VIDEO-BASED 3D OBJECT DETECTION WITH LEARNABLE OBJECT-CENTRIC GLOBAL OPTIMIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

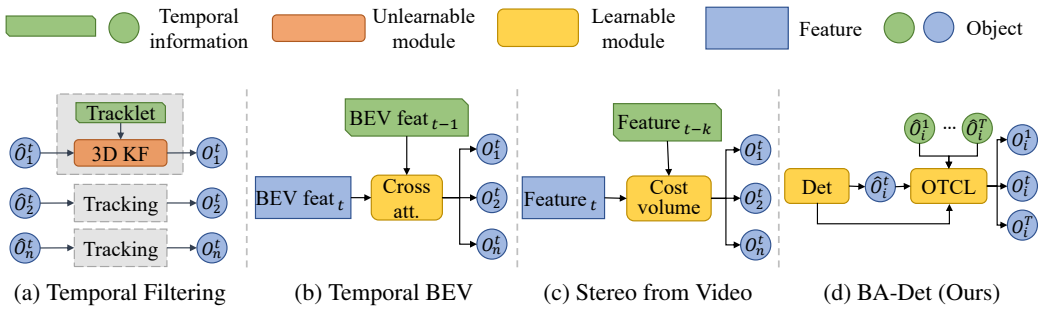
We study utilizing long-term temporal visual correspondence-based optimization for video-based 3D object detection in this work. Visual correspondence refers to one-to-one mappings for pixels across multiple images. Correspondence-based optimization is the cornerstone for 3D scene reconstruction but is less studied in 3D object detection, for that moving objects violate multi-view geometry constraints and are treated as outliers during scene reconstruction. We resolve this issue by treating objects as first-class citizens during correspondence-based optimization. In this work, we propose BA-Det, an end-to-end optimizable object detector with object-centric temporal correspondence learning and object-centric featuremetric bundle adjustment. Empirically, we verify the effectiveness and efficiency of BA-Det for multiple baseline 3D detectors under various setups. Our BA-Det achieves SOTA performance on the large-scale Waymo Open Dataset (WOD) with only marginal computation cost. Codes will be released soon.

1 INTRODUCTION

3D object detection is an important perception task, especially for indoor robots and autonomous-driving vehicles. Recently, image-only 3D object detection (Zhang et al., 2021; Li et al., 2022b) has been proven practical and made great progress. Despite simply relying on the prediction power of deep learning, finding correspondences play an important role in these methods for estimating per-pixel depth and the object pose in the camera frame. Popular correspondences include Perspective-n-Point (PnP) between pre-defined 3D keypoints (Zhang et al., 2021; Li et al., 2022a) and their 2D projections in monocular 3D object detection and Epipolar Geometry (Chen et al., 2020; Guo et al., 2021) in multi-view 3D object detection.

In real-world applications, cameras can capture video streams instead of unrelated frames, which suggests abundant temporal information is readily available for 3D object detection. However, unlike the single frame case, temporal visual correspondence has not been explored much in video 3D object detection. As summarized in Fig. 1, existing methods can be divided into three categories while each has its own limitations. Fig. 1a shows methods using 3D Kalman Filter (Brazil et al., 2020) to smooth the trajectory of each detected object. This approach is detector-agnostic and thus widely adopted, but it is just an output-level smoothing process without any feature learning, so the potential of video is under-exploited. Fig. 1b illustrates the temporal BEV (Bird's-eye view) approach (Li et al., 2022b; Huang & Huang, 2022; Liu et al., 2022) for video 3D object detection. They introduce the multi-frame temporal cross attention for BEV embeddings and train the transformer end-to-end. As for utilizing temporal information, temporal BEV methods rely solely on feature fusion while ignoring any explicit temporal correspondence. Fig. 1c shows stereo from video methods (Wang et al., 2022a;b). These methods explicitly construct pseudo-stereo view by ego motion, and then utilize the correspondence on the epipolar line of two frames for depth estimation, which is naturally cannot utilize more frames. Besides, although improvements can be observed on static objects, an inevitable limitation of these methods is that moving objects break the epipolar constraints. So these methods generally fuse inaccurate stereo depth estimation with monocular depth estimation and leave the end-to-end detection network to adaptively choose from two sources of depth information.

We seek a new method that can handle moving objects and can utilize long-term temporal correspondences. To handle moving objects, we can benefit from the object-centric global optimization



(a) Temporal Filtering (b) Temporal BEV (c) Stereo from Video (d) BA-Det (Ours)

Figure 1: **Illustration of how to leverage temporal information in different video-based 3D object detection paradigms.**

with reprojection constraints in Simultaneous Localization and Mapping (SLAM) (Yang & Scherer, 2019; Li et al., 2018). Instead of directly estimating depth from temporal cues, we utilize them to construct useful constraints to refine the pose prediction from network prediction. Specifically, we construct a non-linear least-square optimization problem in object-centric manner to optimize the pose of objects no matter whether they are moving or not.

For long-term temporal correspondence learning, hand-crafted descriptors like SIFT (Lowe, 2004) or ORB (Rublee et al., 2011) are no longer suitable for our end-to-end object detector. Besides, the long-term temporal correspondence needs to be robust to the viewpoint changes and severe occlusions. These traditional descriptors are too sparse that can not keep available for a long time. So, we expect to learn a dense temporal correspondence for all available frames. In this paper, as shown in Fig. 1d, we propose a video-based 3D object detection paradigm with learnable long-term temporal visual correspondence, called *BA-Det*. Specifically, the detector has two stages. In the first stage, a CenterNet-style monocular 3D object detector is applied for single-frame object detection. After associating the same objects in the video, the second stage detector extracts RoI features in each frame and matches local features on the object among multi-frames. With object-centric featuremetric bundle adjustment loss, our video object detector and temporal feature correspondence are learned jointly. During inference, we utilize the 3D object estimation from the first stage and object feature correspondence to optimize the pose and 3D box size of the object in each frame. Experiment results on the Waymo Open Dataset (WOD) show that our BA-Det could achieve state-of-the-art performance compared with other single-frame and multi-frame object detectors. We also conduct various ablation studies to demonstrate the effectiveness and efficiency of each component in our method.

2 Related Work

2.1 Video Object Detection

In the 2D Video Object Detection task, researchers mainly focus on the challenges caused by the moving objects, such as motion blur, partially captured and different viewpoints. Message passing network () is a common module to aggregate temporal information. FGFA (Zhu et al., 2017) is an end-to-end algorithm utilizing optical flow between frames to warp the corresponding features to the current frame and aggregate them by the attentional mechanism. SELSA (Wu et al., 2019) aggregates the object-level RoI features from other frames. MaskTrack R-CNN (Yang et al., 2019) first focuses on the instance mask instead of the bounding box to represent an object in the video.

As for 3D object detection, LiDAR-based methods (Caesar et al., 2020; Yin et al., 2021; Fan et al., 2022) usually align point clouds from consecutive frames by compensating ego-motion and simply accumulate them to alleviate the sparsity of point clouds. Object-level methods (Qi et al., 2021; You et al., 2022; Chen et al., 2022), handling the multi-frame point clouds of the tracked object, become a new trend. 3D object detection from the monocular video has not received enough attention from researchers. Kinematic3D (Brazil et al., 2020) is a pioneer work decomposing kinematic information into ego-motion and target object motion. However, they only apply 3D Kalman Filter (Kalman,

1960) based motion model for kinematic modeling and only consider the short-term temporal association (4 frames). Recently, BEVFormer (Li et al., 2022b) proposes an attentional transformer method to model the spatial and temporal relationship in the bird's-eye-view (BEV). A concurrent work, DfM (Wang et al., 2022a), inspired by Multi-view Geometry, considers two frames as stereo and applies the cost volume in stereo to estimate depth. However, a critical problem is that moving objects can not be handled in this paradigm.

2.2 Geometry in Videos

Utilizing 3D geometry in videos mainly aims to reconstruct the scene and estimate the camera pose. It is a classic topic of computer vision. Structure from Motion (SfM) (Schönberger & Frahm, 2016) and Multi-view Stereo (MVS) (Schönberger et al., 2016) are two paradigms to estimate the sparse depth from local features and recover the dense scene from every pixel, respectively. In robotics, researchers apply the 3D geometry theory for Simultaneous Localization and Mapping (SLAM) (Mur-Artal et al., 2015). To global optimize the 3D position of the feature points and the camera pose at each time, bundle adjustment algorithm (Triggs et al., 1999) is widely applied. However, most of them can only handle static scenes or manually filter the dynamic region.

In the deep learning era, with the development of object detection, object-level semantic SLAM (Nicholson et al., 2018; Li et al., 2018; Yang & Scherer, 2019) is rising, reconstructing the objects instead of the whole scene, which can handle the dynamic scenes and help the object localization in the video. In feature learning, feature correspondence learning (Sarlin et al., 2020; Sun et al., 2021) has received extensive attention in recent years. Deep learning has greatly changed the pipeline of feature matching. Besides, differentiable bundle adjustment, like BANet (Tang & Tan, 2019), makes the whole 3D geometry system end-to-end learnable.

3 Preliminary: Bundle Adjustment

Bundle Adjustment (Triggs et al., 1999) is a widely utilized global optimization technology in 3D reconstruction, which means optimally adjusting bundles of light rays from a given 3D feature to the camera center within different frames. Specifically, we use $\mathbf{P}_i = [x_i, y_i, z_i]^T$ to denote the i-th 3D point coordinates in the global reference frame. According to the perspective camera model, the image coordinates of the projected 3D point at frame t is

$$\Pi(\mathbf{T}_{gc}^t, \mathbf{P}_i, \mathbf{K}) = \frac{1}{z_i^t}\mathbf{K}(\mathbf{R}_{gc}^t\mathbf{P}_i + \mathbf{t}_{gc}^t), \tag{1}$$

where Π is the perspective projection transformation, $\mathbf{T}_{gc}^t = [\mathbf{R}_{gc}^t, \mathbf{t}_{gc}^t]$ is the camera pose in the global frame at time t, $\mathbf{R}_{gc}^t$ and $\mathbf{t}_{gc}^t$ are the rotation and the translation components of $\mathbf{T}_{gc}^t$, respectively, $\mathbf{K}$ is the camera intrinsic matrix, and z_i^t is the depth of the i-th 3D point in the camera frame at time t.

Bundle adjustment is a nonlinear least square problem to minimize the reprojection error as:

$$\{\bar{\mathbf{T}}_{gc}^t\}_{t=1}^T, \{\bar{\mathbf{P}}_i\}_{i=1}^m = \underset{\{\mathbf{T}_{gc}^t\}_{t=1}^T, \{\mathbf{P}_i\}_{i=1}^m}{\arg\min} \frac{1}{2}\sum_{i=1}^{m}\sum_{t=1}^{T} ||\mathbf{p}_i^t - \Pi(\mathbf{T}_{gc}^t, \mathbf{P}_i, \mathbf{K})||^2, \tag{2}$$

where $\mathbf{p}_i^t$ is the observed image coordinates of 3D point $\mathbf{P}_i$ on frame t. Bundle adjustment can be solved by Gauss-Newton or Levenberg–Marquardt algorithm effectively (Agarwal et al., 2022; Kümmerle et al., 2011).

4 BA-Det: Learnable Object-centric Global Optimized Detector

In this section, we introduce the framework of our BA-Det (Fig. 2), a learnable object-centric global optimization network. The pipeline consists of three parts: (1) First-stage single frame 3D object detector; (2) Second-stage object-centric temporal correspondence learning (OTCL) module; (3) Featuremetric object-centric bundle adjustment loss for temporal feature correspondence learning.

Given a video clip with consecutive frames $\mathcal{V} = \{I_1, I_2, \cdots, I_T\}$, video 3D object detection is to predict the class and other 3D attributes of each object in each frame. Let $\mathcal{O}_k^t$ be the k-th object in

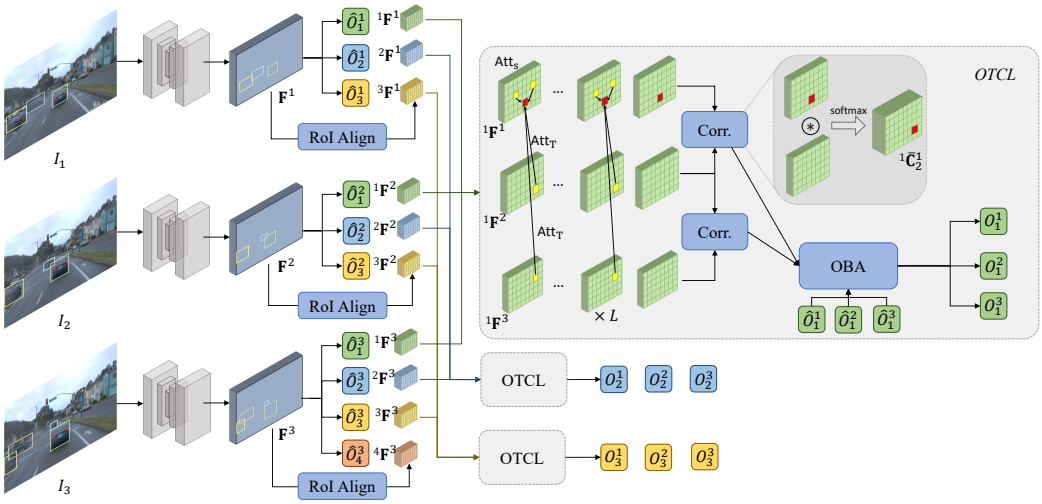


Figure 2: **A overview of the proposed BA-Det framework.** The left part of the framework is the first stage object detector to optimize the 3D object and its 2D bounding box. The second stage is called *OTCL*. In OTCL, we extract the RoI features ${}^k\mathbf{F}^t$ by RoIAlign, aggregate the RoI features and learn object-centric temporal correspondence using featuremetric object-centric bundle adjustment loss.

frame t. The 3D bounding box $\mathbf{B}_k^t$ has the attributes including object center $\mathbf{c}_t^k = [x_c, y_c, z_c]^T$ in the camera frame, size of the bounding box $\mathbf{s}_t^k = [w, h, l]^T$, and orientation $\mathbf{r}_t^k = [r_x, r_y, r_z]^T$. In most 3D object detection datasets, with the flat ground assumption, only yaw rotation r_y is considered.

BA-Det consists of two stages. We use a variant of CenterNet (Zhou et al., 2019) as our first stage object detector for individual frames. We follow MonoFlex (Zhang et al., 2021) in foreground label assignment by using the projected 3D object center on the image. When the projected center is outside the image, we set the target to be the intersection between image edge and the line from 2D center to 3D projected center. We estimate the uncertainty of each bounding box as in MonoFlex. However, instead of ensemble the depth from keypoints and regression, we only used the regressed depth directly. The edge fusion module in MonoFlex is removed for simplicity. We additionally predict the 2D bounding box $\mathbf{b}_k^t$ for each object for the object-centric feature extraction in the second stage.

4.1 Object-Centric Temporal Correspondence Learning

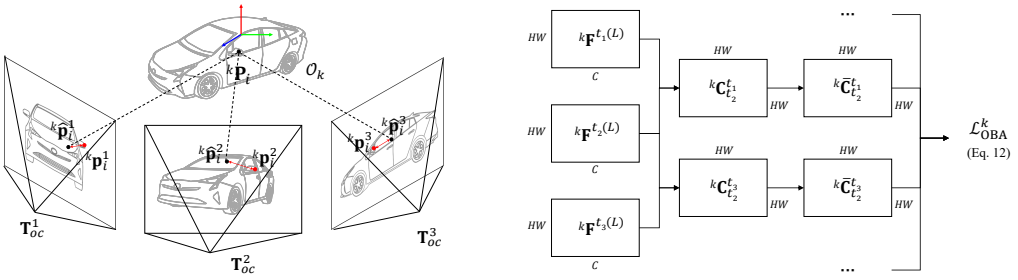
We propose an object-centric temporal correspondence learning (*OTCL*) module as the second stage object detector. Specifically, the OTCL module is designed to learn the correspondence of the dense features for the same object among all available frames. Given a video $\{I_1, I_2, \cdots, I_T\}$ and image features $\{\mathbf{F}^1, \mathbf{F}^2, \cdots, \mathbf{F}^T\}$ from the backbone in the first stage, we extract the RoI features ${}^k\mathbf{F}^t \in \mathbb{R}^{H\times W\times C}$ of the object $\mathcal{O}_k^t$ by the RoIAlign operation (He et al., 2017),

$$ {}^k\mathbf{F}^t = \texttt{RoIAlign}(\mathbf{F}^t, \mathbf{b}_k^t). \tag{3} $$

We apply L layers of cross- and self-attention operations before calculating the correspondence map to aggregate the spatial and temporal information for RoI features. For each layer of attention operations between two adjacent frames t and t':

$$ \begin{cases} {}^k\widetilde{\mathbf{F}}^t = \texttt{Att}_\texttt{S}(Q, K, V) = \texttt{Att}_\texttt{S}({}^k\hat{\mathbf{F}}^t, {}^k\hat{\mathbf{F}}^t, {}^k\hat{\mathbf{F}}^t), \\ {}^k\widetilde{\mathbf{F}}^{t'} = \texttt{Att}_\texttt{S}(Q, K, V) = \texttt{Att}_\texttt{S}({}^k\hat{\mathbf{F}}^{t'}, {}^k\hat{\mathbf{F}}^{t'}, {}^k\hat{\mathbf{F}}^{t'}), \\ {}^k\hat{\mathbf{F}}^{t'} = \texttt{Att}_\texttt{T}(Q, K, V) = \texttt{Att}_\texttt{T}({}^k\widetilde{\mathbf{F}}^{t'}, {}^k\widetilde{\mathbf{F}}^{t}, {}^k\widetilde{\mathbf{F}}^{t}), \end{cases} \tag{4} $$

where ${}^k\hat{\mathbf{F}}^t \in \mathbb{R}^{HW\times C}$ is the flattened RoI feature, $\texttt{Att}_\texttt{S}$ is the spatial self-attention, $\texttt{Att}_\texttt{T}$ is the temporal cross-attention.



(a) Object-centric Bundle Adjustment (OBA). (b) The computation of the featuremetric OBA loss.

Figure 3: **Illustration of object-centric featuremetric bundle adjustment loss.**

We then define the spatial correspondence map between two flattened RoI features. In frame pair (t, t'), we use to $\mathbf{f}_i$ to denote i-th feature in ${}^k\hat{\mathbf{F}}^{(L)}$ ($i \in \{1, 2, \cdots, HW\}$). The correspondence map ${}^k\mathbf{C}_t^{t'} \in \mathbb{R}^{HW \times HW}$ for $\mathcal{O}_k^t$ in two frames is defined as the correlation of two RoI features in two frames:

$$ {}^k\mathbf{C}_t^{t'}[i, i'] = {}^k\mathbf{f}_i^t * {}^k\mathbf{f}_{i'}^{t'} \tag{5}$$

To normalize the correspondence map, we perform softmax over all spatial locations i',

$$ {}^k\bar{\mathbf{C}}_t^{t'}[i, i'] = \texttt{softmax}({}^k\mathbf{C}_t^{t'}[i, i']). \tag{6}$$

4.2 Object-Centric Featuremetric Bundle Adjustment Loss

First, we need to revisit the object-centric bundle adjustment (OBA), as shown in Fig. 3a. As proposed in Object SLAM (Yang & Scherer, 2019; Li et al., 2018), OBA assumes that the object can only have rigid motion relative to the camera. For the object $\mathcal{O}_k$, given the 3D points $\mathcal{P}_k = \{{}^k\mathbf{P}_i\}_{i=1}^m$ on the object, 2D points $\{{}^k\mathbf{p}_i^t\}_{i=1}^m$, 2D features $\{\mathbf{f}[{}^k\mathbf{p}_i^t]\}_{i=1}^m$ at position ${}^k\mathbf{p}_i^t$, and the camera pose $\mathcal{T}_k = \{{}^k\mathbf{T}_{oc}^t\}_{t=1}^T$ in the object reference frame, OBA can be casted as:

$$ \bar{\mathcal{T}}_k, \bar{\mathcal{P}}_k = \underset{\mathcal{T}_k, \mathcal{P}_k}{\arg\min} \frac{1}{2} \sum_{i=1}^{m} \sum_{t=1}^{T} ||{}^k\mathbf{p}_i^t - \Pi({}^k\mathbf{T}_{oc}^t, {}^k\mathbf{P}_i, \mathbf{K})||_2^2. \tag{7}$$

To make the OBA layer end-to-end learnable, we formulate featuremetric (Lindenberger et al., 2021) OBA:

$$ \bar{\mathcal{T}}_k, \bar{\mathcal{P}}_k = \underset{\mathcal{T}_k, \mathcal{P}_k}{\arg\min} \frac{1}{2} \sum_{i=1}^{m} \sum_{t=1}^{T} ||\mathbf{f}[{}^k\mathbf{p}_i^t] - \mathbf{f}[\Pi({}^k\mathbf{T}_{oc}^t, {}^k\mathbf{P}_i, \mathbf{K})]||_2^2. \tag{8}$$

During training, we use the ground-truth object pose in Eq. 8 to learn the correspondence between 2D features. And at inference, we estimate the object pose from the matched 2D features in different frames by minimizing the reprojection error. First, considering the featuremetric reprojection error of frame t

$$ {}^k e_i^t = \sum_{t'=1}^{T} \mathbf{f}[{}^k\mathbf{p}_i^t] - \mathbf{f}[{}^k\mathbf{p}_i^{t'}] \tag{9}$$

$$ = \sum_{t'=1}^{T} \mathbf{f}[{}^k\mathbf{p}_i^t] - \mathbf{f}[\Pi({}^k\bar{\mathbf{T}}_{oc}^t, \Pi^{-1}({}^k\bar{\mathbf{T}}_{oc}^{t'}, {}^k\hat{\mathbf{p}}_i^{t'}, \mathbf{K}, \hat{z}_i^{t'}), \mathbf{K})], \tag{10}$$

where ${}^k\hat{\mathbf{p}}_i^{t'} = \mathcal{H}_\theta(\mathbf{f}[{}^k\mathbf{p}_i^t])$ is the predicted 2D coordinates in frame t' corresponding to $\mathbf{f}[{}^k\mathbf{p}_i^t]$ from network $\mathcal{H}_\theta$. $\Pi^{-1}(\cdot)$ is the inverse projection function to lift the image point to 3D in the object frame. $\hat{z}_i^{t'}$ is the ground-truth depth of ${}^k\hat{\mathbf{p}}_i^{t'}$. ${}^k\bar{\mathbf{T}}_{oc}^t$ and ${}^k\bar{\mathbf{T}}_{oc}^{t'}$ denotes the ground-truth pose of object $\mathcal{O}_k$ in t frame and t' frame, respectively. The featuremetric reprojection loss can be formulated as

$$ \mathcal{L}_{\text{rep}}^k = \sum_{i=1}^{m} \sum_{t=1}^{T} ||{}^k e_i^t||_2^2 = \sum_{i=1}^{m} \sum_{t=1}^{T} \sum_{t'=1}^{T} ||{}^k\mathbf{f}_i^t - {}^k\mathbf{f}_i^{t'}||_2^2 \tag{11}$$

Finally, from Eq. 11, because we obtain the normalized correspondence map $\bar{\mathbf{C}}$ in Sec. 4.1, we use the cosine distance to measure featuremetric reprojection error. With log-likelihood, we formulate the featuremetric OBA loss to supervise the correspondence map:

$$\mathcal{L}_{\text{OBA}}^{k} = -\sum_{i=1}^{m}\sum_{t=1}^{T}\sum_{t'=1}^{T}\log({}^{k}\bar{\mathbf{C}}_{t}^{t'}[{}^{k}\bar{\mathbf{p}}_{i}^{t}, {}^{k}\bar{\mathbf{p}}_{i}^{t'}]). \tag{12}$$

where, $({}^{k}\bar{\mathbf{p}}_{i}^{t}, {}^{k}\bar{\mathbf{p}}_{i}^{t'})$ are the ground-truth corresponding pair of the i-th local feature. The illustration of the loss computation is in Fig. 3b.

4.3 Inference

Single frame 3D object detection. First, we use the first stage object detector to generate single frame proposals, including classes, 2D bounding boxes, and 3D bounding boxes. The 2D boxes are used for the RoI feature extraction, and the 3D boxes are treated as the initial camera pose of the object-centric bundle adjustment.

Object association in 3D space. We follow ImmortalTracker (Wang et al., 2021b), applying the 3D Kalman Filter to associate the boxes predicted from our first stage object detector frame by frame. To make the tracklet robust to the measurement noise and track the object as long as possible, we keep all tracklets as association candidates no matter how long ago the tracklet disappeared. Besides, we ignore the strict condition of tracklet birth, because the bad measurement always belongs to the short tracklet that we will not optimize.

Dense feature matching. To optimize the object pose, we need to obtain the feature correspondence in each frame for the same object. As mentioned in Sec. 4.1, our second-stage object detector can generate a dense correspondence map in all frames. During inference, we match all $H \times W$ dense local features on the RoI feature in the adjacent two frames and first-to-last frames in the time sliding window τ. We use the RANSAC algorithm (Fischler & Bolles, 1981) to filter the outlier. To balance the number of valid features in each frame, we select the top k features for each frame. Note that if the feature number for each frame is not balanced, the optimization will tend to give high weight excessively to the frames with more tracklets and make other frames deviate from the correct pose.

Feature tracking. Then the matched feature pairs are constructed into a graph $\mathcal{G}$. The features are on the vertices. If the features are matched, an edge is connected in the graph. Then we track the feature for the object in all available frames. We use the association method mainly following (Dusmanu et al., 2020). The graph partitioning method is applied to G to make each connected subgraph have the most one vertex per frame. The graph cut is based on the similarity of the matched features.

Object-centric bundle adjustment. Finally, we solve the object-centric global optimization by Levenberg–Marquardt algorithm, and the object pose in each frame and the 3D position of the feature can be global optimized.

Post-processing. We also apply some common post-processing in video object detection techniques like tracklet rescoring (Kang et al., 2017) and bounding box interpolation.

4.4 Discussions

We make some discussions about some previous methods related to BA-Det in this section.

- **BA-Det vs. multi-view geometry-based methods (Wang et al., 2022a).** Although they also try to utilize geometry in the video, they treat continuous frames as stereo to estimate the depth of each frame, ignoring the dynamic objects. So, they add the monocular depth estimator as the residual branch to compensate for the shortage. However, because we separate different objects and treat them as different tracklets, we can easily handle moving objects. With the long time range, objects can move very far from the first frame. We can utilize all available frames, but they only use frames within a very short time window.
- **BA-Det vs. differentiable bundle adjustment, like BANet (Tang & Tan, 2019).** Both of us train a network with bundle adjustment loss in an end-to-end style. However, we focus on the object level instead of the whole scene. The purpose of BA-Det is to optimize the object pose in

Table 1: **The results on WODv1.2 Sun et al. (2020) *val* set.** AP_{70} denotes AP with IoU threshold at 0.7. AP_{50} denotes AP IoU@0.5.† denotes the method utilizing temporal information.

	LEVEL_1				LEVEL_2			
	3D AP_{70}	3D APH_{70}	3D AP_{50}	3D APH_{50}	3D AP_{70}	3D APH_{70}	3D AP_{50}	3D APH_{50}
M3D-RPN Brazil & Liu (2019)	0.35	0.34	3.79	3.63	0.33	0.33	3.61	3.46
PatchNet Ma et al. (2020)	0.39	0.37	2.92	2.74	0.38	0.36	2.42	2.28
PCT Wang et al. (2021a)	0.89	0.88	4.20	4.15	0.66	0.66	4.03	3.99
MonoJSG Lian et al. (2022)	0.97	0.95	5.65	5.47	0.91	0.89	5.34	5.17
GUPNet Lu et al. (2021)	2.28	2.27	10.02	9.94	2.14	2.12	9.39	9.31
DEVIANT Kumar et al. (2022)	2.69	2.67	10.98	10.89	2.52	2.50	10.29	10.20
CaDDN Reading et al. (2021)	5.03	4.99	17.54	17.31	4.49	4.45	16.51	16.28
DID-M3D Peng et al. (2022)	-	-	20.66	20.47	-	-	19.37	19.19
BEVFormer Li et al. (2022b)†	-	7.70	-	30.80	-	6.90	-	27.70
DCD Li et al. (2022a)	12.57	12.50	33.44	33.24	11.78	11.72	31.43	31.25
MonoFlex Zhang et al. (2021) (Baseline)	11.70	11.64	32.26	32.06	10.96	10.90	30.31	30.12
BA-Det(Ours)†	**16.60**	**16.45**	**40.93**	**40.51**	**15.57**	**15.44**	**38.53**	**38.12**

each frame, i.e., for temporal 3D object detection. So, we combine object detection and object-centric local feature learning into an integrated framework and treat temporal feature learning as the second stage of object detection.

5 EXPERIMENTS

5.1 DATASETS AND METRICS

We conduct our experiments on the large-scale autonomous driving dataset, Waymo Open Dataset (WOD) (Sun et al., 2020) v1.2. Five cameras are available in WOD, and following existing methods (Reading et al., 2021; Li et al., 2022a), we only train and evaluate using the images captured from the FRONT camera. Because we mainly focus on rigid objects, we report the results of the VEHICLE class, which is also the mainstream experiment setting. The evaluation metrics are 3D AP and APH (AP weighted by heading accuracy) under the IoU threshold of 0.7 and 0.5. WOD has two difficulty levels. LEVEL_1 is easy, ignoring some ground truth bounding boxes that may be heavily occluded or far from the ego-vehicle. LEVEL_2 includes all ground truth.

5.2 IMPLEMENTATION DETAILS

The first stage network architecture of BA-Det is the same as MonoFlex, with DLA-34 (Yu et al., 2018) backbone, the output feature map is with the stride of 8. In the second stage, the shape of the RoI feature is 60×80. The spatial and temporal attention module is stacked with 4 layers. The implementation is based on the PyTorch framework. We train our model on 8 NVIDIA RTX 3090 GPUs for 14 epochs. Adam optimizer is applied with $\beta_1 = 0.9$ and $\beta_2 = 0.999$. The initial learning rate is 5×10^{-4} and weight decay is 10^{-5}. The learning rate scheduler is one cycle. We use the Levenberg-Marquardt algorithm, implemented by DeepLM (Huang et al., 2021), to solve object-centric bundle adjustment. The maximum iteration of the LM algorithm is 200. For the object which appears in less than 10 frames or the object on which the average keypoint number per frame is less than 5, we do not optimize it.

5.3 COMPARISONS WITH STATE-OF-THE-ART METHODS

We compare our BA-Det with other state-of-the-art methods on WODv1.2 val set. As shown in 1, using the FRONT camera, we outperform the SOTA method DCD (Li et al., 2022a) for about 4AP and 4APH ($\sim$30% improvement) under the 0.7 IoU threshold. Compared with the only temporal method BEVFormer (Li et al., 2022b), we have double points of 3D AP_{70} and 3D APH_{70}. Compared with our baseline MonoFlex (Zhang et al., 2021), we have a gain of 50% evaluated with 3D AP_{70}, thanks to the object-centric global optimization in multi-frames.

5.4 DISTANCE CONDITIONED RESULTS

We report the results with the different depth ranges in Table 2. The results indicate that the single frame methods, like DCD and MonoFlex, are seriously affected by the object distance. When the

Table 2: **The object depth range conditioned result on WODv1.2 Sun et al. (2020) *val* set.** L1 and L2 denote LEVEL_1 and LEVEL_2 difficulty, respectively. †: use temporal information.

	Method	3D AP_{70}			3D APH_{70}			3D AP_{50}			3D APH_{50}		
		0-30	30-50	50-∞	0-30	30-50	50-∞	0-30	30-50	50-∞	0-30	30-50	50-∞
L1	DCD Li et al. (2022a)	32.47	5.94	1.24	32.30	5.91	1.23	62.70	26.35	10.16	62.35	26.21	10.09
	MonoFlex Zhang et al. (2021)	30.64	5.29	1.05	30.48	5.27	1.04	61.13	25.85	9.03	60.75	25.71	8.95
	BA-Det(Ours)†	**37.74**	**11.04**	**3.86**	**37.46**	**10.95**	**3.79**	**71.07**	**37.15**	**14.89**	**70.46**	**36.79**	**14.61**
L2	DCD Li et al. (2022a)	32.30	5.76	1.08	32.19	5.73	1.08	62.48	25.60	8.92	62.13	25.46	8.86
	MonoFlex Zhang et al. (2021)	30.54	5.14	0.91	30.37	5.11	0.91	60.91	25.11	7.92	60.54	24.97	7.85
	BA-Det(Ours)†	**37.61**	**10.72**	**3.37**	**37.33**	**10.63**	**3.31**	**70.83**	**36.14**	**13.62**	**70.23**	**35.79**	**13.37**

Table 3: **Ablation study of each component in BA-Det.**

	LEVEL_1				LEVEL_2			
	3D AP_{70}	3D APH_{70}	3D AP_{50}	3D APH_{50}	3D AP_{70}	3D APH_{70}	3D AP_{50}	3D APH_{50}
MonoFlex (baseline)	11.70	11.64	32.26	32.06	10.96	10.90	30.31	30.12
Our first stage prediction	13.57	13.48	34.70	34.43	12.72	12.64	32.56	32.32
+3D Tracking	14.01	13.93	35.19	34.92	13.13	13.05	33.03	32.78
+ Learnable global optimization	15.85	15.75	38.06	37.76	14.87	14.77	35.72	35.44
+ Tracklet rescoring	16.43	16.30	40.07	39.70	15.41	15.29	37.66	37.31
+ Bbox interpolation	16.60	16.45	40.93	40.51	15.57	15.44	38.53	38.12

object is farther away from the ego-vehicle, the detection performance drops sharply. However, with BA-Det, the gain is almost from the object far away from the ego-vehicle. The 3D AP_{70} and 3D APH_{70} are $3\times$ compared with the baseline when the object is located in $[50\text{m}, \infty)$, $2\times$ in $[30\text{m}, 50\text{m})$ and only $1.2\times$ in $[0\text{m}, 30\text{m})$. This is because we utilize the long-range temporal information for each object. In the same tracklet, the near measurement can help to refine the measurement far away.

5.5 Ablation study

We ablate each component of BA-Det. The results are shown in Table 3 The first stage detector is slightly better than the MonoFlex baseline mainly because we remove the edge fusion module, which is harmful to the truncated objects in WOD. The 3D Tracking step not only associates the objects in different frames but also smooths the object's trajectory. This part of improvement can be regarded as the gain from the Kinematic3D (Brazil et al., 2020). The core of BA-Det is the learnable global optimization module, which obtains the largest increase in all modules. The tracklet-level rescoring module and tracklet interpolation module are also useful.

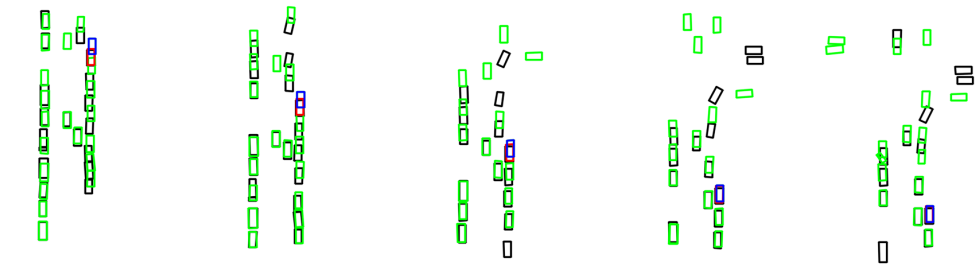
5.6 Further Discussions

BA vs. Object BA. We experiment to discuss whether the object-centric manner is important in temporal global optimization. We modify our pipeline and optimize the whole scene in the Global frame instead of optimizing the object pose in the Object frame, called Static BA in Table 4. Static BA ignores dynamic objects and treats them the same as static objects. Even if there is no problem dealing with static objects, the inability for dynamic objects causes performance loss. Compared with ours, Static BA decreases about 2 AP.

Feature correspondence. As shown in Table 5, we ablate the features used for object-centric bundle adjustment. Compared with traditional ORB feature (Rublee et al., 2011), widely used in SLAM,

Table 4: **Comparison between object-centric BA-Det and the traditional scene-level bundle adjustment (Static BA).** Initial prediction denotes the predictions in the first stage.

	LEVEL_1				LEVEL_2			
	3D AP_{70}	3D APH_{70}	3D AP_{50}	3D APH_{50}	3D AP_{70}	3D APH_{70}	3D AP_{50}	3D APH_{50}
MonoFlex (baseline)	11.70	11.64	32.26	32.06	10.96	10.90	30.31	30.12
Initial prediction	13.57	13.48	34.70	34.43	12.72	12.64	32.56	32.32
Static BA	14.73	14.62	37.89	37.56	13.82	13.72	35.65	35.34
Ours	16.60	16.45	40.93	40.51	15.57	15.44	38.53	38.12



(a) Frame 8. (b) Frame 22. (c) Frame 36. (d) Frame 50. (e) Frame 57.

Figure 4: **Qualitative results from the BEV in different frames.** We use blue and red boxes to denote initial predictions and optimized predictions of the object we highlight. The green and black boxes denote the other boxes and the ground truth. The lower an object in the figure, the closer to the ego-vehicle.

Table 5: **Ablation study about different feature corresponding methods.** $\bar{L}_t$ denotes the average feature tracklet length for each object.

	$\bar{L}_t$	LEVEL_1				LEVEL_2			
		3D AP_{70}	3D APH_{70}	3D AP_{50}	3D APH_{50}	3D AP_{70}	3D APH_{70}	3D AP_{50}	3D APH_{50}
MonoFlex (baseline)	-	11.70	11.64	32.26	32.06	10.96	10.90	30.31	30.12
BA-Det+ ORB feature Rublee et al. (2011)	2.6	14.05	13.96	35.21	34.95	13.17	13.08	33.05	32.81
BA-Det+ Our feature	10	16.60	16.45	40.93	40.51	15.57	15.44	38.53	38.12

our feature learning module predicts denser and better correspondence. We find the total tracklet length is 19.6 frames, and the average feature tracklet in our method is about 10 frames, which means we can keep a long feature dependency and better utilize long-range temporal information. However, the ORB feature's average tracklet length is only 2.6 frames. The results show the short feature tracklet can not refine the long-term object pose well.

Inference latency of each step in BA-Det. The inference latency of each step is shown in Table 6. The most time-consuming part is the first stage object detector, more than 130ms per image, which is the same as the MonoFlex baseline. With all steps in BA-Det, the total inference latency is about 181.5ms. In other words, our BA-Det only takes 50ms more per image, compared with the single-frame detector MonoFlex. Besides, although the dense feature correspondence is calculated, thanks to the shared backbone with the first stage detector and batched process for the objects, the feature correspondence module is not very time-consuming.

Table 6: **Inference latency of each step in BA-Det per image.**

Total latency	181.5ms
First stage	132.6ms
Object tracking	6.6ms
Correspondence	23.0ms
Optimization	19.3ms

5.7 Qualitative Results

In Fig. 4, we show the object-level qualitative results of the first stage predictions and the second stage predictions in different frames. For a tracklet, we can refine the bounding box predictions with better measurements in other frames, even if there is a long time interval between them.

6 Conclusion

In this paper, we propose a video-based 3D object detection paradigm with long-term temporal visual correspondence, called BA-Det. BA-Det is a two-stage object detector that can jointly learn object detection and temporal feature correspondence with proposed Featuremetric OBA Loss. Object-centric bundle adjustment optimizes the first-stage object estimation globally in each frame. BA-Det achieves state-of-the-art performance. Experiment results on the Waymo Open Dataset (WOD) show the effectiveness and efficiency of our method.

REFERENCES

Sameer Agarwal, Keir Mierle, and The Ceres Solver Team. Ceres Solver, 3 2022. URL https://github.com/ceres-solver/ceres-solver.

Garrick Brazil and Xiaoming Liu. M3d-rpn: Monocular 3d region proposal network for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9287–9296, 2019.

Garrick Brazil, Gerard Pons-Moll, Xiaoming Liu, and Bernt Schiele. Kinematic 3d object detection in monocular video. In *European Conference on Computer Vision*, pp. 135–152. Springer, 2020.

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 *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11621–11631, 2020.

Xuesong Chen, Shaoshuai Shi, Benjin Zhu, Ka Chun Cheung, Hang Xu, and Hongsheng Li. Mppnet: Multi-frame feature intertwining with proxy points for 3d temporal object detection. In *European Conference on Computer Vision*, 2022.

Yilun Chen, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Dsgn: Deep stereo geometry network for 3d object detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12536–12545, 2020.

Mihai Dusmanu, Johannes L Schönberger, and Marc Pollefeys. Multi-view optimization of local feature geometry. In *European Conference on Computer Vision*, pp. 670–686. Springer, 2020.

Lue Fan, Ziqi Pang, Tianyuan Zhang, Yu-Xiong Wang, Hang Zhao, Feng Wang, Naiyan Wang, and Zhaoxiang Zhang. Embracing single stride 3d object detector with sparse transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8458–8468, 2022.

Martin A Fischler and Robert C Bolles. Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24 (6):381–395, 1981.

Xiaoyang Guo, Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Liga-stereo: Learning lidar geometry aware representations for stereo-based 3d detector. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3153–3163, 2021.

Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pp. 2961–2969, 2017.

Jingwei Huang, Shan Huang, and Mingwei Sun. Deeplm: Large-scale nonlinear least squares on deep learning frameworks using stochastic domain decomposition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10308–10317, 2021.

Junjie Huang and Guan Huang. Bevdet4d: Exploit temporal cues in multi-camera 3d object detection. *arXiv preprint arXiv:2203.17054*, 2022.

Rudolph Emil Kalman. A new approach to linear filtering and prediction problems. 1960.

Kai Kang, Hongsheng Li, Junjie Yan, Xingyu Zeng, Bin Yang, Tong Xiao, Cong Zhang, Zhe Wang, Ruohui Wang, Xiaogang Wang, et al. T-cnn: Tubelets with convolutional neural networks for object detection from videos. *IEEE Transactions on Circuits and Systems for Video Technology*, 28(10):2896–2907, 2017.

Abhinav Kumar, Garrick Brazil, Enrique Corona, Armin Parchami, and Xiaoming Liu. Deviant: Depth equivariant network for monocular 3d object detection. *arXiv preprint arXiv:2207.10758*, 2022.

Rainer Kümmerle, Giorgio Grisetti, Hauke Strasdat, Kurt Konolige, and Wolfram Burgard. g 2 o: A general framework for graph optimization. In *2011 IEEE International Conference on Robotics and Automation*, pp. 3607–3613. IEEE, 2011.

Peiliang Li, Tong Qin, et al. Stereo vision-based semantic 3d object and ego-motion tracking for autonomous driving. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 646–661, 2018.

Yingyan Li, Yuntao Chen, Jiawei He, and Zhaoxiang Zhang. Densely constrained depth estimator for monocular 3d object detection. In *European Conference on Computer Vision*. Springer, 2022a.

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 (ECCV)*, 2022b.

Qing Lian, Peiliang Li, and Xiaozhi Chen. Monojsg: Joint semantic and geometric cost volume for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1070–1079, 2022.

Philipp Lindenberger, Paul-Edouard Sarlin, Viktor Larsson, and Marc Pollefeys. Pixel-perfect structure-from-motion with featuremetric refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5987–5997, 2021.

Yingfei Liu, Junjie Yan, Fan Jia, Shuailin Li, Qi Gao, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petrv2: A unified framework for 3d perception from multi-camera images. *arXiv preprint arXiv:2206.01256*, 2022.

David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

Yan Lu, Xinzhu Ma, Lei Yang, Tianzhu Zhang, Yating Liu, Qi Chu, Junjie Yan, and Wanli Ouyang. Geometry uncertainty projection network for monocular 3d object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3111–3121, 2021.

Xinzhu Ma, Shinan Liu, Zhiyi Xia, Hongwen Zhang, Xingyu Zeng, and Wanli Ouyang. Rethinking pseudo-lidar representation. In *European Conference on Computer Vision*, pp. 311–327. Springer, 2020.

Raul Mur-Artal, Jose Maria Martinez Montiel, and Juan D Tardos. Orb-slam: a versatile and accurate monocular slam system. *IEEE transactions on robotics*, 31(5):1147–1163, 2015.

Lachlan Nicholson, Michael Milford, and Niko Sünderhauf. Quadricslam: Dual quadrics from object detections as landmarks in object-oriented slam. *IEEE Robotics and Automation Letters*, 4 (1):1–8, 2018.

Liang Peng, Xiaopei Wu, Zheng Yang, Haifeng Liu, and Deng Cai. Did-m3d: Decoupling instance depth for monocular 3d object detection. In *ECCV*, 2022.

Charles R Qi, Yin Zhou, Mahyar Najibi, Pei Sun, Khoa Vo, Boyang Deng, and Dragomir Anguelov. Offboard 3d object detection from point cloud sequences. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6134–6144, 2021.

Cody Reading, Ali Harakeh, Julia Chae, and Steven L Waslander. Categorical depth distribution network for monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8555–8564, 2021.

Ethan Rublee, Vincent Rabaud, Kurt Konolige, and Gary Bradski. Orb: An efficient alternative to sift or surf. In *2011 International conference on computer vision*, pp. 2564–2571. Ieee, 2011.

Paul-Edouard Sarlin, Daniel DeTone, Tomasz Malisiewicz, and Andrew Rabinovich. Superglue: Learning feature matching with graph neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4938–4947, 2020.

Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016.

Jiaming Sun, Zehong Shen, Yuang Wang, Hujun Bao, and Xiaowei Zhou. Loftr: Detector-free local feature matching with transformers. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8922–8931, 2021.

Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2446–2454, 2020.

Chengzhou Tang and Ping Tan. BA-net: Dense bundle adjustment networks. In *International Conference on Learning Representations*, 2019.

Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. Bundle adjustment—a modern synthesis. In *International workshop on vision algorithms*, pp. 298–372. Springer, 1999.

Li Wang, Li Zhang, Yi Zhu, Zhi Zhang, Tong He, Mu Li, and Xiangyang Xue. Progressive coordinate transforms for monocular 3d object detection. *Advances in Neural Information Processing Systems*, 34, 2021a.

Qitai Wang, Yuntao Chen, Ziqi Pang, Naiyan Wang, and Zhaoxiang Zhang. Immortal tracker: Tracklet never dies. *arXiv preprint arXiv:2111.13672*, 2021b.

Tai Wang, Jiangmiao Pang, and Dahua Lin. Monocular 3d object detection with depth from motion. In *European Conference on Computer Vision (ECCV)*, 2022a.

Zengran Wang, Chen Min, Zheng Ge, Yinhao Li, Zeming Li, Hongyu Yang, and Di Huang. Sts: Surround-view temporal stereo for multi-view 3d detection. *arXiv preprint arXiv:2208.10145*, 2022b.

Haiping Wu, Yuntao Chen, Naiyan Wang, and Zhaoxiang Zhang. Sequence level semantics aggregation for video object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 9217–9225, 2019.

Linjie Yang, Yuchen Fan, and Ning Xu. Video instance segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 5188–5197, 2019.

Shichao Yang and Sebastian Scherer. Cubeslam: Monocular 3-d object slam. *IEEE Transactions on Robotics*, 35(4):925–938, 2019.

Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11784–11793, 2021.

Yurong You, Katie Z Luo, Xiangyu Chen, Junan Chen, Wei-Lun Chao, Wen Sun, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Hindsight is 20/20: Leveraging past traversals to aid 3d perception. In *International Conference on Learning Representations*, 2022.

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

Yunpeng Zhang, Jiwen Lu, and Jie Zhou. Objects are different: Flexible monocular 3d object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3289–3298, 2021.

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

Xizhou Zhu, Yujie Wang, Jifeng Dai, Lu Yuan, and Yichen Wei. Flow-guided feature aggregation for video object detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 408–417, 2017.