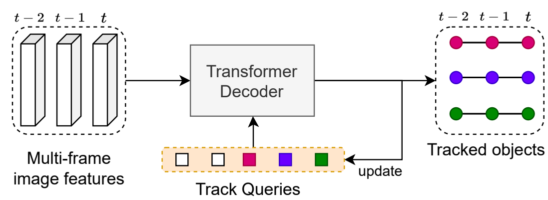
ADA-Track: End-to-End Multi-Camera 3D Multi-Object Tracking with Alternating Detection and Association

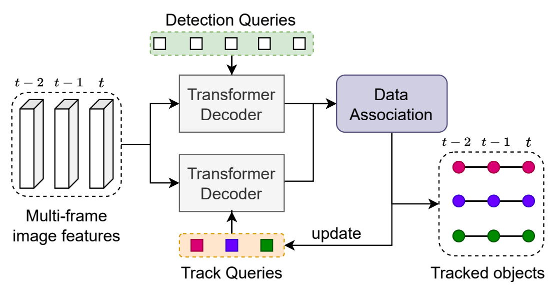
Shuxiao Ding[1,2], Lukas Schneider[1], Marius Cordts[1], Juergen Gall[2,3]
[1]Mercedes-Benz AG, Sindelfingen, [2]University of Bonn
[3]Lamarr Institute for Machine Learning and Artificial Intelligence
{shuxiao.ding,lukas.schneider,marius.cordts}@mercedes-benz.com, gall@iai.uni-bonn.de

Abstract

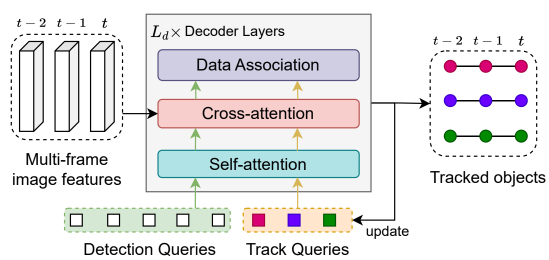
Many query-based approaches for 3D Multi-Object Tracking (MOT) adopt the tracking-by-attention paradigm, utilizing track queries for identity-consistent detection and object queries for identity-agnostic track spawning. Tracking-by-attention, however, entangles detection and tracking queries in one embedding for both the detection and tracking task, which is sub-optimal. Other approaches resemble the tracking-by-detection paradigm, detecting objects using decoupled track and detection queries followed by a subsequent association. These methods, however, do not leverage synergies between the detection and association task. Combining the strengths of both paradigms, we introduce ADA-Track, a novel end-to-end framework for 3D MOT from multi-view cameras. We introduce a learnable data association module based on edge-augmented cross-attention, leveraging appearance and geometric features. Furthermore, we integrate this association module into the decoder layer of a DETR-based 3D detector, enabling simultaneous DETR-like query-to-image cross-attention for detection and query-to-query cross-attention for data association. By stacking these decoder layers, queries are refined for the detection and association task alternately, effectively harnessing the task dependencies. We evaluate our method on the nuScenes dataset and demonstrate the advantage of our approach compared to the two previous paradigms. Code is available at https://github.com/dsx0511/ADA-Track.



(a) **Tracking-by-attention** (TBA).



(b) **Tracking-by-detection** (TBD).



(c) **Our proposed paradigm.**

Figure 1. Paradigm comparison of query-based end-to-end MOT. Our proposed paradigm (1c) leverages the advantages of the coupled architecture of tracking-by-attention (1a) and the decoupled task-specific queries of tracking-by-detection (1b).

1. Introduction

Accurate and consistent 3D Multi-Object Tracking (MOT) is critical for ensuring the reliability and safety of autonomous driving. Recently, vision-centric perception solely relying on multi-view cameras has garnered significant attention in the autonomous driving community, thanks to lower cost of sensors and the advancements of transformers for computer vision. Within this domain, two predominant approaches have emerged: one transforms multi-view features into an intermediate dense Bird's-Eye View (BEV) representation [23, 26, 31, 35, 64], while the other leverages object queries [8] that directly interact with the multi-view images [42, 51, 58, 61, 66] to construct an object-centric representation. Owing to the advantages in modeling object motion, the latter has been extended to query-based MOT in many works [42, 51, 58, 61, 66].

Among the query-based MOT approaches, the majority adopts the *tracking-by-attention* (TBA) paradigm [42, 61]. As illustrated in Figure 1a, TBA utilizes track queries (colored squares) to consistently detect the same identity across frames and introduces object queries (white squares) to initialize tracks for newly appearing objects in each frame. However, this highly entangled design is sub-optimal for balancing the detection and tracking performance. Firstly, each track query consisting of a single embedding is tasked with accomplishing both, detection and tracking, while the two tasks share the same network architecture. Furthermore, track queries for identity-aware tracking and object queries for identity-agnostic detection are also processed by identical network weights. We argue that such an approach is sub-optimal for extracting task-specific information from the single query representation. Secondly, data association is implicitly addressed using self-attention between all queries. Although it effectively integrates information of query relations into query refinement, a notable drawback arises during inference. The network only outputs one confidence score for each object, but it is unclear whether it represents a detection or an association confidence. This requires sophisticated manually tuned post-processing.

Other query-based MOT approaches [30, 34, 51] use decoupled detection and track queries to solve detection and tracking tasks independently. Both query types will be associated explicitly in a heuristic or learnable module, as shown in Figure 1b. However, this still inherits the decoupled design of the *tracking-by-detection* (TBD) paradigm and struggles to optimize and harmonize both tasks effectively.

In this paper, we argue that detection and tracking is a chicken-and-egg problem: accurate detection enables robust initialization and straightforward association to tracks, while well-established tracks incorporate temporal context to mitigate potential detection errors. Our method elegantly addresses this challenge by leveraging synergies in both tasks while decoupling them. We propose ADA-Track, a novel query-based end-to-end multi-camera 3D MOT framework that conducts object detection and explicit association in an alternating manner, as shown in Figure 1c. We propagate track queries across frames representing a unique object instance, while generating decoupled detection queries that detect all objects in each frame. Inspired by [16], we propose a learned data association module based on an edge-augmented cross-attention [27]. In this module, edge features between track and detection queries represent association information. These features are incorporated into attention calculations, updated layer-by-layer, and further used to output affinity scores. Different from [16], we include appearance features in the nodes and geometric features in the edges, resulting in a fully differentiable appearance-geometric reasoning. We then integrate the learned association module into each transformer decoder layer of a query-based multi-camera 3D detector, *e.g.* DETR3D [55]. In this way, the decoder layer sequentially conducts a query-to-image cross-attention to refine query representations for object detection and a query-to-query cross-attention to refine query and edge representations for data association. By stacking the decoder layers, iteratively refined query and edge features provide useful information to each other, resulting in a harmonized optimization of the detection and tracking task.

We evaluate our method on the nuScenes dataset [6] and compare our proposed alternating detection and association paradigm with approaches based on the other two paradigms. While achieving state-of-the-art performance, our proposed paradigm can easily be combined with various query-based 3D detectors.

2. Related Work

Multi-camera 3D detection Current research on multi-camera 3D object detection falls into two major categories. The first category transforms multi-view image features into a dense Bird's-Eye View (BEV) representation using CNNs [25, 26, 33, 48] or transformers [35, 59]. Although it has been demonstrated that temporal fusion in BEV effectively boosts the detection performance [25, 35, 47] or supports downstream tasks [23, 31, 64], such a structured representation may struggle to effectively model object motion. The second category contains works that follow DETR [8], where sparse object queries interact with multi-view images [17, 40, 55]. This sparse query-based representation facilitates effective object-centric temporal fusion by interacting queries with multi-frame sensor data [38] or propagating queries across frames [39, 54]. Consequently, end-to-end detection and tracking methods built on top of query-based detectors have emerged as a popular choice.

Tracking-by-attention Proposed concurrently by TrackFormer [42] and MOTR [61], tracking-by-attention (TBA) leverages track queries to detect objects and simultaneously maintain their consistent identities across frames. TBA regards MOT as a multi-frame set prediction problem, which relies on the self-attention for implicit association and a bipartite matching to force identity-consistent target assignment, similar to a learned duplicate removal [8, 15, 22]. MUTR3D [62] extends the tracking-by-attention paradigm to multi-camera 3D MOT based on a DETR3D [55] detector, which additionally updates the 3D reference point of each query besides query feature propagation. STAR-Track [18] improves MUTR3D by proposing a Latent Motion Model (LMM) to update the query appearance feature based on geometric motion prediction. PF-Track [46] proposes a joint tracking and prediction framework, utilizing a memory bank of queries to refine queries and predict future locations over an extended horizon for occlusion handling. Despite the fully differentiable design, TBA processes the same query representation using shared network weights for both detection and tracking tasks, which inevitably affects the balance of both tasks. In this work, we address this problem by introducing decoupled task-dependent queries with a differentiable association module, while an alternating optimization strongly couples both tasks in a more reasonable manner.

Tracking by detection Many tracking-by-detection (TBD) approaches use a standalone algorithm for data association that can be combined with arbitrary object detectors. SORT [4], for instance, uses a Kalman Filter as the motion model and associates the objects using Hungarian Matching [28]. Subsequent works [1, 7, 19, 57, 63] improve SORT and achieve competitive performance in many 2D MOT benchmarks [14, 43]. Similar pipelines for 3D MOT [2, 11, 12, 32, 45, 52, 53, 56, 65] have also demonstrated promising performance. Besides model-based approaches, learning-based methods usually formulate the association problem using a graph structure and solve it using GNNs [5, 10, 44, 49, 50, 60] or transformers [13, 15, 67]. TBD in a joint detection and tracking framework has also become a popular choice combined with query-based detectors [8, 55]. These approaches usually decouple detection and track queries, process them independently, and associate them based on IoU [51], box center [58], pixel-wise distribution [66] or a learned metric [30, 34]. Although these works are end-to-end trainable, the association module is still separated from the upstream detector, and detection and tracking are processed sequentially, limiting the effective utilization of task dependencies. In our work, we address this problem by stacking detection and association modules in an alternating fashion. In doing so, we utilize the synergies between both tasks.

3. Approach

An overview of ADA-Track is shown in Figure 2. For each frame t, feature maps F_c^t are extracted from multi-view images I_c^t using a CNN for each camera c. A set of *track queries* Q_{T}^t (depicted as colored squares in Figure 2) are propagated from the previous frame in order to consistently detect the same identity across frames. A fixed number of N_D *detection queries* Q_{D}^t (white squares in Figure 2) is randomly initialized and responsible to detect all objects in the current frame. Following recent works, we assign a 3D reference to each of those queries [40, 55]. The transformer decoder layer first conducts a *self-attention* between queries and an *image-to-query cross-attention* (*e.g.* DETR3D [55] or PETR [40]) to refine queries for the object detection task. Subsequently, a *query-to-query edge-augmented cross-attention* integrates both query types and edge features and refines them for the data association task. The overall transformer decoder layer is repeated L_d times to alternately refine query and edge features for both detection and association task. Finally, a track update module associates the track and detection queries and generates track queries Q_{T}^{t+1} for the next frame.

3.1. Joint detection and association decoder layer

Next, we discuss the decoding process for a single frame t and omit the notation of the frame index t for simplicity.

Query-to-query self-attention Existing approaches with decoupled queries [30, 34, 51] process track and detection queries independently and conduct self-attention only within the same query type. In contrast, our approach seeks a joint optimization of the representations of both query types. We concatenate both Q_{T} and Q_{D} and apply self-attention among all queries regardless of their type. This self-attention enables detection queries to leverage track queries as prior information, leading to a more targeted interaction with image features in the subsequent layer.

Image-to-query cross-attention Next, both query types Q_{T} and Q_{D} attend to the multi-view image features F_c with cross-attention. Our approach is compatible with various sparse query-based 3D detectors, thus cross-attention can be implemented in multiple fashions, *e.g.* DETR3D [55] or PETR [40]. The interaction between queries and images refines the query features to form an object-centric representation. Subsequently, the network predicts bounding boxes and category confidences using MLPs for each query. This results in track boxes $B_{\mathrm{T}}^{(l)}$ from track queries and detection boxes $B_{\mathrm{D}}^{(l)}$ from detection queries, where l denotes the layer index of the decoder layer. Each box $b_i = [c_i, s_i, \theta_i, v_i] \in \mathbb{R}^9$ is parameterized by 3D box center $c_i \in \mathbb{R}^3$, 3D box size $s_i \in \mathbb{R}^3$, yaw angle $\theta_i \in \mathbb{R}$ and BEV velocity $v_i \in \mathbb{R}^2$. Both sets of boxes are used to calculate auxiliary box losses as well as to build the position encoding for the edge features that we will discuss next.

Query-to-query edge-augmented cross-attention Our association module requires a differentiable and lightweight architecture that can be integrated into each decoder layer. To this end, we opt for a learned association module that was recently proposed by 3DMOTFormer [16]. The module is based on an Edge-Augmented Graph Transformer [27] to learn the affinity between tracks and detections. Different from [16], the query positions change across decoder layers when refining them iteratively in the joint detection and tracking framework. Therefore, we opt for a fully-connected graph instead of a distance truncated graph to

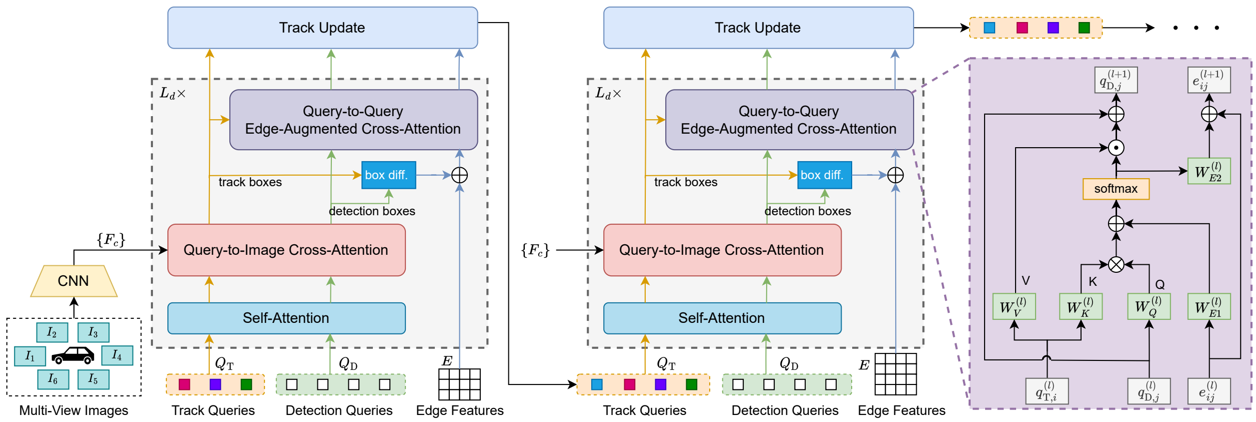


Figure 2. Overview of our ADA-Track framework. The transformer decoder takes decoupled track and detection queries, zero-initialized edge features, and multi-view image features as input. Each decoder layer first refines query features using a self-attention and a query-to-image cross-attention for object detection. Then a query-to-query edge-augmented cross-attention is applied to refine detection query and edge features for data association. By stacking this decoder layer, query features are updated for both tasks alternately and iteratively. A track update module associates both query sets and produces track queries for the next frame.

ensure the same graph structure in different layers, enabling the edge features to iterate over layers.

Our learned association leverages both appearance and geometric features. Formally, for each decoder layer l, we use track $Q_\mathrm{T}^{(l)} \in \mathbb{R}^{N_T \times d_k}$ and detection queries $Q_\mathrm{D}^{(l)} \in \mathbb{R}^{N_D \times d_k}$ as node features to provide appearance information obtained from the image-to-query cross-attention, where d_k is the number of channels. An MLP embeds aggregated pair-wise box differences to produce a relative positional encoding $E_\mathrm{pos}^{(l)} \in \mathbb{R}^{N_D \times N_T \times d_k}$ for edge features, *i.e.* $E_\mathrm{pos}^{(l)} = \mathrm{MLP}(B_\mathrm{diff}^{(l)})$, where $B_\mathrm{diff}^{(l)} = \{b_{\mathrm{diff},ij}^{(l)}\} \in \mathbb{R}^{N_T \times N_D \times 9}$. These box position features are defined for each pair $\{i, j\}$ of track $b_{\mathrm{T},i}^{(l)} \in \mathbb{R}^9$ and detection boxes $b_{\mathrm{D},j}^{(l)} \in \mathbb{R}^9$ by calculating their absolute difference $b_{\mathrm{diff},ij}^{(l)} = |b_{\mathrm{T},i}^{(l)} - b_{\mathrm{D},j}^{(l)}|$. The position encoding $E_\mathrm{pos}^{(l)}$ is added to the edge features $E^{(l)} \in \mathbb{R}^{N_D \times N_T \times d_k}$ as part of the input to the edge-augmented cross-attention, *i.e.* $E^{(l)} \leftarrow E_\mathrm{pos}^{(l)} + E^{(l)}$. As the initial edge features $E^{(0)}$ are zero-initialized for each frame, the input edge features are equal to the edge position encoding for the first layer $l = 1$, *i.e.* $E^{(1)} = E_\mathrm{pos}^{(1)}$.

As shown in the right part of Figure 2, we treat track queries $Q_\mathrm{T}^{(l)}$ as source set (key and value) and detection queries $Q_\mathrm{D}^{(l)}$ as target set (query) in the edge-augmented cross-attention. The edge-augmented attention

$$A^{(l)} = \mathrm{softmax}\Big(\frac{(Q_\mathrm{D}^{(l)} W_Q^{(l)})(Q_\mathrm{T}^{(l)} W_K^{(l)})^T}{\sqrt{d_k}} + E^{(l)} W_{E1}^{(l)}\Big) \tag{1}$$

takes both dot-product and edge features into consideration, where $\{W_Q^{(l)}, W_K^{(l)}\} \in \mathbb{R}^{d_k \times d_k}$ and $W_{E1}^{(l)} \in \mathbb{R}^{d_k \times 1}$ are learnable weights. The feature representation of the targets, *i.e.* the detection queries, as well as the edge features are updated using the attention $A^{(l)} \in \mathbb{R}^{N_D \times N_T \times 1}$, *i.e.*

$$Q_\mathrm{D}^{(l+1)} = Q_\mathrm{D}^{(l)} + \hat{A}^{(l)}(Q_\mathrm{T}^{(l)} W_V^{(l)}),\, E^{(l+1)} = E^{(l)} + A^{(l)} W_{E2}^{(l)}, \tag{2}$$

where $W_V^{(l)} \in \mathbb{R}^{d_k \times d_k}$ and $W_{E2}^{(l)} \in \mathbb{R}^{1 \times d_k}$ are learnable weights and $\hat{A}^{(l)} \in \mathbb{R}^{N_D \times N_T}$ is $A^{(l)}$ with squeezed third dimension. The update of $Q_\mathrm{D}^{(l)}$ enables a feature integration of tracking and association information, resulting in a better query-to-image interaction for the next layer $l + 1$.

As there are no existing tracks in the first frame, we skip the edge-augmented cross-attention for $t = 1$ and hence the decoder layer becomes identical to the object detector, *e.g.* DETR3D [55] or PETR [40].

3.2. Track Update

After all L_d layers of the decoder, we obtain the final track and detection queries $Q_T^{(L_d)}$ and $Q_E^{(L_d)}$ as well as edge features $E^{(L_d)}$. The track update module associates both query types and propagates feature embeddings of track queries Q_T^{t+1} and their corresponding reference points C_T^{t+1} to $t{+}1$.

Data association We use different association schemes for training and inference. During training, we first match tracks and detections to ground-truth objects (details in Section 3.3). If a track and a detection query are matched to the same ground-truth identity, both queries are considered as a matched pair. All track queries that are unmatched with a ground-truth are terminated, while all detection queries that are matched to the newly appearing ground-truth objects spawn a new track. During inference, we apply an MLP followed by a sigmoid function on the final edge features $E^{(L_d)}$ to estimate the affinity scores S between all track-detection pairs, *i.e.* $S = \mathrm{sigmoid}(\mathrm{MLP}(E^{(L_d)})) \in \mathbb{R}^{N_D \times N_T}$. Then, the score matrix S is used as matching

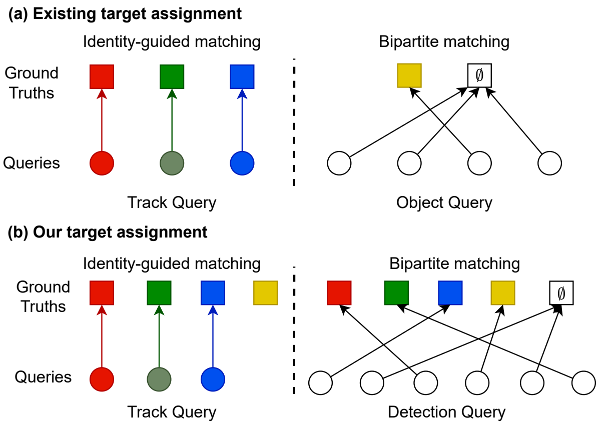


Figure 3. Target assignments: Tracking-by-attention applies identity-guided matching for track queries and then matches detection queries to remaining ground truths using the Hungarian Algorithm. Our method employs the same matching rules for both query types, but detection queries are matched to all ground-truths.

costs of a Hungarian Algorithm [28] to obtain a one-to-one matching. We heuristically keep the unmatched tracks for $T_d = 5$ frames and mark them as temporally inactive tracks before termination. An unmatched detection box initializes a new track, if its confidence is higher than $\tau_{\text{new}} = 0.4$.

Feature and box update Given a matched pair $\{i, j\}$, the track instance i is assigned with a track query $q_{\text{T},i}^{(L_d)}$ and a predicted track box $b_{\text{T},i}$. The same is for the detection instance j with detection query $q_{\text{D},j}^{(L_d)}$ and detection box $b_{\text{D},j}$. Hence, we need to determine the resulting query and box for this associated pair. Similar to [16], we empirically choose the detection query $q_{\text{D},j}^{(L_d)}$ and detection box $b_{\text{D},j}$ to represent the associated pair $\{i, j\}$, which corresponds to an update of its associated track i at frame t: $\hat{q}_{\text{T},i}^t = q_{\text{D},j}^{(L_d)}$ and $\hat{b}_{\text{T},i}^t = b_{\text{D},j}$. We will analyse this choice in Section 4.4. For unmatched detection or track queries, we directly use their respective features and boxes. Eventually, the track queries $\hat{Q}_{\text{T}}^t = \{\hat{q}_{\text{T},i}^t\}$ and their corresponding boxes $\hat{B}_{\text{T}}^t = \{\hat{b}_{\text{T},i}^t\}$ determine the final output of frame t.

Query Propagation We directly use the updated query features $\hat{Q}_{\text{T}}^t$ for the next frame, *i.e.* $Q_{\text{T}}^{t+1} = \hat{Q}_{\text{T}}^t$. We also propagate their 3D reference point after applying a simple motion update following MUTR3D [62]. Concretely, the 3D reference point (box center) $\hat{c}_{\text{T},i}^t$ and the BEV velocity $\hat{v}_{\text{T},i}^t$ are extracted from the box parameter of $\hat{b}_{\text{T},i}^t$. We then predict the reference point for the next frame $t+1$ with a constant velocity assumption: $c_{\text{T},i}^{t+1} = \hat{c}_{\text{T},i}^t + \hat{v}_{\text{T},i}^t \Delta t$, where Δt is the time difference between both frames. We transform the predicted reference points c_{T}^{t+1} to the new vehicle coordinate system with ego-motion compensation. Together, the track queries combined with the predicted reference points $\{Q_{\text{T}}^{t+1}, C_{\text{T}}^{t+1}\}$ serve as the input for frame $t+1$.

3.3. Training

Target assignment Tracking-by-attention approaches use identity-guided matching for the track queries, while the Hungarian Algorithm [28] matches the remaining ground-truth boxes and the object queries. In this work, track and detection queries detect objects independently and are explicitly associated with each other, which requires isolated matching for both query types. To achieve that, we match all ground-truth identities with detection queries using the Hungarian Algorithm, in addition to the same identity-guided matching for track queries. Figure 3 illustrates this difference in target assignment. These matching results assign the targets for the box losses. In addition, we apply an explicit loss function for the association module, where we regard it as a binary classification problem. If a ground-truth identity is matched with both, a track query and a detection query, the edge between these two queries should be classified as positive. In all other cases, *e.g.* one of both queries is matched with $\emptyset$, or both queries are matched to different ground truths, the association target for this detection-track pair should be negative.

Loss function For the bounding box loss in both query types, we follow [40, 55] and use the Focal Loss [37] as the classification loss $\mathcal{L}_{\text{cls}}$ and ℓ_1-loss as the regression loss $\mathcal{L}_{\text{reg}}$. The track and detection queries contribute to separate loss terms. This results in $\mathcal{L}_{\text{cls,T}}$ and $\mathcal{L}_{\text{reg,T}}$ for the track and $\mathcal{L}_{\text{cls,D}}$ and $\mathcal{L}_{\text{reg,D}}$ for the detection part. For association, we use Focal Loss with $\alpha = 0.5$ and $\gamma = 1.0$, denoted as $\mathcal{L}_{\text{asso}}$. We include auxiliary losses after each intermediate decoder layer for all the previously mentioned loss terms.

Given a training sequence with T frames, the losses for the detection queries are calculated for all T frames, whereas the losses for track queries and association are calculated from the second frame onwards. The overall loss for the whole training sequence can be formulated as

$$\begin{aligned}\mathcal{L} = &\sum_{t=1}^{T} (\lambda_{\text{cls}} \mathcal{L}_{\text{cls,D}}^t + \lambda_{\text{reg}} \mathcal{L}_{\text{reg,D}}^t) + \\ &\sum_{t=2}^{T} (\lambda_{\text{cls}} \mathcal{L}_{\text{cls,T}}^t + \lambda_{\text{reg}} \mathcal{L}_{\text{reg,T}}^t + \lambda_{\text{loss}} \mathcal{L}_{\text{asso}}^t).\end{aligned} \tag{3}$$

We use $\lambda_{\text{cls}} = 2.0, \lambda_{\text{reg}} = 0.25$ following existing works [40, 55] and $\lambda_{\text{asso}} = 10.0$. The impact of the loss weight λ_{asso} is further analyzed in the supplementary.

4. Experiments

4.1. Experiment Setup

Dataset We evaluate our approach on the nuScenes [6] dataset. NuScenes is a large-scale dataset for autonomous driving with 700, 150, and 150 sequences for training, validation, and testing. Each sequence is 20 seconds in length.

	Method	Paradigm	AMOTA↑	AMOTP↓	Recall↑	MOTA↑	IDS↓	FP ↓	FN ↓	TP↑
DETR3D	TBA-Baseline [62]	TBA	0.321	1.448	0.452	0.283	**474**	15269	43828	57595
	TBD-Baseline	TBD	0.350	1.427	0.467	0.308	944	**14316**	42980	57973
	MUTR3D [62]	TBA	0.294	1.498	0.427	0.267	3822	–	–	–
	DQTrack [34]	TBD	0.367	**1.351**	–	–	1120	–	–	–
	ADA-Track (ours)	ours	**0.378**	1.391	**0.507**	**0.343**	981	15443	**38466**	**62450**
	STAR-Track† [18]	TBA	0.379	**1.358**	0.501	0.360	**372**	–	–	–
	ADA-Track-long† (ours)	ours	**0.392**	1.375	**0.523**	**0.363**	897	14711	36945	64055
PETR	TBA-Baseline [62]	TBA	0.407	1.357	0.511	0.369	271	14401	39487	62139
	TBD-Baseline	TBD	0.452	1.275	0.540	0.405	1035	**13873**	35269	65593
	PF-Track [46] (w/o extension)	TBA	0.453	–	–	–	642	–	–	–
	PF-Track‡ [46]	TBA	**0.479**	**1.227**	0.590	**0.435**	**181**	–	–	–
	ADA-Track (ours)	ours	**0.479**	1.246	**0.602**	0.430	767	15385	**31402**	**69728**

Table 1. Results on the nuScenes validation set. We group methods by using the DETR3D or PETR detector. Tracking-by-attention and tracking-by-detection baselines are shown in the first two columns of each group. The remaining part compares our method (shown in light gray background) with existing works. † denotes total training epochs of 48. ‡ indicates annotations from future frames are required.

Method	AMOTA↑	AMOTP↓	Recall↑	MOTA↑	IDS↓
DEFT [9]	0.177	1.564	0.338	0.156	6901
QD-3DT [24]	0.217	1.550	0.375	0.198	6856
CC-3DT [20]	0.410	1.274	0.538	0.357	3334
Time3D [30]	0.214	1.360	–	0.173	–
MUTR3D [62]	0.270	1.494	0.411	0.245	6018
PF-Track [46]	0.434	1.252	0.538	0.378	**249**
STAR-Track [18]	0.439	1.256	**0.562**	**0.406**	607
ADA-Track (ours)	**0.456**	**1.237**	0.559	**0.406**	834

Table 2. Comparison with state-of-the-art end-to-end multi-camera 3D MOT approaches on the nuScenes test split. The bottom part compares our method with other query-based methods that use DETR3D or PETR. The top part shows other methods.

The sensor equipment contains LiDAR, RADAR, six cameras covering 360° Field of View as well as IMU and GPS.

Metrics The primary metrics for 3D MOT on nuScenes are AMOTA (average multi-object tracking accuracy) and AMOTP (average multi-object tracking precision) [56]. AMOTA is an average of the MOTAR over multiple recalls, where MOTAR is the recall-normalized MOTA at the corresponding recall r. The calculation of MOTA takes IDS (identity switch), FP (false positive) and FN (false negative) into consideration. AMOTP are the averaged position errors of all TPs (true positive) over all recalls. NuScenes also uses secondary metrics from CLEAR MOT [3], including MOTA, MOTP, IDS, FP, FN, *etc*. All the values of the CLEAR MOT metrics are reported at the recall R, where the highest MOTA is reached, *i.e.* $R = \text{argmax}_r \text{MOTA}_r$.

Implementation details To compare with most multi-camera MOT works [18, 34, 46, 62], we validate our approach using two query-based detectors: DETR3D [55] and PETR [40]. For DETR3D experiments, input images are full-resolution 1600×900 and we use ResNet-101 [21] as the backbone with an FPN [36]. For PETR experiments, we crop the images to 1600×640 and use VoVNetV2 [29] with FPN [36] as image feature extractor. We initialize the model using the corresponding single-frame detector checkpoints pre-trained for 24 epochs. We then train our tracker for 24 epochs on sampled mini-sequences which contain $T = 3$ frames. For all DETR3D experiments, we freeze the weights of the image backbone and FPN following STAR-Track [18]. For PETR, we only freeze the image backbone following PF-Track [46]. All models are trained using a cosine-annealing schedule with an initial learning rate of $2e^{-4}$ and an AdamW [41] optimizer with a weight decay of $1e^{-2}$. All DETR3D experiments are trained on four V100 GPUs, while all PETR experiments use eight A100 GPUs. Each GPU holds one batch element.

4.2. Paradigm Comparison

Baselines To validate the superiority of our proposed alternating detection and association paradigm, we first compare our method with query-based tracking-by-attention and tracking-by-detection baselines that are also without bells and whistles. We select MUTR3D [62] as the tracking-by-attention baseline by reproducing it on DETR3D and PETR. For the tracking-by-detection baseline, we modify our architecture to follow Figure 1b. To that end, we first use standard DETR3D or PETR decoders to process track and detection queries independently. Both sets of queries are then fed into the association module by stacking our proposed query-to-query edge-augmented cross-attentions (Section 3.1). This architecture shares the implementation details with ADA-Track as described in Section 4.1, *e.g.* batch size, training epochs, or weight freezing. We denote both baselines as **TBA-Baseline** and **TBD-Baseline**.

Results Table 1 shows the comparison on the nuScenes validation split, where the baselines are shown in the top part of each detector group. For both detectors, the TBD-Baselines achieve significantly higher AMOTA than the TBA-Baselines. On top, ADA-Track outperforms the TBD-Baselines by 2.8%P AMOTA for DETR3D and 2.7%P for PETR, respectively, while achieving a considerably higher recall. Considering the secondary metrics, we observe that the TBA-Baseline produces fewer IDS than the TBD-Baseline and ADA-Track, *i.e.* both methods with explicit association. However, TBA-Baseline achieves a low IDS by tracking only the easy cases, resulting in a substantially higher FN and lower MOTA. Compared to TBD-Baseline, ADA-Track again improves MOTA by a much higher TP and lower FN. These observations show the importance of the decoupled queries with explicit association as in the TBD-Baseline and in ADA-Track, which produces distinguishable query representations for both tasks and thus improves their performances. ADA-Track further improves over the TBD-Baseline by fully utilizing the task interdependency using alternating detection and association.

4.3. Comparison with Existing Works

We compare ADA-Track with existing works based on TBA or TBD in the remaining part of each detector group in Table 1. Due to an implementation issue, MUTR3D [62] reported a lower performance than our TBA-Baseline, which is a reproduction of MUTR3D with fewer training epochs and fixed backbone during training. DQ-Track [34] also uses decoupled queries and a sophisticated learned association module following TBD. ADA-Track outperforms it by 1.2%P AMOTA, which again highlights the effectiveness of our alternating detection and association design. The training of STAR-Track [18] requires an initialization on a pre-trained MUTR3D checkpoint, which results in a total training epoch of 48. For a fair comparison, we additionally train our model for 48 epochs, denoted as ADA-Track-long, which outperforms STAR-Track by 1.3%P AMOTA.

Comparing based on the PETR detector, we achieve on-par performance (0.479 AMOTA) with PF-Track [46]. However, PF-Track [46] is a joint tracking and prediction method, utilizing a track extension module to replace low-confidence detections with predicted trajectories when outputting tracking results, which additionally requires supervision from future frames. Our ADA-Track is a pure tracking method but still achieves the same AMOTA. Compared to PF-Track without track extension, ADA-Track achieves considerably higher AMOTA by 2.6%P.

We compare ADA-Track with end-to-end methods on the test split in Table 2, where we train ADA-Track based on DETR3D [55] with VoVNetV2-99 [29] backbone on both training and validation set. Compared to query-based methods using DETR3D or PETR in the second part of Table 2, ADA-Track achieves 0.456 AMOTA and 1.237 AMOTP, outperforming the recent state-of-the-art methods STAR-Track [18] and PF-Track [46] by 1.7%P and 2.2%P AMOTA, respectively. Compared to other non-query-based methods, ADA-Track also achieves the best performance, improving CC-3DT [20] that used a stronger BEVFormer [35] detector by 4.6% AMOTA.

In addition, we highlight that ADA-Track can be combined with many components proposed in existing tracking-by-attention works. For instance, the Past and Future Reasoning in [46] or the Latent Motion Model in [18] improve the query embedding or the motion update during query propagation, while we use the simple approach as in MUTR3D. These extensions can be seamlessly integrated into our framework and further improvements are expected.

4.4. Ablation Study

Number of training frames Table 3 shows the impact of varying the training sample length T. We observe that an increasing sample length increases the AMOTA and many secondary metrics. A particularly substantial improvement occurs when increasing from a sample length of 2 to 3 frames, resulting in a notable 3.2%P increase in AMOTA. The reason is that the autoregressive training scheme becomes active when $T \geq 3$, allowing association results to propagate into subsequent frames. This enables optimization through gradients across multiple frames to optimize the whole sequence, significantly boosting robustness during inference. A further increase in training frames from 3 to 4 yields marginal improvements but a considerable computational and memory overhead during training. Thus, we use $T = 3$ as the default setting.

Joint optimization One of the main contributions of this work is the introduction of an alternating detection and association paradigm, where both tasks iteratively inform each other layer-by-layer for joint optimization. To assess its effectiveness, we combine the predictions of the bounding boxes and the association scores from different decoder layers. The first part of Table 4 illustrates using bounding boxes from the last layer alongside association scores from decoder layers 1 to 5. A notable increase in AMOTA is observed when using association scores from the second instead of the first layer. Using association scores from higher layers leads to gradually increased AMOTA. We also observe a considerable reduction in IDS. Together, this shows the iterative improvements of the association. The second part of Table 4 presents results using boxes from various layers combined with association scores from the last layer. We can observe a similar tendency in AMOTA increase with higher layers as in the first part, where the most significant increase occurs from the first to the second layer. In addition, using box predictions in later layers results in a notable improvement in AMOTP, confirming the iterative optimiza-

T	AMOTA↑	AMOTP↓	Recall↑	MOTA↑	IDS↓
2	0.346	1.397	0.477	0.295	1099
3	0.378	**1.391**	0.507	0.343	981
4	**0.379**	1.405	**0.516**	**0.345**	**816**

Table 3. Ablation study on the training sample length T.

box	asso	AMOTA↑	AMOTP↓	Recall↑	MOTA↑	IDS↓
6	1	0.342	1.420	0.485	0.298	2063
6	2	0.369	1.399	0.485	0.330	1392
6	3	0.372	1.399	0.498	0.341	1088
6	4	0.373	1.398	0.508	0.338	1090
6	5	0.374	1.397	0.501	**0.343**	968
1	6	0.279	1.465	0.398	0.248	1431
2	6	0.364	1.413	0.480	0.329	1197
3	6	0.368	1.406	0.489	0.337	1023
4	6	0.373	1.399	**0.513**	0.342	1023
5	6	0.373	**1.390**	0.501	0.339	**924**
6	6	**0.378**	1.391	0.507	**0.343**	981

Table 4. Ablation study on combining bounding box and association outputs from different decoder layers.

w_t	AMOTA↑	AMOTP↓	Recall↑	MOTA↑	IDS↓
0	**0.378**	**1.391**	0.507	**0.343**	981
0.3	0.367	1.405	0.474	0.330	**865**
0.5	0.371	1.393	**0.508**	0.322	904
0.7	0.363	1.418	0.501	0.325	967
1.0	0.365	1.398	0.500	0.325	928

Table 5. Ablation study on the feature update weight w_t.

tion of localization precision. Overall, combining outputs from distinct layers yields gradual performance increases when outputs from higher layers are used. This observation shows that the relevance between box and association predictions is not only constrained within the same layer but across different layers, which validates the iterative optimization through stacked layers of both tasks.

Feature update weight We analyze the impact of the feature update as discussed in Section 3.2. Here we generalize the feature update to a weighted average, *i.e.* $q_{\mathrm{T},i} = w_{\mathrm{T}} q_{\mathrm{T},i}^{(L_d)} + (1 - w_{\mathrm{T}}) q_{\mathrm{D},j}^{(L_d)}$, where w_t is the update rate. As shown in Table 5, substituting track query features with detection query features ($w_t = 0$) yields the optimal performance, surpassing other values by 0.7% to 1.5% in AMOTA. This can be attributed to the fact that detection queries inherently incorporate track query features within their representation through edge-augmented cross-attention. Thus, an additional merging between detection and track query features is unnecessary.

Box update Both track and detection queries predict bounding boxes, resulting in two candidates for each instance when two queries are associated. We use the box from the detection side to represent this associated pair and validate this choice in Table 6. As shown in the first row (Exp. A), selecting the track box results in an AMOTA decrease of 1.3%P compared to our approach (Exp. C), which shows the better quality of detection boxes. One could argue that detection queries are further refined in the edge-augmented cross-attention but track queries are not. In the second row of Table 6, we additionally reverse the edge-augmented cross-attention by using tracks as queries and detections as keys. This leads to an additional AMOTA decrease of 2.2%P. This finding also verifies the necessity of the decoupled queries for decoupled tasks, where using detection queries to locate objects and associate them to track queries is more effective than relying on track queries to locate objects.

Exp.	use track box	track as query	AMOTA↑	AMOTP↓	Recall↑
A	✓		0.365	**1.390**	0.489
B	✓	✓	0.343	1.423	0.489
C			**0.378**	1.391	**0.507**

Table 6. Ablation study on box update. *use track box* denotes that boxes predicted from track queries are used as output. *track as query* denotes that track queries are regarded as queries and detection queries as keys in the edge-augmented cross-attention.

5. Conclusion

We presented a novel query-based multi-camera 3D multi-object tracking approach, termed ADA-Track. We observed that decoupling the detection and association task while simultaneously leveraging the synergies between these tasks is the key to accomplish high-quality tracking. In line with this finding, we proposed a paradigm that conducts both, detection and association, in an alternating manner. In addition, we proposed a learned association module based on edge-augmented cross-attention which can be seamlessly integrated into any query-based decoder. Extensive experiments show the effectiveness of our approach compared to other paradigms, while achieving state-of-the-art performance on the nuScenes tracking benchmark.

Acknowledgement This work is a result of the joint research project STADT:up (Förderkennzeichen 19A22006O). The project is supported by the German Federal Ministry for Economic Affairs and Climate Action (BMWK), based on a decision of the German Bundestag. The author is solely responsible for the content of this publication. Juergen Gall has been supported by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) GA 1927/5-2 (FOR 2535 Anticipating Human Behavior) and the ERC Consolidator Grant FORHUE (101044724).

References

[1] Nir Aharon, Roy Orfaig, and Ben-Zion Bobrovsky. Bot-sort: Robust associations multi-pedestrian tracking. *ArXiv*, abs/2206.14651, 2022. 3

[2] Nuri Benbarka, Jona Schröder, and Andreas Zell. Score refinement for confidence-based 3d multi-object tracking. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8083–8090. IEEE, 2021. 3

[3] 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

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

[5] Guillem Brasó and Laura Leal-Taixé. Learning a neural solver for multiple object tracking. In *IEEE/CVF conference on computer vision and pattern recognition*, pages 6247–6257, 2020. 3

[6] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 2, 5

[7] Jinkun Cao, Jiangmiao Pang, Xinshuo Weng, Rawal Khirodkar, and Kris Kitani. Observation-centric sort: Rethinking sort for robust multi-object tracking. In *IEEE/CVF conference on computer vision and pattern recognition*, pages 9686–9696, 2023. 3

[8] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *European conference on computer vision*, pages 213–229. Springer, 2020. 1, 2, 3

[9] Mohamed Chaabane, Peter Zhang, J Ross Beveridge, and Stephen O'Hara. Deft: Detection embeddings for tracking. *ArXiv*, abs/2102.02267, 2021. 6

[10] Cheng-Che Cheng, Min-Xuan Qiu, Chen-Kuo Chiang, and Shang-Hong Lai. Rest: A reconfigurable spatial-temporal graph model for multi-camera multi-object tracking. In *IEEE/CVF International Conference on Computer Vision*, pages 10051–10060, 2023. 3

[11] Hsu-kuang Chiu, Antonio Prioletti, Jie Li, and Jeannette Bohg. Probabilistic 3d multi-object tracking for autonomous driving. *ArXiv*, abs/2001.05673, 2020. 3

[12] Hsu-kuang Chiu, Jie Li, Rareş Ambruş, and Jeannette Bohg. Probabilistic 3d multi-modal, multi-object tracking for autonomous driving. In *IEEE international conference on robotics and automation (ICRA)*, pages 14227–14233. IEEE, 2021. 3

[13] Peng Chu, Jiang Wang, Quanzeng You, Haibin Ling, and Zicheng Liu. Transmot: Spatial-temporal graph transformer for multiple object tracking. In *IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 4870–4880, 2023. 3

[14] Patrick Dendorfer, Hamid Rezatofighi, Anton Milan, Javen Qinfeng Shi, Daniel Cremers, Ian D. Reid, Stefan Roth, Konrad Schindler, and Laura Leal-Taixé. Mot20: A benchmark for multi object tracking in crowded scenes. *ArXiv*, abs/2003.09003, 2020. 3

[15] Shuxiao Ding, Eike Rehder, Lukas Schneider, Marius Cordts, and Juergen Gall. End-to-end single shot detector using graph-based learnable duplicate removal. In *DAGM German Conference on Pattern Recognition*, pages 375–389. Springer, 2022. 2, 3

[16] Shuxiao Ding, Eike Rehder, Lukas Schneider, Marius Cordts, and Juergen Gall. 3dmotformer: Graph transformer for online 3d multi-object tracking. In *IEEE/CVF International Conference on Computer Vision*, pages 9784–9794, 2023. 2, 3, 5

[17] Simon Doll, Richard Schulz, Lukas Schneider, Viviane Benzin, Markus Enzweiler, and Hendrik PA Lensch. Spatialdetr: Robust scalable transformer-based 3d object detection from multi-view camera images with global cross-sensor attention. In *European Conference on Computer Vision*, pages 230–245. Springer, 2022. 2

[18] Simon Doll, Niklas Hanselmann, Lukas Schneider, Richard Schulz, Markus Enzweiler, and Hendrik PA Lensch. Startrack: Latent motion models for end-to-end 3d object tracking with adaptive spatio-temporal appearance representations. *IEEE Robotics and Automation Letters*, 2023. 2, 6, 7

[19] Yunhao Du, Zhicheng Zhao, Yang Song, Yanyun Zhao, Fei Su, Tao Gong, and Hongying Meng. Strongsort: Make deepsort great again. *IEEE Transactions on Multimedia*, 2023. 3

[20] Tobias Fischer, Yung-Hsu Yang, Suryansh Kumar, Min Sun, and Fisher Yu. Cc-3dt: Panoramic 3d object tracking via cross-camera fusion. In *Conference on Robot Learning*, pages 2294–2305. PMLR, 2023. 6, 7

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 6

[22] Jan Hosang, Rodrigo Benenson, and Bernt Schiele. Learning non-maximum suppression. In *IEEE conference on computer vision and pattern recognition*, pages 4507–4515, 2017. 2

[23] Anthony Hu, Zak Murez, Nikhil Mohan, Sofía Dudas, Jeffrey Hawke, Vijay Badrinarayanan, Roberto Cipolla, and Alex Kendall. Fiery: Future instance prediction in bird's-eye view from surround monocular cameras. In *IEEE/CVF International Conference on Computer Vision*, pages 15273–15282, 2021. 1, 2

[24] 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. 6

[25] Junjie Huang and Guan Huang. Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. *ArXiv*, abs/2203.17054, 2022. 2

[26] Junjie Huang, Guan Huang, Zheng Zhu, Yun Ye, and Dalong Du. Bevdet: High-performance multi-camera 3d object de-

tection in bird-eye-view. *ArXiv*, abs/2112.11790, 2021. 1, 2

[27] Md Shamim Hussain, Mohammed J Zaki, and Dharmashankar Subramanian. Global self-attention as a replacement for graph convolution. In *28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 655–665, 2022. 2, 3

[28] Harold W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics (NRL)*, 52, 1955. 3, 5

[29] Youngwan Lee, Joong-won Hwang, Sangrok Lee, Yuseok Bae, and Jongyoul Park. An energy and gpu-computation efficient backbone network for real-time object detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, 2019. 6, 7

[30] Peixuan Li and Jieyu Jin. Time3d: End-to-end joint monocular 3d object detection and tracking for autonomous driving. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3885–3894, 2022. 2, 3, 6

[31] Peizheng Li, Shuxiao Ding, Xieyuanli Chen, Niklas Hanselmann, Marius Cordts, and Juergen Gall. Powerbev: A powerful yet lightweight framework for instance prediction in bird's-eye view. In *Thirty-Second International Joint Conference on Artificial Intelligence*, pages 1080–1088, 2023. 1, 2

[32] Xiaoyu Li, Tao Xie, Dedong Liu, Jinghan Gao, Kun Dai, Zhiqiang Jiang, Lijun Zhao, and Ke Wang. Poly-mot: A polyhedral framework for 3d multi-object tracking. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 9391–9398. IEEE, 2023. 3

[33] Yinhao Li, Han Bao, Zheng Ge, Jinrong Yang, Jianjian Sun, and Zeming Li. Bevstereo: Enhancing depth estimation in multi-view 3d object detection with temporal stereo. In *AAAI Conference on Artificial Intelligence*, pages 1486–1494, 2023. 2

[34] Yanwei Li, Zhiding Yu, Jonah Philion, Anima Anandkumar, Sanja Fidler, Jiaya Jia, and Jose Alvarez. End-to-end 3d tracking with decoupled queries. In *IEEE/CVF International Conference on Computer Vision*, pages 18302–18311, 2023. 2, 3, 6, 7

[35] Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Yu Qiao, and Jifeng Dai. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European conference on computer vision*, pages 1–18. Springer, 2022. 1, 2, 7

[36] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017. 6

[37] Tsung-Yi Lin, Priya Goyal, Ross B. Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42:318–327, 2017. 5

[38] Xuewu Lin, Tianwei Lin, Zixiang Pei, Lichao Huang, and Zhizhong Su. Sparse4d: Multi-view 3d object detection with sparse spatial-temporal fusion. *ArXiv*, abs/2211.10581, 2022. 2

[39] Xuewu Lin, Tianwei Lin, Zixiang Pei, Lichao Huang, and Zhizhong Su. Sparse4d v2: Recurrent temporal fusion with sparse model. *ArXiv*, abs/2305.14018, 2023. 2

[40] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. In *European Conference on Computer Vision*, pages 531–548. Springer, 2022. 2, 3, 4, 5, 6

[41] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018. 6

[42] Tim Meinhardt, Alexander Kirillov, Laura Leal-Taixe, and Christoph Feichtenhofer. Trackformer: Multi-object tracking with transformers. In *IEEE/CVF conference on computer vision and pattern recognition*, pages 8844–8854, 2022. 1, 2

[43] Anton Milan, Laura Leal-Taixé, Ian D. Reid, Stefan Roth, and Konrad Schindler. Mot16: A benchmark for multi-object tracking. *ArXiv*, abs/1603.00831, 2016. 3

[44] Duy MH Nguyen, Roberto Henschel, Bodo Rosenhahn, Daniel Sonntag, and Paul Swoboda. Lmgp: Lifted multicut meets geometry projections for multi-camera multi-object tracking. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8866–8875, 2022. 3

[45] Ziqi Pang, Zhichao Li, and Naiyan Wang. Simpletrack: Understanding and rethinking 3d multi-object tracking. In *European Conference on Computer Vision*, pages 680–696. Springer, 2022. 3

[46] Ziqi Pang, Jie Li, Pavel Tokmakov, Dian Chen, Sergey Zagoruyko, and Yu-Xiong Wang. Standing between past and future: Spatio-temporal modeling for multi-camera 3d multi-object tracking. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17928–17938, 2023. 3, 6, 7

[47] Jinhyung Park, Chenfeng Xu, Shijia Yang, Kurt Keutzer, Kris M Kitani, Masayoshi Tomizuka, and Wei Zhan. Time will tell: New outlooks and a baseline for temporal multi-view 3d object detection. In *International Conference on Learning Representations*, 2022. 2

[48] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *European Conference on Computer Vision*, pages 194–210. Springer, 2020. 2

[49] Kha Gia Quach, Pha Nguyen, Huu Le, Thanh-Dat Truong, Chi Nhan Duong, Minh-Triet Tran, and Khoa Luu. Dyglip: A dynamic graph model with link prediction for accurate multi-camera multiple object tracking. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13784–13793, 2021. 3

[50] 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*, abs/2101.04206, 2021. 3

[51] Peize Sun, Jinkun Cao, Yi Jiang, Rufeng Zhang, Enze Xie, Zehuan Yuan, Changhu Wang, and Ping Luo. Transtrack: Multiple object tracking with transformer. *ArXiv*, abs/2012.15460, 2020. 1, 2, 3

[52] Li Wang, Xinyu Zhang, Wenyuan Qin, Xiaoyu Li, Jinghan Gao, Lei Yang, Zhiwei Li, Jun Li, Lei Zhu, Hong Wang, et al.

Camo-mot: Combined appearance-motion optimization for 3d multi-object tracking with camera-lidar fusion. *IEEE Transactions on Intelligent Transportation Systems*, 2023. 3

[53] Qitai Wang, Yuntao Chen, Ziqi Pang, Naiyan Wang, and Zhaoxiang Zhang. Immortal tracker: Tracklet never dies. *ArXiv*, abs/2111.13672, 2021. 3

[54] Shihao Wang, Yingfei Liu, Tiancai Wang, Ying Li, and Xiangyu Zhang. Exploring object-centric temporal modeling for efficient multi-view 3d object detection. In *IEEE/CVF International Conference on Computer Vision*, pages 3621–3631, 2023. 2

[55] Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *Conference on Robot Learning*, pages 180–191. PMLR, 2022. 2, 3, 4, 5, 6, 7

[56] Xinshuo Weng, Jianren Wang, David Held, and Kris Kitani. 3d multi-object tracking: A baseline and new evaluation metrics. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10359–10366. IEEE, 2020. 3, 6

[57] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. *IEEE International Conference on Image Processing (ICIP)*, pages 3645–3649, 2017. 3

[58] 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*, 45(6):7820–7835, 2022. 1, 2, 3

[59] Chenyu Yang, Yuntao Chen, Hao Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Yu Qiao, Lewei Lu, et al. Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 17830–17839, 2023. 2

[60] Jan-Nico Zaech, Alexander Liniger, Dengxin Dai, Martin Danelljan, and Luc Van Gool. Learnable online graph representations for 3d multi-object tracking. *IEEE Robotics and Automation Letters*, 7(2):5103–5110, 2022. 3

[61] Fangao Zeng, Bin Dong, Yuang Zhang, Tiancai Wang, Xiangyu Zhang, and Yichen Wei. Motr: End-to-end multiple-object tracking with transformer. In *European Conference on Computer Vision*, pages 659–675. Springer, 2022. 1, 2

[62] Tianyuan Zhang, Xuanyao Chen, Yue Wang, Yilun Wang, and Hang Zhao. Mutr3d: A multi-camera tracking framework via 3d-to-2d queries. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 4537–4546, 2022. 2, 5, 6, 7

[63] Yifu Zhang, Pei Sun, Yi Jiang, Dongdong Yu, Zehuan Yuan, Ping Luo, Wenyu Liu, and Xinggang Wang. Bytetrack: Multi-object tracking by associating every detection box. In *European Conference on Computer Vision*, 2021. 3

[64] Yunpeng Zhang, Zheng Zhu, Wenzhao Zheng, Junjie Huang, Guan Huang, Jie Zhou, and Jiwen Lu. Beverse: Unified perception and prediction in birds-eye-view for vision-centric autonomous driving. *ArXiv*, 2205.09743, 2022. 1, 2

[65] Yifu Zhang, Xinggang Wang, Xiaoqing Ye, Wei Zhang, Jincheng Lu, Xiao Tan, Errui Ding, Peize Sun, and Jingdong Wang. Bytetrackv2: 2d and 3d multi-object tracking by associating every detection box. *ArXiv*, abs/2303.15334, 2023. 3

[66] Zelin Zhao, Ze Wu, Yueqing Zhuang, Boxun Li, and Jiaya Jia. Tracking objects as pixel-wise distributions. In *European Conference on Computer Vision*, pages 76–94. Springer, 2022. 1, 2, 3

[67] Xingyi Zhou, Tianwei Yin, Vladlen Koltun, and Philipp Krähenbühl. Global tracking transformers. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8771–8780, 2022. 3