention [8] is adopted to pursue localized tracking results and avoid background noise. Further, P3AFormer employs a multi-scale pixel-wise association procedure to recover object IDs robustly. Ablation studies demonstrate the effectiveness of these lines of improvement.

Besides these pixel-wise techniques, we consider a few whistles and bells during the training of P3AFormer. First, we use Hungarian matching [4] (different from direct mapping) to enhance detection accuracy. Second, inspired by the empirical findings of YOLOX [22], we use strong data augmentation, namely mosaic augmentation, in the training procedure of P3AFormer. Further, P3AFormer preserves all low-confident predictions [75] to ensure strong association.

We submit our results to the MOT17 test server and obtain 81.2% MOTA and 78.1% IDF1, outperforming all previous work. On the MOT20 benchmark, we report test accuracy of 78.1% MOTA and 76.4% IDF1, surpassing existing transformer-based work by a large margin. We further validate our approach on the KITTI dataset. It outperforms state-of-the-art methods. Besides, we validate the generalization of the pixel-wise techniques on other MOT frameworks and find that these pixel-wise techniques generalize well to other paradigms.

2 Related Work

2.1 Transformer-based multiple-object tracking

We first discuss concurrent transformer-based MOT approaches. TrackFormer [41] applies the vanilla transformer to the MOT domain and progressively handles newly appeared tracks. TransTrack [51] and MOTR [71] take similar approaches to update the track queries from frames. They explore a tracking-by-attention scheme, while we propose to track objects as pixel-wise distributions. TransCenter [68] conducts association via center offset prediction. It emphasizes a similar concept of dense query representation. However, the model design and tracking schemes are different from ours. TransMOT [11] is a recent architecture to augment the transformer with spatial-temporal graphs. It is noted that our P3AFormer does not use graph attention. We validate different paradigms and model components in experiments to support the design choices of our method.

2.2 Conventional multi-object tracking

The widely used MOT framework is tracking-by-detection [40,2,63,20,53,76,75,1]. DeepSORT [63] leverages the bounding box overlap and appearance features from the neural network to associate bounding boxes predicted by an off-the-shelf detector [47]. Yang et al. [53] propose a graph-based formulation to link detected objects. Other work tries different formulations. For example, Yu et al. [65] formulate MOT into a decision-making problem in Markov Decision Processes (MDP). CenterTrack [80] tracks objects through predicted center offsets. Besides, it is also investigated to reduce post-processing overhead by joint detection and tracking [61,38,5]. Another line of work [46,25,29] leverages graph neural networks to learn the temporal object relations.

2.3 Transformer revolution

Transformer architectures achieved great success in natural language processing (NLP) [56,14]. Recently, transformer demonstrated strong performance in various vision tasks, such as image classification [15,35,55,67], object detection [4,82], segmentation [9,8,66], 3d recognition [60] and pose estimation [32,31]. The seminal work [4] proposes a simple framework DETR for end-to-end object detection. MaskFormer [9] utilizes a meta-architecture to generate pixel embeddings and object proposals via transformers jointly. Previous transformers use masks in attention to restrict attention region [56] or force the computation to be local [35,77].

2.4 Video object detection

Tracking by detection paradigm requires accurate object detection and robust feature learning from videos [64,27]. Zhu et al. [83] propose dense feature propagation to aggregate features from nearby frames. The follow-up work [81] improves aggregation and keyframe scheduling. The MEGA model [7] combines messages from different frames on local and global scales. These methods do not consider video object detection with transformers. TransVOD [79] proposes aggregating the transformer output queries from different frames via a temporal query encoder. TransVOD cannot be directly applied to our setting because it is not an online algorithm and does not make pixel-wise predictions.

2.5 Pixel-wise techniques

Pixel-wise techniques have been proven effective in various applications in computer vision. Dense fusion [57] and pixel-wise voting network [44,18] are proposed to overcome occlusions in the object pose estimation [19]. DPT [45] uses a dense prediction transformer for monocular depth estimation and semantic segmentation. Pyramid vision transformer [59] replaces convolutional neural networks by attention to dense prediction tasks. Yuan et al. [70] presents a high-resolution transformer for human pose estimation. Our P3AFormer, instead, explores the power of pixel-wise techniques in the MOT domain.

3 Pixel-wise Propagation, Prediction and Association

Different from tracking objects via bounding boxes [75,51,1] or as points [68,80], we propose to track objects as pixel-wise distributions. Specifically, P3AFormer first extracts features from each single frame (Sec. 3.1), summarizes features from different frames via pixel-wise feature propagation (Sec. 3.2) and predicts pixel-wise object distributions via an object decoder (Sec. 3.3). The training targets are listed in Sec. 3.4. During inference, P3AFormer conducts pixel-wise association (Sec. 3.5) to build tracks from object distributions.

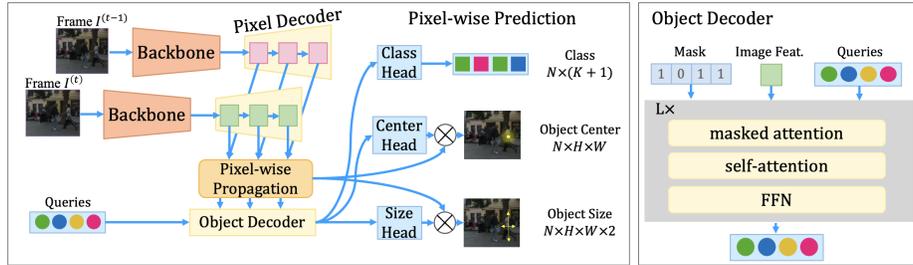


Fig. 2. Diagram of P3AFormer model. **(Left)** The backbone encodes the input images, and the pixel decoder produces pixel-level multi-frame feature embeddings. Then the object decoder predicts latent object features, which are passed through several MLP heads to produce class distribution and the pixel-wise representations for object center and size. **(Right)** The detailed structure of the object decoder. It uses masked attention, self-attention, and feed-forward networks (FFN) to update the query embedding. The add and normalization layers are omitted in this figure for simplicity.

3.1 Single-frame feature extraction

As shown on the top-left of Figure 2, P3AFormer uses a backbone to generate latent features and a pixel decoder to produce pixel-wise heatmaps. The details are as follows.

Backbone. The input to the P3AFormer model is a set of continuous frames from a video. For simplicity and following previous work [80,68], we take two consecutive frames $\mathcal{I}^{(t-1)}$ and $\mathcal{I}^{(t)}$ as input. A backbone generates low-resolution features $\mathbf{F}^{(t)} \in \mathbb{R}^{d \times H_{\mathbf{F}} \times W_{\mathbf{F}}}$ from the input image $\mathcal{I}^{(t)}$, where d is the feature dimension, $H_{\mathbf{F}}$ and $W_{\mathbf{F}}$ are the height and width of the extracted feature maps. Another backbone (sharing weight with the first backbone) extracts the previous-frame feature $\mathbf{F}^{(t-1)}$ from $\mathcal{I}^{(t-1)}$.

Pixel decoder. P3AFormer uses the pixel decoder [9], which is a transformer decoder, to up-sample the features $\mathbf{F}^{(t)}$ and generate per-pixel representation $\mathbf{P}_i^{(t)}$ where i is the current decoding level. The pixel encoder is also applied to the previous-frame feature $\mathbf{F}^{(t-1)}$ to get the pixel-wise feature $\mathbf{P}_i^{(t-1)}$. In our work, we use a multi-scale deformable attention transformer [82] as the pixel decoder.

3.2 Pixel-wise feature propagation

Extracting temporal context from nearby frames is very important in MOT [63,71,80]. We use the pixel-wise flow-guided feature propagation [83] to summarize features between frames (shown in the middle-left of Figure 2). Formally, given a flow network Φ [16], the flow guidance can be represented as $\Phi(\mathcal{I}^{(t-1)}, \mathcal{I}^{(t)})$. After

that, a bilinear warping function \mathcal{W} [83] transforms the previous-frame feature to align with the current feature as

$$\mathbf{P}_l^{(t-1)\rightarrow(t)} = \mathcal{W}(\mathbf{P}_l^{(t-1)}, \Phi(\mathcal{I}^{(t-1)}, \mathcal{I}^{(t)})). \quad (1)$$

We then compute the fused feature as

$$\bar{\mathbf{P}}_l^{(t)} = \mathbf{P}_l^{(t)} + w^{(t-1)\rightarrow(t)} \mathbf{P}_l^{(t-1)\rightarrow(t)}, \quad (2)$$

where the weight $w^{(t-1)\rightarrow(t)}$ is the pixel-wise cosine similarity [83] between the warped feature $\mathbf{P}_l^{(t-1)\rightarrow(t)}$ and the reference feature $\mathbf{P}_l^{(t)}$. The pixel-wise cosine similarity function is provided in the supplementary file for reference. The shape of \mathbf{P}_l is denoted as $H_{P_l} \times W_{P_l} \times d$.

3.3 Pixel-wise predictions

P3AFormer uses a transformer-based object decoder to generate object proposals. The object proposals are combined with the pixel-wise embeddings to get pixel-wise object distributions. As shown in the right part of Figure 2, the object decoder follows the standard transformer [4,56], which transforms N learnable positional embeddings $\mathbf{Q}_l \in \mathbb{R}^{N \times d}$ using L attention layers.

Since an image from an MOT dataset [12,13] often involves a large number of small objects, local features around the objects are often more important than features at long range [76]. Inspired by the recent discovery [8] that masked attention [56,77,8] can promote localized feature learning, P3AFormer uses masked attention in each layer of the object decoder. The mask matrix \mathbf{M}_l is initialized as an all-zero matrix, and it is determined by the center heatmaps at the previous level (presented in Eq. (5)). The standard masked attention [56,77,8] can be denoted as

$$\mathbf{X}_l = \text{softmax}(\mathbf{M}_{l-1} + \mathbf{Q}_l \mathbf{K}_l^T), \mathbf{V}_l + \mathbf{X}_{l-1}, \quad (3)$$

where X_l is the hidden query feature at layer l while the query feature is computed by a linear mapping from the hidden query feature of $\mathbf{Q}_l = f_Q(\mathbf{X}_{l-1})$. The key feature matrix $\mathbf{K}_l \in \mathbb{R}^{H_{P_l} W_{P_l} \times d}$ and the value feature matrix $\mathbf{V}_l \in \mathbb{R}^{H_{P_l} W_{P_l} \times d}$ are derived from the image feature as $\mathbf{K}_l = f_K(\mathbf{P}_l)$, $\mathbf{V}_l = f_V(\mathbf{P}_l)$. These functions f_Q , f_K and f_V are all linear transformations. As shown in the right part of Figure 2, after the masked attention, the hidden query feature passes through the standard self-attention and feed-forward networks. Please refer to [56,9] for these operator details.

At each level, the query embeddings are decoded via three MLP heads to get three embeddings corresponding to object class $\mathcal{E}_l^{cls} = \text{MLP}_{cls}(\mathbf{Q}_l) \in \mathbb{R}^{N \times d}$, the object center $\mathcal{E}_l^{ctr} = \text{MLP}_{ctr}(\mathbf{Q}_l) \in \mathbb{R}^{N \times d}$, and the size of the object $\mathcal{E}_l^{sz} = \text{MLP}_{sz}(\mathbf{Q}_l) \in \mathbb{R}^{N \times d \times 2}$. The hidden dimension of the object size embedding is doubled because the bounding box size is represented in two dimensions (x- and y-axis). Given the full-image representation $\bar{\mathbf{P}}_l$ and the object-centric

embeddings \mathcal{E}_l^{ctr} and \mathcal{E}_l^{sz} , we compute the center heatmaps $c_l \in \mathbb{R}^{N \times H_{P_l} \times W_{P_l}}$ and size maps $s_l \in \mathbb{R}^{N \times H_{P_l} \times W_{P_l} \times 2}$ via dot products as

$$\begin{aligned} c_l[i, h, w] &= \text{sigmoid}(\bar{\mathbf{P}}_l[h, w, :] \cdot \mathcal{E}_l^{ctr}[i, :]), \\ s_l[i, h, w, j] &= \text{sigmoid}(\bar{\mathbf{P}}_l[h, w, :] \cdot \mathcal{E}_l^{sz}[i, :, j]). \end{aligned} \quad (4)$$

After getting the center heatmaps, the attention mask corresponding to i -th object and position (x, y) is updated as

$$\mathbf{M}_{l-1}(i, x, y) = \begin{cases} 0, & \text{if } c_{l-1}[i, x, y] > 0.5, \\ -\infty, & \text{otherwise.} \end{cases} \quad (5)$$

Such a mask restricts the attention to the local region around the object center, much benefiting the tracking performance (as shown in Sec. 4.4).

3.4 Training targets

P3AFormer leverages the bipartite Hungarian matching [4,82] to match N predicted objects to K ground-true objects. During classification, the unmatched object classes are set to an additional category called ‘‘no object’’ (\emptyset). Following MaskFormer [9], we adopt a pixel-wise metric (instead of bounding boxes) in the Hungarian matching.

First, we construct the ground-true heatmap h_l^i for an object via a Gaussian function where the Gaussian center is at the object center, and the Gaussian radius is proportional to the object size [80,68]. Given the predicted center heatmaps \hat{h}_l^i , class distribution \hat{p}_l^i of the i th object, and the corresponding ground true center heatmaps h_l^i and object class c_l^i , we compute the association cost between the prediction and the ground truth via the pixel-wise cost function of

$$\mathcal{L}_{pixel-wise} = \sum_l \sum_i \left(-\log \hat{p}_l^i(c_l^i) + \mathbf{1}_{c_l^i \neq \emptyset} |\hat{h}_l^i - h_l^i| \right). \quad (6)$$

P3AFormer further computes three losses given the matching between predictions and ground-true objects: (1) cross-entropy loss between the predicted and ground-true classes; (2) focal loss [80] between the predicted center heatmaps and ground-true center heatmaps; (3) size loss computed by the $L1$ loss between predicted and ground true size. The final loss is a weighted form of these three losses summarized for all levels.

3.5 Pixel-wise association

After representing objects as a pixel-wise distribution, P3AFormer adopts a pixel association procedure to recover object tracks across frames. This procedure is conducted from the first frame ($t = 0$) to the current frame ($t = T$) in a frame-by-frame manner, which means P3AFormer is an online tracker.

We sketch the association procedure at the timestep $t = T$ in Figure 3. A track τ_k corresponds to an object with a unique ID k . We store into τ_k the

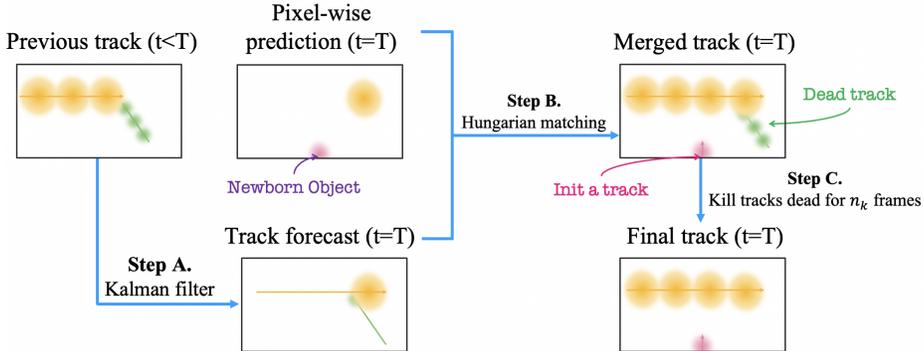


Fig. 3. Pixel-wise association scheme in P3AFormer. One object is represented as a pixel-wise distribution, denoted by spheres with the radial gradient change in this figure. We use one arrow and spheres on the arrow to denote a track. **Step A:** P3AFormer feeds the previous track into the Kalman filter [42] to produce the track forecast. **Step B:** A Hungarian algorithm matches predictions and the track forecast based on a pixel-wise cost function (Eq. (6)). P3AFormer initializes a new track if a newborn object is detected. **Step C:** The dead tracks are removed when an object is occluded or moves out of the image, and the final track is obtained.

bounding boxes $\tau_k.\text{bbox}$ (recovered by the predicted center and size), the score $\tau_k.\text{score}$ (the peak value in the center heatmap), predicted class $\tau_k.\text{class}$ and the center heatmap $\tau_k.\text{heatmap}$.

In **step A**, We use the Kalman Filter [42,75] to predict the location of objects in the current frame ($t = T$) based on previous center locations ($t < T$). The heatmaps are translated along with the forecast movement of the object center via bilinear warping. **Step B** uses the Hungarian algorithm to match the pixel-wise prediction with the track forecast. P3AFormer only matches objects under the same category and omits the “no-object” category. The association cost for Hungarian matching is the L1 distance between a track’s forecast heatmap and an object’s predicted heatmap. We accept a matching if the association cost between the matched track and prediction is smaller than a threshold η_m . A new track $\tau_{k'}$ would be initialized for an untracked object k' if its confidence $\tau_{k'}.\text{score}$ is larger than η_s . In **step C**, the dead tracks that are not matched with any prediction for n_k frames are killed. All the above thresholds η_m , η_s , and η_k are hyper-parameters detailed in Sec. 4.2 and we provide more algorithm details in the supplementary file.

4 Experiments

P3AFormer accomplishes superior results to previous MOT approaches on three public benchmarks. We then ablate each component of the P3AFormer to demonstrate the effectiveness of the pixel-wise techniques. After that, we generalize the proposed pixel-wise techniques to other MOT frameworks.

Table 1. Results on the MOT17 test set. We list transformer-based approaches on bottom of the table and others above. The numbers are from original papers released by authors. **Bold** numbers indicates the best model. We use † to denote unpublished work (prior to ECCV’22) and “W&B” represents whistles and bells.

Methods	MOTA ↑	IDF1 ↑	MT ↑	ML ↓	FP ↓	FN ↓	IDSW ↓
FairMOT [76]	73.7	72.3	19.5	36.6	12201	248047	2072
LSST17 [21]	54.7	62.3	20.4	40.1	26091	228434	1243
Tracktor v2 [1]	56.5	55.1	21.1	35.3	8866	235449	3763
GMOT [29]	50.2	47.0	19.3	32.7	29316	246200	5273
CenterTrack [80]	67.8	64.7	34.6	24.6	18498	160332	3039
QuasiDense [43]	68.7	66.3	40.6	21.9	26589	146643	3378
SiamMOT [49]	65.9	63.3	34.6	23.9	14076	200672	2583
PermaTrack [54]	73.8	68.9	43.8	17.2	28998	115104	3699
CorrTracker [58]	76.5	73.6	47.6	12.7	29808	99510	3369
ByteTrack† [75]	80.3	77.3	53.2	14.5	25491	83721	2196
MOTR† [71]	73.4	68.6	42.9	19.1	27939	119589	2439
TransTrack† [51]	74.5	63.9	46.8	11.3	28323	112137	3663
TransCenter† [68]	73.2	62.2	40.8	18.5	23112	123738	4614
TransMOT† [11]	76.7	75.1	51.0	16.4	36231	93150	2346
P3AFormer	69.2	69.0	34.8	28.8	18621	152421	2769
P3AFormer (+W&B)	81.2	78.1	54.5	13.2	17281	86861	1893

4.1 Datasets

MOT17 [12]. The MOT17 dataset is focused on multiple persons tracking in crowded scenes. It has 14 video sequences in total and seven sequences for testing. The MOT17 dataset is the most popular dataset to benchmark MOT approaches [40,75,63,68,71]. Following previous work [75,80] during validation, we split the MOT17 datasets into two sets. We use the first split for training and the second for validation. We denote this validation dataset as **MOT17-val** for convenience. The best model selected during validation is trained on the full MOT17 dataset and is submitted to the test server under the “private detection” setting. The main metrics are MOTA, IDF1, MT, ML, FP, FN, and IDSW, and we refer the readers to [12] for details of these metrics. For MOT17, we add CrowdHuman [48], Cityperson [73], and ETHZ [17] into the training sets following [75,68]. When training on an image instead of a video with no neighboring frames, the P3AFormer model replicates it and takes two identical images as input.

MOT20 [13]. The MOT20 dataset consists of eight new sequences in crowded scenes. We train on the MOT20 training split with the same hyper-parameters as the MOT17 dataset. We submit our inferred tracks to the public server of MOT20 [13] under the “private detection” protocol. The evaluation metrics are the same as MOT17.

Table 2. Results on the MOT20 test set. The transformer-based approaches are the last three ones. We use **bold** numbers to indicate the best approach. We use † to denote unpublished work (prior to ECCV’22) and “W&B” represents whistles and bells.

Methods	MOTA †	IDF1 †	MT †	ML †	FP †	FN †	IDs †
MLT [74]	48.9	54.6	30.9	22.1	45660	216803	2187
FairMOT [76]	61.8	67.3	68.8	7.6	103440	88901	5243
CorrTracker [58]	65.2	69.1	66.4	8.9	79429	95855	5183
Semi-TCL [30]	65.2	70.1	61.3	10.5	61209	114709	4139
CSTrack [33]	66.6	68.6	50.4	15.5	25404	144358	3196
GSDT [61]	67.1	67.5	53.1	13.2	31913	135409	3131
SiamMOT [49]	67.1	69.1	49.0	16.3	-	-	-
RelationTrack [69]	67.2	70.5	62.2	8.9	61134	104597	4243
SOTMOT [78]	68.6	71.4	64.9	9.7	57064	101154	4209
ByteTrack [75]	77.8	75.2	69.2	9.5	26249	87594	1223
TransTrack [51]	65.0	59.4	50.1	13.4	27197	150197	3608
TransCenter [68]	61.9	50.4	49.4	15.5	45895	146347	4653
P3AFormer	60.3	56.2	50.4	13.5	43221	157564	4522
P3AFormer (+W&B)	78.1	76.4	70.5	7.4	25413	86510	1332

KITTI [23]. The KITTI tracking benchmark contains annotations for eight different classes while only two classes “car” and “pedestrian” are evaluated [80,54]. Twenty-one training sequences and 29 test sequences are presented in the KITTI benchmark. We split the training sequences into halves for training and validation following [80]. Besides the common metrics, KITTI uses an additional metric of HOTA [39] to balance the effect of detection and association.

4.2 Implementation details

Backbone. We mainly use ResNet [26] and Swin-Transformer [35] as the backbone in P3AFormer. For ResNet, we use the ResNet-50 [26] configuration. For Swin-Transformer, we use the Swin-B backbone [35]. We use Swin-B in all final models submitted to the leaderboards and ResNet-50 for validation experiments. The hidden feature dimension is $d = 128$.

Pixel decoder. We adopt the deformable DETR decoder [82] as the multi-scale pixel-wise decoder. Specifically, we use six deformable attention layers to generate feature maps, and the resolutions are the same as Mask2Former [8]. We have in total $L = 3$ layers of feature maps. We add sinusoidal positional and learnable scale-level embedding to the feature maps following [9].

Pixel-wise feature propagation. During feature propagation, we use the simple version of FlowNet [16,83] pre-trained on the Flying Chairs [16] dataset. The generated flow field is scaled to match the resolution of feature maps with bilinear interpolation.

Table 3. Results on the KITTI test split. We show results of two classes “car” and “person” on top and bottom splits. The numbers are from [54]. We use **bold** numbers to indicate the best approach of each class.

Classes	Methods	HOTA \uparrow	MOTA \uparrow	MT \uparrow	PT \downarrow	ML \downarrow
Car	MASS [28]	68.3	84.6	74.0	23.1	2.9
	IMMDP [65]	68.7	82.8	60.3	27.5	12.2
	AB3D [62]	69.8	83.5	67.1	21.5	11.4
	TuSimple [10]	71.6	86.3	71.1	22.0	6.9
	SMAT [24]	71.9	83.6	62.8	31.2	6.0
	TrackMPNN [46]	72.3	87.3	84.5	13.4	2.2
	CenterTrack [80]	73.0	88.8	82.2	15.4	2.5
	PermaTrack [54]	78.0	91.3	85.7	11.7	2.6
	P3AFormer	78.4	91.2	86.5	10.9	2.3
Person	AB3D [62]	35.6	38.9	17.2	41.6	41.2
	TuSimple [10]	45.9	57.6	30.6	44.3	25.1
	TrackMPNN [46]	39.4	52.1	35.1	46.1	18.9
	CenterTrack [80]	40.4	53.8	35.4	43.3	21.3
	PermTrack [54]	48.6	66.0	48.8	35.4	15.8
	P3AFormer	49.0	67.7	49.1	33.2	14.5

Object decoder. The object decoder also has $L = 3$ layers. We adopt $N = 100$ queries, which are initialized as all-zeros and are learnable embeddings [8] during training. No dropout [50] is adopted since it would deteriorate the performance of the meta architecture [8].

Pixel-wise association. The thresholds in the pixel-wise association are $\eta_m = 0.65$ and $\eta_s = 0.80$ on all benchmarks. We found that the P3AFormer model is robust under a wide range of thresholds (see supplementary). The lost tracks are deleted if they do not appear after $n_k = 30$ frames.

Training process. The input image is of shape 1440×800 for MOT17/MOT20 and 1280×384 for KITTI. Following [22,75], we use data augmentation, such as Mosaic [3] and Mixup [72,36]. We use AdamW [37] with an initial learning rate of 6×10^{-5} . We adopt the poly learning rate schedule [6] with weight decay 1×10^{-4} . The full training procedure lasts for 200 epochs. The P3AFormer models are all trained with eight Tesla V100 GPUs. The specific configurations of the losses are provided in the supplementary. The run-time analysis of different models is provided in the supplementary.

4.3 Comparisons on public benchmarks

We first compare the P3AFormer model with several baselines on the MOT17 test sets, and the results are presented in Table 1. With whistles and bells, P3AFormer outperforms all previous approaches on the two major metrics of



Fig. 4. Visualization of center heatmaps and tracking results of P3AFormer on MOT17-val. Each row corresponds to one video. The center heatmaps of different objects are put together into one frame and are best viewed on the screen.

MOTA and IDF1. Besides, P3AFormer surpasses the concurrent unpublished transformer-based approaches by a large margin (4.5% MOTA and 3.0% IDF1). P3AFormer outperforms the strong unpublished baseline ByteTrack [75] while our model differs from theirs. Further, our association procedure does not involve additional training parameters, unlike those of [68,29,71,80].

We also report results on the MOT20 test server in Table 2. Again, P3AFormer demonstrates superior performance with whistles and bells. It outperforms SOTA methods [76,58,49,78] and even the unpublished work [75,69]. Besides, P3AFormer outperforms the concurrent transformer-based work by a large margin (13.1% MOTA and 17.0% IDF1). It achieves the best results on this leaderboard.

A comparison between P3AFormer and the baselines on the KITTI dataset is given in Table 3. Our work outperforms all baselines on two object classes. Notably, P3AFormer surpasses the strong baseline PermaTrack [54] that lever-

Table 4. Comparison of the pixel-wise techniques on MOT17-val. Please refer to Sec. 4.4 for more details.

Methods	mAP \uparrow	MOTA \uparrow	IDF1 \uparrow
w/o All	39.1	68.3	66.8
Pro.	43.5	73.6	73.2
Pre.	42.1	72.8	74.5
Pre.+Ass.	41.9	71.8	74.0
Pro.+Pre.	48.3	69.1	72.3
Pro.+Pre.+Ass.	48.3	78.4	76.0

Table 5. Ablation of the whistles and bells on the MOT17-val (see Sec. 4.5).

Methods	mAP \uparrow	MOTA \uparrow	IDF1 \uparrow
w/o All	46.1	71.3	72.1
w/o Mask.	48.0	71.4	74.7
w/o Mix.	47.8	76.6	74.8
w/o Mosiac	46.7	74.0	71.9
w/o LQ	47.9	77.6	75.1
w/o Bbox	48.3	79.1	74.8
with All	48.3	78.4	76.0

Table 6. Generalization of pixel-wise techniques to other trackers (refer to Sec. 4.6) on the MOT17-val. The validation results of P3AFormer are provided at the bottom for reference.

Methods	MOTA \uparrow	IDF1 \uparrow	MT \uparrow	ML \downarrow	FP \downarrow	FN \downarrow	IDSW \downarrow
Tractor [1]	61.9	64.7	35.3	21.4	323	42454	326
Tractor [1] + Pro.	63.3	67.1	37.5	20.4	310	40930	279
Tractor [1] + Pro. + Pre.	64.6	69.3	38.1	17.9	287	39523	238
Tractor [1] + Pro. + Pre. + Ass.	73.1	72.3	45.0	16.9	224	30000	208
P3AFormer	78.9	76.3	54.3	13.6	216	23462	193

ages additional synthetic training data to overcome occlusions. Intriguingly, our P3AFormer does not need those additional training data.

4.4 Effectiveness of pixel-wise techniques

We decouple the P3AFormer’s pixel-wise techniques and validate the contribution of each part. We use “Pro.” to denote the pixel-wise feature propagation, “Pre.” to denote pixel-wise prediction, and “Ass.” to denote the pixel-wise association. The details of the ablated models are in the supplementary file.

The results are presented in Table 4. We also report the detection mean average precision (mAP [34]). The results are much worse when removing all pixel-wise techniques (the first row of Table 4). Compared to the last row, the incomplete system yields 9.2% mAP, 10.1% MOTA, and 9.2% IDF1 lower results.

When we remove the pixel-wise propagation or pixel-wise prediction (2nd and 3rd rows of Table 4), the results are worse in terms of the detection mAP. Finally, we try different combinations of pixel-wise techniques (4th and 5th rows of Table 4). These combinations improve the tracking performance.

4.5 Influence of training techniques

P3AFormer also incorporates several techniques for training transformers. The results are presented in Table 5. We use “Mask.” to represent mask attention – it

is beneficial to detection (0.3 mAP) and association (1.3% IDF1). We then verify the effect of mixing datasets (CrowdHuman [48], Cityperson [73] and ETHZ [17]) by comparison with only using MOT17 dataset (denoted as “w/o Mix.” in Table 5). It is also clear that using external datasets improves detection and tracking performance. Besides, we notice that using Mosaic augmentation (4th row of Table 5), using learnable query (5th row in Table 5) and connecting all bounding boxes (6th row in Table 5) all slightly improve P3AFormer.

4.6 Generalizing to other trackers

Although our pixel-wise techniques are implemented on the transformer structure, one can apply the pixel-wise techniques to other trackers. We consider the tracking-by-detection tracker Tractor [1], which is based on Faster R-CNN [47] with a camera motion model and a re-ID network.

First, we apply pixel-wise feature propagation to Tractor. Second, we change the output shape of faster-RCNN to predict pixel-wise information. After that, we remove the association procedure of the Tractor and replace it with our dense association scheme. More details are included in this generalization experiment in the supplementary file. The results of the above models are presented in Table 6. It is clear that tracking objects as pixel-wise distributions also improves CNN-based frameworks.

4.7 Visualization of results

Visualization of tracking results in comparison to several transformer-based approaches is provided in Figure 1. The P3AFormer can robustly track small objects without many ID switches. Besides, we provide the visualization of center heatmaps and tracking results of P3AFormer in Figure 4. Even when the objects are heavily occluded, the predicted pixel-wise distribution can provide useful clues to recover the relationship between objects.

5 Conclusion

In this paper, we have presented the P3AFormer, which tracks objects as pixel-wise distributions. First, P3AFormer adopts dense feature propagation to build connections through frames. Second, P3AFormer generates multi-level heatmaps to preserve detailed information from the input images. Finally, the P3AFormer exploits pixel-wise predictions during the association procedure, making the association more robust.

P3AFormer obtains state-of-the-art results on three public benchmarks of MOT17, MOT20, and KITTI. P3AFormer demonstrates strong robustness against occlusion and outperforms concurrent transformer-based frameworks significantly. The proposed pixel-wise techniques can be applied to other trackers and may motivate broader exploration of transformers in future work. We will also study the transformer architecture and make it more efficient to satisfy the real-time tracking requirement.

References

1. Bergmann, P., Meinhardt, T., Leal-Taixe, L.: Tracking without bells and whistles. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 941–951 (2019)
2. Bewley, A., Ge, Z., Ott, L., Ramos, F., Upcroft, B.: Simple online and realtime tracking. In: 2016 IEEE international conference on image processing (ICIP). pp. 3464–3468. IEEE (2016)
3. Bochkovskiy, A., Wang, C.Y., Liao, H.Y.M.: Yolov4: Optimal speed and accuracy of object detection. arXiv preprint arXiv:2004.10934 (2020)
4. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European conference on computer vision. pp. 213–229. Springer (2020)
5. Chaabane, M., Zhang, P., Beveridge, J.R., O’Hara, S.: Deft: Detection embeddings for tracking. arXiv preprint arXiv:2102.02267 (2021)
6. Chen, L.C., Papandreou, G., Kokkinos, I., Murphy, K., Yuille, A.L.: Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. IEEE transactions on pattern analysis and machine intelligence **40**(4), 834–848 (2017)
7. Chen, Y., Cao, Y., Hu, H., Wang, L.: Memory enhanced global-local aggregation for video object detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (June 2020)
8. Cheng, B., Misra, I., Schwing, A.G., Kirillov, A., Girdhar, R.: Masked-attention mask transformer for universal image segmentation. arXiv preprint arXiv:2112.01527 (2021)
9. Cheng, B., Schwing, A., Kirillov, A.: Per-pixel classification is not all you need for semantic segmentation. Advances in Neural Information Processing Systems **34** (2021)
10. Choi, W.: Near-online multi-target tracking with aggregated local flow descriptor. In: Proceedings of the IEEE international conference on computer vision. pp. 3029–3037 (2015)
11. Chu, P., Wang, J., You, Q., Ling, H., Liu, Z.: Transmot: Spatial-temporal graph transformer for multiple object tracking. arXiv preprint arXiv:2104.00194 (2021)
12. Dendorfer, P., Osep, A., Milan, A., Schindler, K., Cremers, D., Reid, I., Roth, S., Leal-Taixé, L.: Motchallenge: A benchmark for single-camera multiple target tracking. International Journal of Computer Vision **129**(4), 845–881 (2021)
13. Dendorfer, P., Rezatofghi, H., Milan, A., Shi, J., Cremers, D., Reid, I., Roth, S., Schindler, K., Leal-Taixé, L.: Mot20: A benchmark for multi object tracking in crowded scenes. arXiv preprint arXiv:2003.09003 (2020)
14. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805 (2018)
15. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al.: An image is worth 16x16 words: Transformers for image recognition at scale. arXiv preprint arXiv:2010.11929 (2020)
16. Dosovitskiy, A., Fischer, P., Ilg, E., Hausser, P., Hazirbas, C., Golkov, V., Van Der Smagt, P., Cremers, D., Brox, T.: Flownet: Learning optical flow with convolutional networks. In: Proceedings of the IEEE international conference on computer vision. pp. 2758–2766 (2015)

17. Ess, A., Leibe, B., Schindler, K., Van Gool, L.: A mobile vision system for robust multi-person tracking. In: 2008 IEEE Conference on Computer Vision and Pattern Recognition. pp. 1–8. IEEE (2008)
18. Fan, Z., Song, Z., Xu, J., Wang, Z., Wu, K., Liu, H., He, J.: Object level depth reconstruction for category level 6d object pose estimation from monocular rgb image (2022)
19. Fan, Z., Song, Z., Xu, J., Wang, Z., Wu, K., Liu, H., He, J.: Object level depth reconstruction for category level 6d object pose estimation from monocular rgb image. arXiv preprint arXiv:2204.01586 (2022)
20. Feichtenhofer, C., Pinz, A., Zisserman, A.: Detect to track and track to detect. In: Proceedings of the IEEE international conference on computer vision. pp. 3038–3046 (2017)
21. 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)
22. Ge, Z., Liu, S., Wang, F., Li, Z., Sun, J.: Yolox: Exceeding yolo series in 2021. arXiv preprint arXiv:2107.08430 (2021)
23. Geiger, A., Lenz, P., Stiller, C., Urtasun, R.: Vision meets robotics: The kitti dataset. *The International Journal of Robotics Research* **32**(11), 1231–1237 (2013)
24. Gonzalez, N.F., Ospina, A., Calvez, P.: Smat: Smart multiple affinity metrics for multiple object tracking. In: International Conference on Image Analysis and Recognition. pp. 48–62. Springer (2020)
25. He, J., Huang, Z., Wang, N., Zhang, Z.: Learnable graph matching: Incorporating graph partitioning with deep feature learning for multiple object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5299–5309 (2021)
26. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
27. Huang, Z., Zhang, T., Heng, W., Shi, B., Zhou, S.: Real-time intermediate flow estimation for video frame interpolation. In: Proceedings of the European Conference on Computer Vision (ECCV) (2022)
28. Karunasekera, H., Wang, H., Zhang, H.: Multiple object tracking with attention to appearance, structure, motion and size. *IEEE Access* **7**, 104423–104434 (2019)
29. Li, J., Gao, X., Jiang, T.: Graph networks for multiple object tracking. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 719–728 (2020)
30. Li, W., Xiong, Y., Yang, S., Xu, M., Wang, Y., Xia, W.: Semi-tcl: Semi-supervised track contrastive representation learning. arXiv preprint arXiv:2107.02396 (2021)
31. Li, W., Liu, H., Ding, R., Liu, M., Wang, P., Yang, W.: Exploiting temporal contexts with strided transformer for 3d human pose estimation. *IEEE Transactions on Multimedia* (2022)
32. Li, W., Liu, H., Tang, H., Wang, P., Van Gool, L.: Mhformer: Multi-hypothesis transformer for 3d human pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 13147–13156 (2022)
33. Liang, C., Zhang, Z., Lu, Y., Zhou, X., Li, B., Ye, X., Zou, J.: Rethinking the competition between detection and reid in multi-object tracking. arXiv preprint arXiv:2010.12138 (2020)
34. Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: European conference on computer vision. pp. 740–755. Springer (2014)

35. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10012–10022 (2021)
36. Liu, Z., Li, S., Wu, D., Chen, Z., Wu, L., Guo, J., Li, S.Z.: Unveiling the power of mixup for stronger classifiers. arXiv preprint arXiv:2103.13027 (2021)
37. Loshchilov, I., Hutter, F.: Decoupled weight decay regularization. arXiv preprint arXiv:1711.05101 (2017)
38. Lu, Z., Rathod, V., Votel, R., Huang, J.: Retinatrack: Online single stage joint detection and tracking. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 14668–14678 (2020)
39. Luiten, J., Osep, A., Dendorfer, P., Torr, P., Geiger, A., Leal-Taixé, L., Leibe, B.: Hota: A higher order metric for evaluating multi-object tracking. *International journal of computer vision* **129**(2), 548–578 (2021)
40. Luo, W., Xing, J., Milan, A., Zhang, X., Liu, W., Kim, T.K.: Multiple object tracking: A literature review. *Artificial Intelligence* **293**, 103448 (2021)
41. Meinhardt, T., Kirillov, A., Leal-Taixe, L., Feichtenhofer, C.: Trackformer: Multi-object tracking with transformers. arXiv preprint arXiv:2101.02702 (2021)
42. Meinhold, R.J., Singpurwalla, N.D.: Understanding the kalman filter. *The American Statistician* **37**(2), 123–127 (1983)
43. Pang, J., Qiu, L., Li, X., Chen, H., Li, Q., Darrell, T., Yu, F.: Quasi-dense similarity learning for multiple object tracking. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 164–173 (2021)
44. Peng, S., Liu, Y., Huang, Q., Zhou, X., Bao, H.: Pvnet: Pixel-wise voting network for 6dof pose estimation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 4561–4570 (2019)
45. Ranftl, R., Bochkovskiy, A., Koltun, V.: Vision transformers for dense prediction. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 12179–12188 (2021)
46. Rangesh, A., Maheshwari, P., Gebre, M., Mhatre, S., Ramezani, V., Trivedi, M.M.: Trackmpnn: A message passing graph neural architecture for multi-object tracking. arXiv preprint arXiv:2101.04206 (2021)
47. Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. *Advances in neural information processing systems* **28** (2015)
48. Shao, S., Zhao, Z., Li, B., Xiao, T., Yu, G., Zhang, X., Sun, J.: Crowdhuman: A benchmark for detecting human in a crowd. arXiv preprint arXiv:1805.00123 (2018)
49. Shuai, B., Berneshawi, A., Li, X., Modolo, D., Tighe, J.: Siammot: Siamese multi-object tracking. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 12372–12382 (2021)
50. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research* **15**(1), 1929–1958 (2014)
51. Sun, P., Cao, J., Jiang, Y., Zhang, R., Xie, E., Yuan, Z., Wang, C., Luo, P.: Transtrack: Multiple object tracking with transformer. arXiv preprint arXiv:2012.15460 (2020)
52. Sun, S., Akhtar, N., Song, X., Song, H., Mian, A., Shah, M.: Simultaneous detection and tracking with motion modelling for multiple object tracking. In: European Conference on Computer Vision. pp. 626–643. Springer (2020)

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. Tokmakov, P., Li, J., Burgard, W., Gaidon, A.: Learning to track with object permanence. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10860–10869 (2021)
55. Tu, Z., Talebi, H., Zhang, H., Yang, F., Milanfar, P., Bovik, A., Li, Y.: Maxvit: Multi-axis vision transformer. arXiv preprint arXiv:2204.01697 (2022)
56. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. *Advances in neural information processing systems* **30** (2017)
57. Wang, C., Xu, D., Zhu, Y., Martín-Martín, R., Lu, C., Fei-Fei, L., Savarese, S.: Densefusion: 6d object pose estimation by iterative dense fusion. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 3343–3352 (2019)
58. Wang, Q., Zheng, Y., Pan, P., Xu, Y.: Multiple object tracking with correlation learning. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3876–3886 (2021)
59. Wang, W., Xie, E., Li, X., Fan, D.P., Song, K., Liang, D., Lu, T., Luo, P., Shao, L.: Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV). pp. 568–578 (October 2021)
60. Wang, X., Zhu, Z., Qin, F., Ye, Y., Huang, G., Chi, X., He, Y., Wang, X.: Mvster: Epipolar transformer for efficient multi-view stereo. arXiv preprint arXiv:2204.07346 (2022)
61. Wang, Y., Kitani, K., Weng, X.: Joint object detection and multi-object tracking with graph neural networks. In: 2021 IEEE International Conference on Robotics and Automation (ICRA). pp. 13708–13715. IEEE (2021)
62. Weng, X., Kitani, K.: A baseline for 3d multi-object tracking. arXiv preprint arXiv:1907.03961 **1**(2), 6 (2019)
63. 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)
64. Wu, W., Zhang, D., Fu, Y., Shen, C., Zhou, H., Cai, Y., Luo, P.: End-to-end video text spotting with transformer. arXiv preprint arXiv:2203.10539 (2022)
65. Xiang, Y., Alahi, A., Savarese, S.: Learning to track: Online multi-object tracking by decision making. In: Proceedings of the IEEE international conference on computer vision. pp. 4705–4713 (2015)
66. Xie, E., Wang, W., Yu, Z., Anandkumar, A., Alvarez, J.M., Luo, P.: Segformer: Simple and efficient design for semantic segmentation with transformers. *Advances in Neural Information Processing Systems* **34** (2021)
67. Xu, R., Xiang, H., Tu, Z., Xia, X., Yang, M.H., Ma, J.: V2x-vit: Vehicle-to-everything cooperative perception with vision transformer. arXiv preprint arXiv:2203.10638 (2022)
68. Xu, Y., Ban, Y., Delorme, G., Gan, C., Rus, D., Alameda-Pineda, X.: Transcenter: Transformers with dense queries for multiple-object tracking. arXiv preprint arXiv:2103.15145 (2021)
69. Yu, E., Li, Z., Han, S., Wang, H.: Relationtrack: Relation-aware multiple object tracking with decoupled representation. *IEEE Transactions on Multimedia* (2022)

70. Yuan, Y., Fu, R., Huang, L., Lin, W., Zhang, C., Chen, X., Wang, J.: Hrformer: High-resolution transformer for dense prediction. arXiv preprint arXiv:2110.09408 (2021)
71. Zeng, F., Dong, B., Wang, T., Zhang, X., Wei, Y.: Motr: End-to-end multiple-object tracking with transformer. arXiv preprint arXiv:2105.03247 (2021)
72. Zhang, H., Cisse, M., Dauphin, Y.N., Lopez-Paz, D.: mixup: Beyond empirical risk minimization. arXiv preprint arXiv:1710.09412 (2017)
73. Zhang, S., Benenson, R., Schiele, B.: Citypersons: A diverse dataset for pedestrian detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 3213–3221 (2017)
74. Zhang, Y., Sheng, H., Wu, Y., Wang, S., Ke, W., Xiong, Z.: Multiplex labeling graph for near-online tracking in crowded scenes. IEEE Internet of Things Journal **7**(9), 7892–7902 (2020). <https://doi.org/10.1109/JIOT.2020.2996609>
75. Zhang, Y., Sun, P., Jiang, Y., Yu, D., Yuan, Z., Luo, P., Liu, W., Wang, X.: Bytetrack: Multi-object tracking by associating every detection box. arXiv preprint arXiv:2110.06864 (2021)
76. Zhang, Y., Wang, C., Wang, X., Zeng, W., Liu, W.: Fairmot: On the fairness of detection and re-identification in multiple object tracking. International Journal of Computer Vision **129**(11), 3069–3087 (2021)
77. Zhao, Z., Samel, K., Chen, B., et al.: Proto: Program-guided transformer for program-guided tasks. Advances in Neural Information Processing Systems **34** (2021)
78. Zheng, L., Tang, M., Chen, Y., Zhu, G., Wang, J., Lu, H.: Improving multiple object tracking with single object tracking. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2453–2462 (2021)
79. Zhou, Q., Li, X., He, L., Yang, Y., Cheng, G., Tong, Y., Ma, L., Tao, D.: Transvod: End-to-end video object detection with spatial-temporal transformers. arXiv preprint arXiv:2201.05047 (2022)
80. Zhou, X., Koltun, V., Krähenbühl, P.: Tracking objects as points. In: European Conference on Computer Vision. pp. 474–490. Springer (2020)
81. Zhu, X., Dai, J., Yuan, L., Wei, Y.: Towards high performance video object detection. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 7210–7218 (2018)
82. Zhu, X., Su, W., Lu, L., Li, B., Wang, X., Dai, J.: Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159 (2020)
83. Zhu, X., Wang, Y., Dai, J., Yuan, L., Wei, Y.: Flow-guided feature aggregation for video object detection. In: Proceedings of the IEEE international conference on computer vision. pp. 408–417 (2017)