

Divide and Conquer: Improving Multi-Camera 3D Perception with 2D Semantic-Depth Priors and Input-Dependent Queries

Qi Song, Qingyong Hu[✉], Chi Zhang, Yongquan Chen, Rui Huang[✉]

Abstract—3D perception tasks, such as 3D object detection and Bird’s-Eye-View (BEV) segmentation using multi-camera images, have drawn significant attention recently. Despite the fact that accurately estimating both semantic and 3D scene layouts are crucial for this task, existing techniques often neglect the synergistic effects of semantic and depth cues, leading to the occurrence of classification and position estimation errors. Additionally, the input-independent nature of initial queries also limits the learning capacity of Transformer-based models. To tackle these challenges, we propose an input-aware Transformer framework that leverages Semantics and Depth as priors (named SDTR). Our approach involves the use of an S-D Encoder that explicitly models semantic and depth priors, thereby disentangling the learning process of object categorization and position estimation. Moreover, we introduce a Prior-guided Query Builder that incorporates the semantic prior into the initial queries of the Transformer, resulting in more effective input-aware queries. Extensive experiments on the nuScenes and Lyft benchmarks demonstrate the state-of-the-art performance of our method in both 3D object detection and BEV segmentation tasks.

Index Terms—3D object detection, Bird’s-Eye-View (BEV) segmentation, multi-camera, 3D perception.

I. INTRODUCTION

3D perception is a significant problem in computer vision with diverse applications, including autonomous driving [3] and robot navigation [4]. Of the various tasks involved in 3D perception, multi-camera 3D object detection [5]–[8] and Bird’s-Eye-View (BEV) segmentation [9]–[13] are two representatives that have drawn widespread attention. Despite their different objectives, these tasks aim to infer both *semantic categories* and *3D positions* from 2D cues given by multiple cameras, which makes them ill-posed and entangled with semantic and geometric understanding, presenting significant challenges.

There are two mainstream approaches to multi-camera 3D perception tasks: 1) Depth-based methods, which estimate pseudo-depth to project multi-view image features into 3D space [1], [14]–[16]. These methods focus on improving depth estimation quality but often overlook the role of semantic

cues in reducing classification errors and acting as priors for object localization. Consequently, they tend to underperform, where classification errors and localization errors often appear together, as seen in Figure 1 (b). 2) Transformer-based methods, which construct a set of randomly initialized object queries of 3D space [2], [8] or BEV space [17], [18] and retrieve relevant image features using a cross-attention mechanism without depth or semantic guidance. However, the input-independent nature of these queries (*i.e.*, all input images share the same object queries) makes training more difficult and reduces detection sensitivity to distant objects, as shown in Figure 1 (c).

In this paper, our objective is to develop an effective framework that addresses the challenges outlined above. We believe that *semantics and depth are equally essential to 3D perception but are implicitly learned and tightly coupled in existing networks*, restricting the full exploitation of valuable information. To overcome this limitation, we propose explicitly incorporating semantics and depth as prior knowledge to divide features for classification and position estimation. Additionally, we investigate strategies to make queries input-sensitive for transformer-based methods, alleviating the issues associated with their input-independent nature. Our findings suggest that certain prior knowledge can facilitate achieving this objective.

In particular, we present a transformer-based framework, named SDTR, which models both semantic and depth representations as prior knowledge. Our SDTR consists of two key designs: 1) an S-D Encoder with two branches to reason semantic and depth information contained in 2D images with explicit supervision, enabling the network to focus on relevant features and joint objectives; and 2) a Prior-guided Query Builder (PQB) that incorporates image-specific semantic guidance into the initial queries, transforming input-independent queries into input-aware queries, and improving the network’s perception capability in complex scenarios. The proposed SDTR is demonstrated to be highly accurate in deducing both semantic categories and 3D positions, as verified in Figure 1(d). Extensive experiments on the nuScenes and Lyft datasets further demonstrate the superiority of our method against other state-of-the-art 3D perception approaches. The main contributions of this paper can be summarized as follows:

- We propose SDTR, a transformer-based framework that incorporates semantic and depth priors to improve the capability of inferring both semantic categories and 3D positions.

This work was supported in part by Shenzhen Science and Technology Program under grant No. JCYJ20220818103006012 and ZDSYS20211021111415025. (Corresponding author: Qingyong Hu and Rui Huang.)

Qi Song, Chi Zhang, Yongquan Chen, and Rui Huang are with School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, Guangdong, 518172, China. Qingyong Hu is with the Department of Computer Science, University of Oxford, OX1 3PR, UK. (e-mail: qisong@link.cuhk.edu.cn; huqingyong15@outlook.com; chizhang1@link.cuhk.edu.cn; yqchen@cuhk.edu.cn; ruihuang@cuhk.edu.cn).

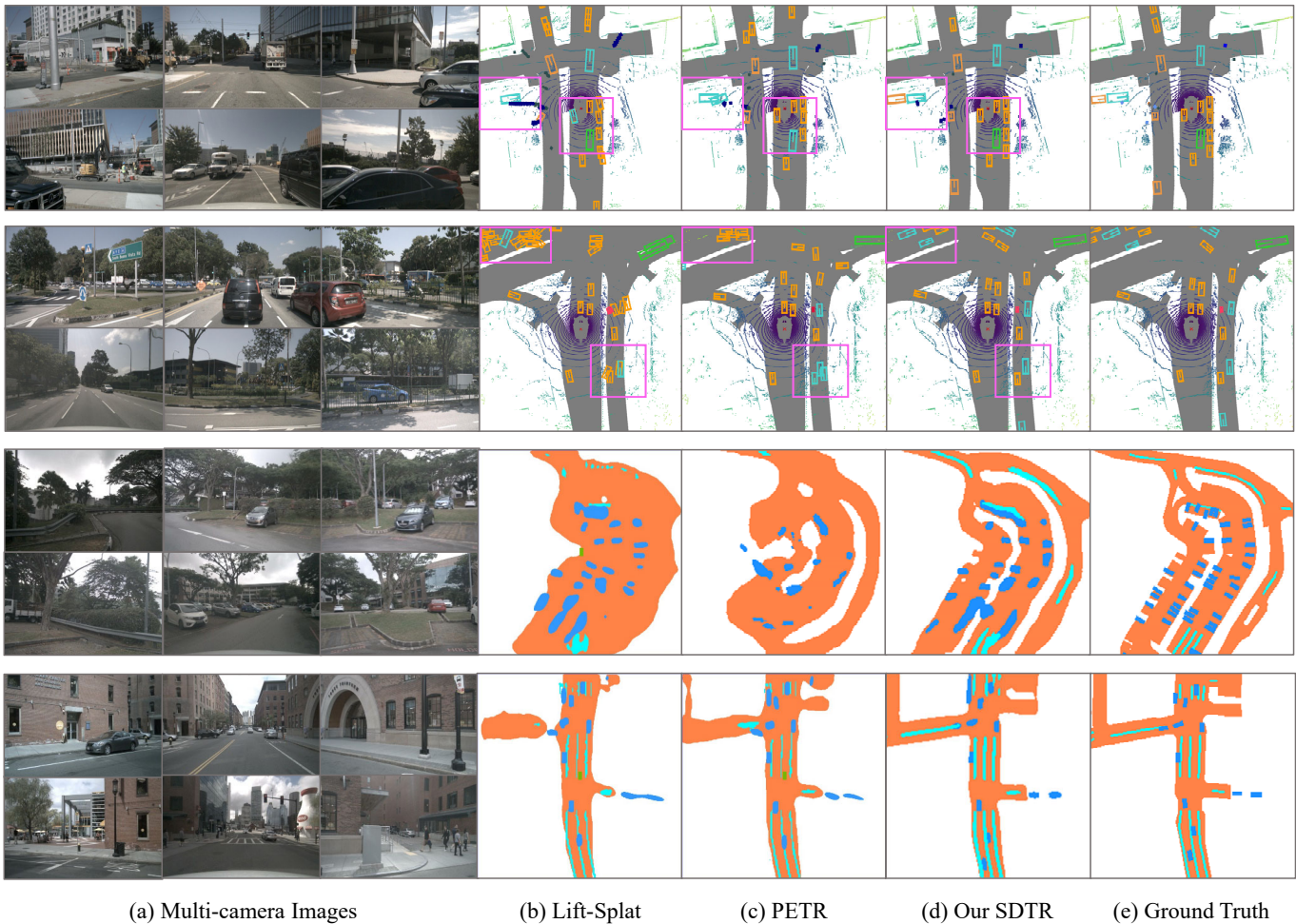


Fig. 1. **Illustration of the multi-camera 3D perception task.** Given the images collected by the cameras from different angles on the vehicle, this task aims to generate segmentation results from the BEV perspective and the 3D object detection results. (a) Multi-camera image inputs. (b) Lift-Splat [1] fails to leverage semantic clues in 3D perception, resulting in inaccurate predictions. (c) PETR [2] confuses the spatial arrangement and semantic categories of distant objects without image-specific guidance. (d) Our proposed SDTR model utilizes both semantic-depth priors and input-dependent queries, resulting in significantly improved predictions.

- We introduce an S-D Encoder with explicit supervision to capture both semantic and depth representations, while the Prior-guided Query Builder is designed to encode data-dependent semantic priors and generate input-aware queries.
- We demonstrate the effectiveness of our framework on two popular 3D perception tasks, including 3D object detection and BEV segmentation, with significant improvements over state-of-the-art methods on the nuScenes and Lyft datasets.

II. RELATED WORK

A. Multi-camera 3D Object Detection

Multi-camera 3D object detection is a challenging task that involves predicting multi-class 3D bounding boxes from multi-view images. Early work CenterNet [21] predicts 3D properties based on the center point of 2D boxes. Recently, transformer networks [22], [23] have shown promising results in reformulating object detection tasks by constructing a set of object queries and using cross-attention to search for relevant

image features. DETR3D [8], as a follow-up work of DETR [24], back-projects the 3D reference points into the image plane to index valid 2D features. PETR [2] perceives the 3D scene information by using the initialized object queries of 3D space. In the wake of rapid advancements in Bird’s Eye View (BEV) representation—a favored approach in navigation tasks due to its succinct 2D portrayal of the 3D environment—researchers have put forward to devise a set of BEV queries. These queries facilitate the transformation of perspective between BEV and the image features through cross attention. Both BEVSegFormer [18] and BEVFormer [17] employ BEV queries to extract valid features for their ultimate predictions. Nevertheless, the inherent input-independent nature of these queries inadvertently diminishes the detection sensitivity within intricate scenes. To address this, BEVFormerV2 [25] pioneers a two-stage BEV detector by integrating the first-stage proposals with learnable queries to form the second-stage object queries. However, these queries excessively depend on the precision of high-level 3D detection and continue to exhibit deficiencies in associating input information with learnable

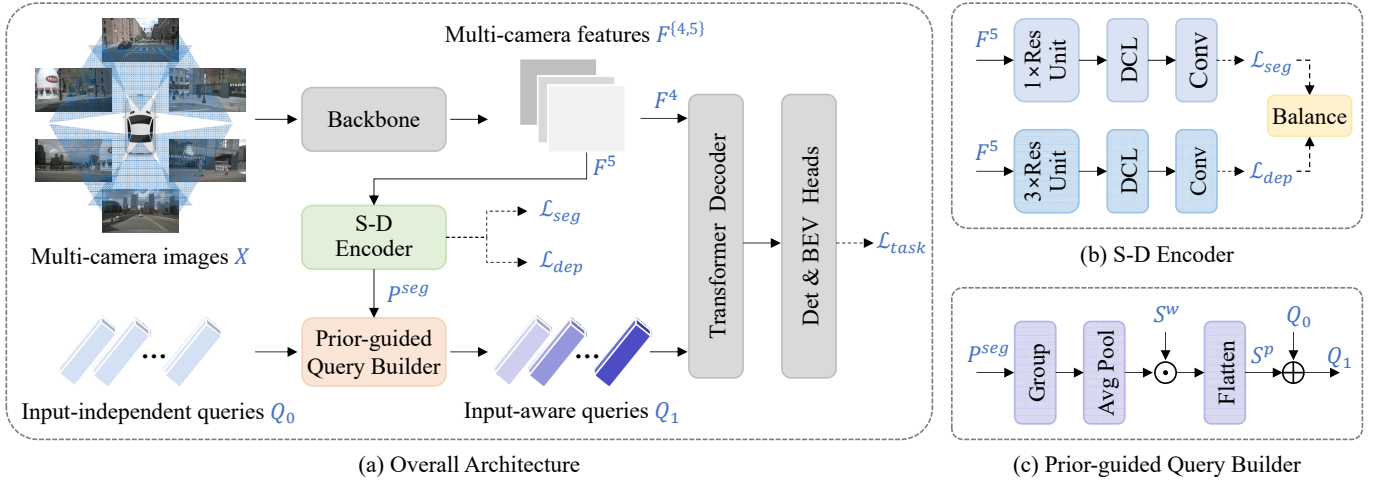


Fig. 2. **Overview of the proposed SDTR framework.** Our model comprises two key components, *i.e.*, the S-D Encoder and the Prior-guided Query Builder, which are designed to effectively extract semantic and depth representations and convert input-independent queries into input-aware queries, respectively. SDTR is capable of producing 3D detection and BEV segmentation results using task-specific heads. Specifically, ResUnit and DCL denote the residual unit in [19] and the dilated convolution layer in [20] respectively.

queries. To mitigate these limitations, we introduce a novel approach in this paper that involves the incorporation of global semantic priors into the initial queries. This strategy facilitates the generation of input-aware queries, thereby enhancing the flexibility and expressivity of the model.

B. BEV Segmentation

BEV segmentation is a task of segmenting objects in the bird’s eye view (BEV), which differs from 2D semantic segmentation [26]–[31] mainly due to the introduction of perspective shift, leading to ill-posed 2D-to-3D geometry inference. Traditional methods [32]–[34] use inverse perspective mapping (IPM) to project features from the image plane into the BEV plane. However, IPM only works well in estimating flat road layouts but inevitably introduces errors for 3D objects. To avoid errors introduced by IPM, a handful of works including VED [35] and VPN [36] directly learn the transformation relation between two planes using the multilayer perceptron. Nevertheless, these approaches damage the spatial information and harms the feature details. Recently, PON [37] and PanopticBEV [38] are proposed to use dense transformer layers to map the image features into the BEV space. Lift-Splat [1] utilizes the implicit depth distribution to lift multi-view images into 3D coordinates. Saha et al. [39] introduce a graph neural network to predict BEV objects from monocular images with spatial reasoning. Another line of work, such as CVT [40] and PYVA [41], adopts a cross-view transformer to implicitly learn geometric transformation. In this work, the cross-view transformer is chosen to map perspective view features into bird’s eye view for its strong expressiveness.

C. Auxiliary Learning for 3D Perception

Camera-based 3D perception attempts to understand the semantic layout of the environment and the 3D measurements of objects within it, all from 2D image inputs. This represents a highly intricate and inherently ill-posed learning

challenge. In a bid to alleviate convergence difficulties, recent research has been exploring auxiliary tasks as a means of providing comprehensive guidance for the backbone during feature extraction. One prominent avenue of this research utilizes the monocular depth estimation branch [42]–[46] to enhance the ability to interpret 3D geometric understanding from 2D imagery. Notably, BEVStereo [47] and SOLOFusion [48] exploit depth estimation to lift 2D image features into 3D space. Moreover, BEVDepth [49] integrates depth supervision to enhance depth prediction capabilities. Similarly, the recent work by Dwivedi et al. [50] proposes a novel transformation layer that effectively exploits depth maps to project 2D image features to the BEV space. DAT [51] incorporates depth information to improve the cross-attention mechanism and leverages depth-aware negative suppression loss to prevent duplicate predictions along depth axes. A parallel line of research seeks to enhance perception performance through auxiliary image detection. Works such as AutoAlign [52], MVX-Net [53], and M²BEV [54] incorporate a 2D detection head as an additional training signal. Meanwhile, BEVFormerV2 [25] constructs a 3D detection head upon the backbone to predict 3D bounding boxes in the perspective view. Furthermore, SimMOD [55] encompasses both 2D detection and 3D key information regression as auxiliary tasks. The commonality in these approaches is the focus on augmenting 3D geometric or 2D detection supervision to refine the backbone features. However, these methods often overlook the mutually beneficial relationship between the categorical cues and 3D geometry in 2D-to-3D perception. Given that the framework should be adaptable to different 3D perception tasks, *e.g.*, 3D detection, BEV segmentation, etc., this paper opts for a more nuanced semantic supervision, rather than simple 2D detection. In particular, we propose a novel framework that concurrently and comprehensively examines both semantic and depth information with explicit supervision to enhance representation learning. By utilizing semantic and depth priors, our model can more effectively comprehend the 3D scene geometry and

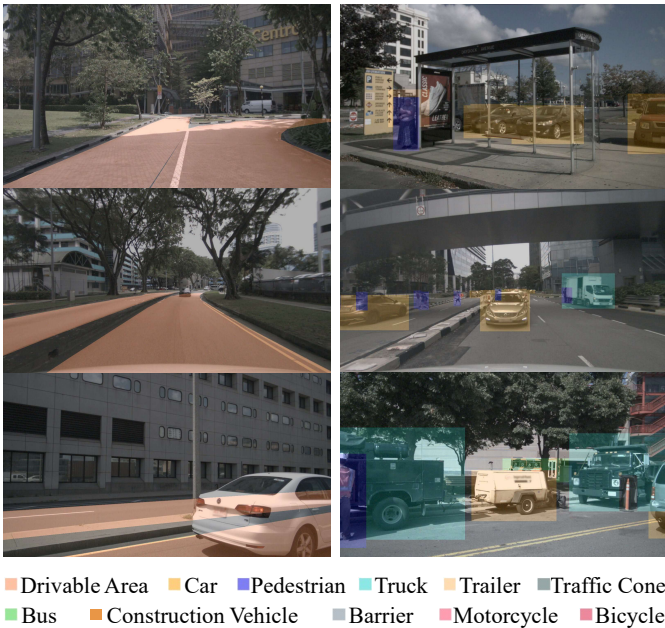


Fig. 3. **Visualization of the generated 2D semantic labels.** The first column exhibits the ground truth of the drivable area, while the subsequent column portrays the object-specific semantic labels. For clear illustration, the semantic labels have been superimposed on the original RGB images, thereby facilitating a more intuitive understanding.

provide more precise object segmentation in the BEV space.

III. METHOD

A. Overview

For the multi-camera 3D perception task, N multi-camera images $X = \{X_i \in \mathbb{R}^{3 \times H_i \times W_i}\}_N$ are given, each with associated extrinsic matrices $E = \{E_i \in \mathbb{R}^{3 \times 4}\}_N$, and intrinsic matrices $I = \{I_i \in \mathbb{R}^{3 \times 3}\}_N$. As depicted in Figure 2(a), we first pass the input images through a backbone network to extract multi-view image features, $F = \{F_i \in \mathbb{R}^{C \times H \times W}\}_N$. In particular, we use the output features of the 4th and 5th stages for subsequent processing. After that, the S-D Encoder is applied on F^5 to jointly predict 2D segmentation map $P^{seg} = \{P_i^{seg} \in \mathbb{R}^{C_s \times H \times W}\}_N$, and depth map $P^{dep} = \{P_i^{dep} \in \mathbb{R}^{C_d \times H \times W}\}_N$. The estimated 2D segmentation map P^{seg} is further utilized to interact with the initial object queries Q_0 in the Prior-guided Query Builder, enabling initial queries with awareness of class-wise semantics. Then, the newly generated queries Q_1 , along with the image features F^4 , are input to the transformer decoder. Finally, we employ the 3D detection head and the BEV segmentation head separately or jointly for the final prediction. In particular, the 3D detection head includes two branches for classification and regression, similar to previous works like DETR3D [8], while the BEV segmentation head is comprised of an MLP network followed by a sigmoid layer.

B. S-D Encoder

In the realm of multi-camera 3D perception, it is important to note the fundamental discrepancy between the coordinate

systems of input images and output predictions. Moreover, this task necessitates not only the estimation of missing 3D layouts, but also the inference of their corresponding semantic information. Nonetheless, existing methods typically rely on indirect and limited supervision from 3D perception labels, which hinders the network’s ability to learn an optimal representation, further exacerbating the difficulty of the task. To address the limitations, we present a semantic-depth joint perception module that incorporates two auxiliary branches to effectively leverage both types of information present in 2D images, with the aim of improving the accuracy and robustness of the 3D perception task. The detailed architecture of our S-D Encoder is shown in Figure 2(b).

Considering it is highly challenging for an end-to-end neural network to generate precise depth or semantics, with only indirect and limited supervision from the 3D perception labels, we adopt a disassembled learning process, wherein each branch is explicitly supervised. This enables the accurate learning of semantic and depth features by both branches, which in turn contributes to the 3D perception performance. To acquire 2D semantic labels, we first back-project the BEV labels of the drivable area onto the image plane based on the camera parameters, and then employ the annotations from 2D detection to generate object labels. Figure 3 shows some examples of the generated 2D semantic labels. Within the context of the BEV segmentation task, since the segmentation of both road and object elements is required, road labels and object labels are concatenated to form the auxiliary semantic labels. In contrast, for the 3D object detection task, only object labels are utilized, and the segmentation of road elements is not considered. As for the depth labels, we utilize the point clouds present in the dataset to derive the ground truth.

On the other hand, to optimize the interplay between semantics and depth in both auxiliary branches, we also endeavored to achieve a balance between them through careful consideration of both architecture design and loss combination. Specifically, given the ill-posed nature of monocular depth estimation, we empirically integrated three Residual Units [19] into the depth branch, while allocating a single unit to the segmentation branch, taking into account the discrepancy in task complexity. Moreover, we empirically explore various combinations of loss weights to achieve the optimal trade-off, as detailed in Table VI.

C. Prior-guided Query Builder

Existing transformer-based approaches such as PETR [2] employ a collection of trainable anchor points in 3D space as the initial phase. Despite the fact that encoding 3D space information helps ensure convergence, the initial queries remain randomly initialized and input-independent. This introduces a considerable level of ambiguity to the model learning process and reduces detection sensitivity towards intricate scenarios. To tackle this issue, we propose the integration of data-dependent semantic priors into the initial queries, as illustrated in Figure 4(b), thereby generating input-aware queries.

We have noticed the existence of another method, namely PAP [56], which similarly integrates 2D predictions to formulate query priors. However, the strategy employed by PAP

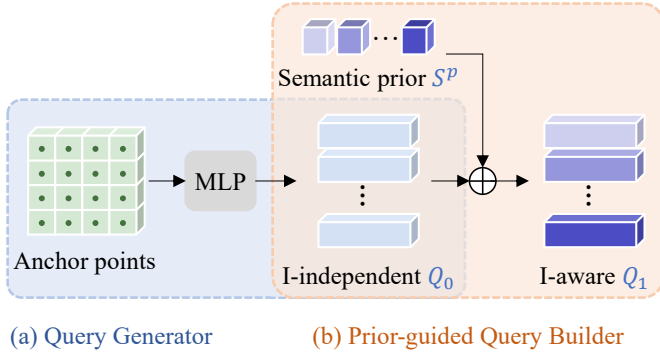


Fig. 4. **Comparison of Query Generator in PETR [2] and our Prior-guided Query Builder.** (a) In PETR, the queries Q_0 are randomly initialized and input-independent. (b) In contrast, SDTR generates input-aware queries Q_1 by encoding image-specific semantic priors, which enhances flexibility and expressiveness.

diverges significantly during the processing of these 2D priors. Initially, PAP executes the cropping of depth maps, feature maps, and semantic maps based on the predicted 2D boxes. This approach presumes a concentrated generation of queries centered around individual objects, an assumption that may prove incompatible with the requisites of BEV segmentation. This is because, the BEV segmentation task involves not only object segmentation but also the segmentation of map elements, necessitating a broader scope of query generation. Furthermore, PAP employs multiple 2D cues (such as 2D boxes, semantic maps, and depth maps) to construct the query priors, which may introduce cumulative errors due to the sparseness of depth labels in nuScenes dataset. In contrast, our PQB framework leverages only semantic maps, ensuring awareness of map elements and ensuring higher efficiency.

The architecture of our Prior-guided Query Builder is presented in Figure 2(c). Given that the essence of the transformer architecture lies in the identification of relevant features through cross-attention, it follows that for tasks such as 3D detection or segmentation, the selected feature points ought to be derived from the foreground or meaningful objects within the image. In this context, semantic segmentation can serve as a means of acquiring such effective features as prior information. Therefore, the first step of our module involves the integration of semantic priors obtained from valuable multi-view features into the query process, which can be formulated as follows:

$$S^p = \Psi(P^{seg}, S^w) \quad (1)$$

where $\Psi(\cdot)$ is a collection of operations that map the 2D segmentation map to semantic priors, S^p and S^w denote semantic priors and class-specific weights, respectively.

Specifically, given the 2D segmentation map $P^{seg} \in \mathbb{R}^{N \times C_s \times H \times W}$, where N and C_s are the number of multi-view images and semantic classes, H and W are the spatial dimensions of the feature. The group operation is utilized to aggregate all the views $\mathbb{R}^{C_s \times (N \times H \times W)}$. Subsequently, average pooling is utilized to generate the global semantic representations $\mathbb{R}^{C_s \times (N_q / C_s)}$, where N_q corresponds to the number of queries. Moreover, due to the dissimilarities in

the distribution of categories within the image, the global semantic representations are multiplied by a trainable class-specific weight $S^w \in \mathbb{R}^{C_s}$. The results are flattened to form the semantic priors $S^p \in \mathbb{R}^{N_q}$.

Finally, we incorporate S^p into the input-independent queries Q_0 to obtain the final input-dependent queries Q_1 :

$$Q_1 = S^p + Q_0 \quad (2)$$

The generated semantic priors can filter out the required global semantic representations, which provide image-specific guidance for the initial queries. In our experiments, the inclusion of semantic priors resulted in a significant enhancement of the detection performance, particularly for attribute prediction, with minimal computation overhead. Further discussions are provided in Section V-B.

D. Losses

Our network is trained by leveraging a combination of losses, comprising task-specific losses \mathcal{L}_{task} , 2D segmentation loss \mathcal{L}_{seg} , and depth estimation loss \mathcal{L}_{dep} for the auxiliary branches:

$$\mathcal{L}_{total} = \mathcal{L}_{task} + \gamma_{seg}\mathcal{L}_{seg} + \gamma_{dep}\mathcal{L}_{dep} \quad (3)$$

In our experiments, we explore the tasks of 3D detection and BEV segmentation, where \mathcal{L}_{task} is set to either \mathcal{L}_{det} or \mathcal{L}_{bev} for single-task learning, or the weighted sum of \mathcal{L}_{det} and \mathcal{L}_{bev} for multi-task learning. The hyperparameter γ is determined empirically to balance the auxiliary branches.

For \mathcal{L}_{det} in 3D object detection,

$$\mathcal{L}_{det} = \mathcal{L}_{cls} + \mathcal{L}_{reg} \quad (4)$$

where \mathcal{L}_{cls} is the focal loss for object classification. \mathcal{L}_{reg} is the L_1 loss for regression.

IV. EXPERIMENTAL SETUP

A. Datasets and Metrics

1) *Datasets*: Our proposed approach is evaluated on two extensively used large-scale autonomous driving datasets: nuScenes [3] and Lyft [60]. The nuScenes dataset comprises 1000 scenes captured in Boston and Singapore, while the Lyft dataset includes 180 scenes. Both datasets provide images captured from 6 calibrated surround-view cameras and LiDAR scans, enabling us to explicitly supervise semantic and depth estimations. Additionally, every scene provides the extrinsic and intrinsic parameters of the cameras. As the Lyft dataset does not offer a canonical train/val split, we adopt the division settings in FIERY [15] and re-implement previous methods to ensure a fair comparison.

2) *Evaluation Metrics*: For 3D object detection, we employ the standard evaluation metrics including mean Average Precision (mAP), nuScenes Detection Score (NDS), and five True Positive (TP) metrics: mean Average Translation Error (mATE), mean Average Scale Error (mASE), mean Average Orientation Error (mAOE), mean Average Velocity Error (mAVE), and mean Average Attribute Error (mAAE).

TABLE I

3D DETECTION RESULTS ON THE nuSCENES *val* SET. TO ENSURE A FAIR COMPARISON, ALL THE REPORTED MODELS WERE TRAINED WITHOUT THE INCORPORATION OF TEMPORAL INFORMATION. THE SYMBOL † INDICATES THAT THE MODEL WAS FINE-TUNED AND TESTED WITH TEST TIME AUGMENTATION.

Method	Backbone	Resolution	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
BEVDepth [49]	R50	512×1408	0.359	0.312	0.718	0.278	0.638	1.150	0.334
PETR [2]	R50	512×1408	0.367	0.317	0.840	0.280	0.616	0.954	0.233
SDTR (Ours)	R50	512×1408	0.384	0.331	0.799	0.280	0.616	0.904	0.212
FCOS3D† [57]	R101	900×1600	0.415	0.343	0.725	0.263	0.422	1.292	0.153
DETR3D [8]	R101	900×1600	0.425	0.346	0.773	0.268	0.383	0.842	0.216
PGD† [58]	R101	900×1600	0.428	0.369	0.683	0.260	0.439	1.268	0.185
BEVFormer-S [17]	R101	900×1600	0.448	0.375	0.725	0.272	0.391	0.802	0.200
SimMOD [55]	R101	900×1600	0.455	0.366	0.698	0.264	0.340	0.784	0.197
Ego3RT [59]	R101	900×1500	0.450	0.375	0.657	0.268	0.391	0.850	0.206
BEVDepth [49]	R101	512×1408	0.408	0.376	0.659	0.267	0.543	1.059	0.335
PETR [2]	R101	512×1408	0.441	0.366	0.717	0.267	0.412	0.834	0.190
SDTR (Ours)	R101	512×1408	0.462	0.380	0.657	0.267	0.386	0.806	0.167
M ² BEV [54]	X101	900×1600	0.470	0.417	0.647	0.275	0.377	0.834	0.245
DETR3D [8]	V2-99	900×1600	0.374	0.303	0.860	0.278	0.437	0.967	0.235
BEVDet [7]	Swin-T	512×1408	0.417	0.349	0.637	0.269	0.490	0.914	0.268
PETR [2]	Swin-T	512×1408	0.431	0.361	0.732	0.273	0.497	0.808	0.185
SDTR (Ours)	V2-99	512×1408	0.482	0.430	0.643	0.265	0.406	0.830	0.192

The NDS metric is a weighted sum of mAP and five TP metrics. Additionally, in the realm of Bird’s Eye View (BEV) segmentation, the widely used Intersection over Union (IoU) metric is adopted as the primary evaluation metric.

B. Implementation Details

Our implementation is based on the PyTorch framework [61], and our work is focused on two primary 3D perception tasks: 3D object detection and BEV segmentation. Specifically, we employ the ResNet series [19] and VoVNetV2 [62] as the backbone networks, generating the output features $F^{\{4,5\}}$ with 1/16 input resolution. The transformer decoder with 6 layers is adopted to constantly update the queries.

In our experiments, we resize and crop the multi-view images to 512×1408 for use as network inputs. To define the perception ranges, we set the X and Y axes to [-61.2m, 61.2m], and the Z axis to [-10m, 10m]. Our network is trained using the AdamW optimizer [63] with a weight decay of $1e-2$. The learning rate is initialized to $2e-4$ and decayed using the cosine annealing policy [64]. It is worth noting that, unlike previous works which are typically trained on Tesla A100 or V100 GPUs, we conduct all of our experiments using 8×2080Ti GPUs. The training process runs for 24 epochs with a batch size of 8.

In our 3D object detection task, we leverage 900 detection queries and employ the Focal Loss [65] for object classification, along with the L1 loss for 3D bounding box regression. For BEV segmentation, we conduct experiments using 625 BEV segmentation queries and utilize the weighted cross-entropy loss for supervision on the predicted BEV map. Additionally, for both tasks, we employ the binary cross-entropy loss as an auxiliary loss for both depth estimation and 2D segmentation.

V. EXPERIMENTAL RESULTS

A. State-of-the-art Comparisons

When comparing the proposed SDTR with other state-of-the-art (SOTA) methods, it is observed that the majority of these SOTA approaches rely heavily on full—or even supra-maximal—resolution to achieve superior results. Interestingly, the performance enhancements often derive more from the use of high-definition images rather than intrinsic improvements in the methodology itself. Furthermore, prior studies typically utilize a large batch size (e.g., 32) [7], [49], [57]–[59] to ensure the convergence of the model. This requirement inevitably leads to these models being trained solely on high-capacity GPUs such as A100/V100, significantly constraining the practical applicability of the models. In view of these observations, the experiments in this study have been primarily conducted at a lower resolution (specifically, 512×1408) and with a reduced batch size of 8. These adjustments aim to establish a more equitable basis for comparison and enhance the generalizability of the model.

1) *3D Object Detection*: We evaluate the performance of our proposed method and other state-of-the-art approaches on the *val* and *test* splits of the nuScenes dataset, as presented in Table I and Table II, respectively. On the *val* set, the proposed SDTR outperforms existing paradigms in terms of both NDS and mAP metrics across various existing paradigms, and notably, it is achieved with a smaller input resolution. Notably, our model achieves superior performance in accurately reasoning about the 3D scene geometry and semantic category by utilizing both semantic and depth priors, surpassing existing depth-based (e.g., BEVDepth and M²BEV) and transformer-based (e.g., PETR) approaches. On the *test* set, SDTR also surpasses the previous best method with higher scores of

TABLE II

3D DETECTION RESULTS ON THE nuSCENES *test* SET. TO ENSURE A FAIR COMPARISON, ALL THE REPORTED MODELS WERE TRAINED WITHOUT THE INCORPORATION OF TEMPORAL INFORMATION. THE SYMBOL † DENOTES THAT THE METHOD USES TEST TIME AUGMENTATION. THE BEVDet, DETR3D, PETR, BEVDEPTH, BEVFORMER, AND SDTR ARE ALL TRAINED WITH CBGS [66].

Method	Backbone	Resolution	NDS↑	mAP↑	mATE↓	mASE↓	mAOE↓	mAVE↓	mAAE↓
CenterNet [21]	DLA	-	0.400	0.338	0.658	0.255	0.629	1.629	0.142
Ego3RT [59]	R101	900×1500	0.443	0.389	0.599	0.268	0.470	1.169	0.172
FCOS3D† [57]	R101	900×1600	0.428	0.358	0.690	0.249	0.452	1.434	0.124
PGD† [58]	R101	900×1600	0.448	0.386	0.626	0.245	0.451	1.509	0.127
PETR [2]	R101	900×1600	0.455	0.391	0.647	0.251	0.433	0.933	0.143
SimMOD [55]	R101	900×1600	0.464	0.382	0.623	0.252	0.394	0.863	0.132
Graph-DETR3D [67]	R101	900×1600	0.472	0.418	0.668	0.250	0.440	0.876	0.139
M ² BEV [54]	X101	900×1600	0.474	0.429	0.583	0.254	0.376	1.053	0.190
BEVDet [7]	Swin-S	768×2112	0.463	0.398	0.556	0.239	0.414	1.010	0.153
PETR [2]	Swin-S	768×2112	0.481	0.434	0.641	0.248	0.437	0.894	0.143
DD3D† [68]	V2-99	-	0.477	0.418	0.572	0.249	0.368	1.014	0.124
Ego3RT [59]	V2-99	900×1500	0.473	0.425	0.549	0.264	0.433	1.014	0.145
DETR3D [8]	V2-99	900×1600	0.479	0.412	0.641	0.255	0.394	0.845	0.133
BEVDet [7]	V2-99	900×1600	0.488	0.424	0.524	0.242	0.373	0.950	0.148
SimMOD [55]	V2-99	900×1600	0.494	0.417	0.570	0.248	0.387	0.813	0.126
Graph-DETR3D [67]	V2-99	900×1600	0.495	0.425	0.621	0.251	0.386	0.790	0.128
BEVFormer-S [17]	V2-99	900×1600	0.495	0.435	0.589	0.254	0.402	0.842	0.131
PETR [2]	V2-99	900×1600	0.504	0.441	0.593	0.249	0.383	0.808	0.132
PETR [2]	V2-99	512×1408	0.495	0.437	0.601	0.248	0.405	0.841	0.142
SDTR (Ours)	V2-99	512×1408	0.505	0.449	0.579	0.250	0.392	0.833	0.140

TABLE III

BEV SEGMENTATION RESULTS ON THE nuSCENES *val* SET. PLEASE NOTE THAT THE TOP SECTION EMPLOYED DIFFERENT BEV GRID SETTINGS OR VALIDATION SPLITS, WHILE THE MIDDLE SECTION FOCUSED EXCLUSIVELY ON SINGLE-CLASS SEGMENTATION, AS OPPOSED TO THE MULTI-CLASS SEGMENTATION STUDIES WE UTILIZED. BOTH SECTIONS ARE INCLUDED SOLELY FOR REFERENCE PURPOSES. TO ENSURE FAIRNESS, WE FURTHER CATEGORIZED THE REMAINING METHODS INTO TWO GROUPS BASED ON THE APPLICATION OF AUXILIARY SUPERVISION. * DENOTES THAT TEMPORAL INFORMATION IS UTILIZED. † REPRESENTS THAT GRAPH NEURAL NETWORK IS EMPLOYED.

Method	IoU-Drive↑	IoU-Lane↑	IoU-Veh.↑
VED [35]	0.547	-	0.088
VPN [36]	0.580	-	0.255
PON [37]	0.604	-	0.247
LSF [50]	0.611	-	0.378
FISHING [69]	-	-	0.300
STA* [13]	0.707	-	0.360
Image2Map* [70]	0.745	-	0.397
Ego3RT [59]	0.796	0.475	-
OFT [71]	0.717	0.181	0.301
Lift-Splat [1]	0.729	0.199	0.321
FIERY-S [15]	-	-	0.358
FIERY* [15]	-	-	0.382
PedLam† [39]	0.814	-	0.498
BEVFormer-S [17]	0.807	0.213	0.432
BEVFormer* [17]	0.801	0.257	0.448
M ² BEV [54]	0.759	0.380	-
SDTR (Ours)	0.841	0.476	0.450

50.5% NDS and 44.9% mAP, while utilizing *only 1/2 of the input size* compared to its competitors.

TABLE IV

BEV SEGMENTATION RESULTS ON THE LYFT DATASET. DUE TO THE FACT THAT PREVIOUS STUDIES EMPLOYED DIFFERENT VALIDATION SPLITS FOR THE LYFT DATASET, WE HAVE RE-IMPLEMENTED THESE METHODS IN THE INTEREST OF ENSURING A FAIR COMPARISON.

Method	IoU-Car↑	IoU-Vehicle↑
Lift-Splat [1]	0.389	0.382
FIERY-S [15]	-	0.410
SDTR (Ours)	0.457	0.451

2) *BEV Segmentation:* We further provide a comparative analysis of our proposed method against previous state-of-the-art BEV segmentation approaches on both the nuScenes dataset and the Lyft dataset. The results are presented in Table III and Table IV, respectively. It is worth noting that our approach consistently outperforms all existing methods and sets a new state-of-the-art performance across all categories. While one prior work, BEVFormer [17], achieves a high IoU score on the nuScenes dataset by leveraging temporal information and taking full-resolution images as input, our method still achieves better results even in the absence of temporal clues and using smaller input size of 512x1408.

3) *Visualization Results:* Figure 1 showcases a visual comparison between our proposed SDTR model and two classical methods in both 3D object detection and BEV segmentation. Owing to the semantic-depth priors and the input-dependent queries, our SDTR model exhibits strong 3D detection performance while ensuring consistent segmentation. Additionally, Figure 5 and Figure 6 provide further qualitative results for 3D object detection and BEV segmentation respectively. These

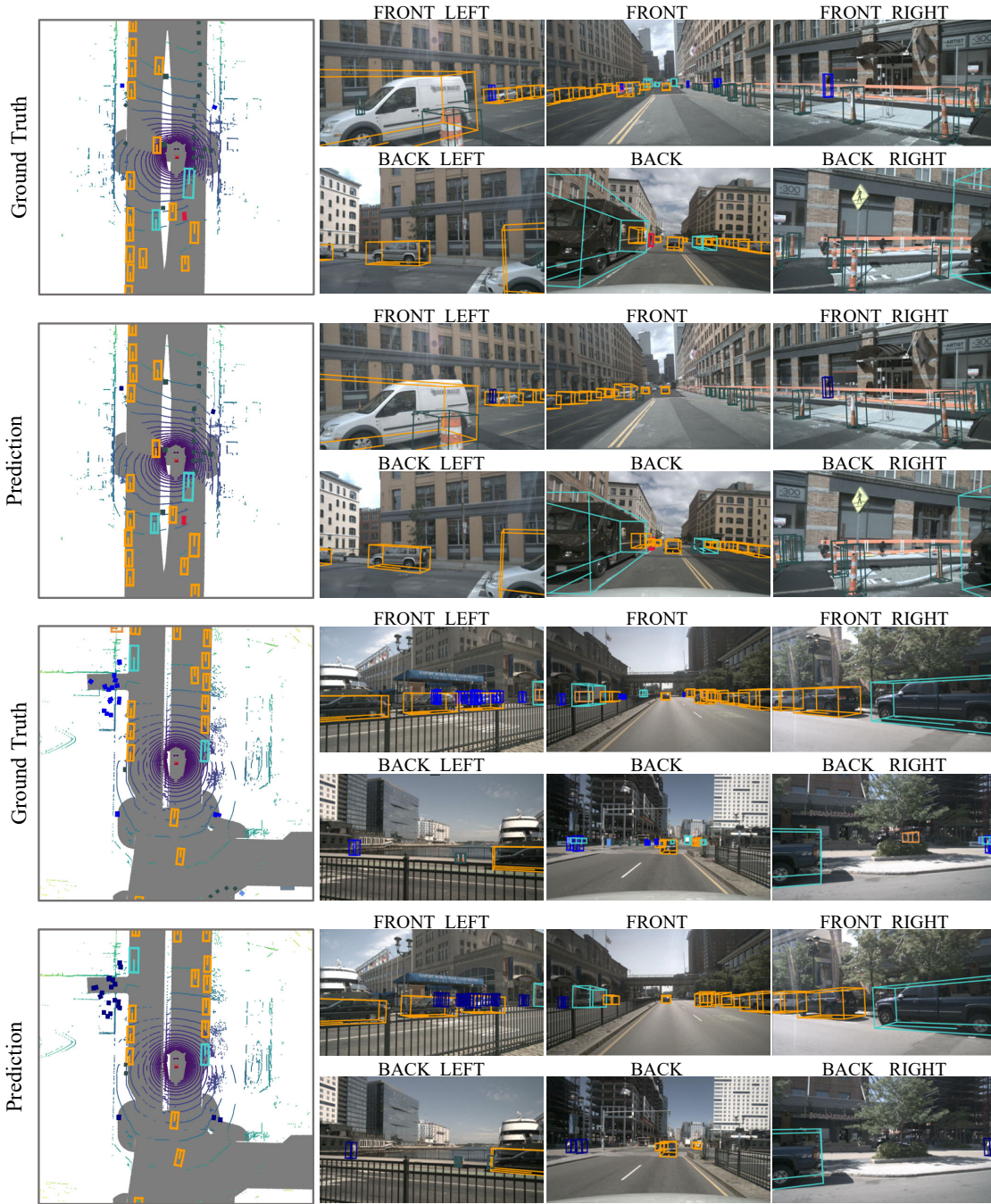


Fig. 5. **Visualization results for 3D object detection.** We show the 3D bounding box predictions in multi-camera images and the bird’s-eye-view. The 3D bounding boxes are drawn with different colors to distinguish different classes.

visualizations underscore the proficiency of our SDTR model in executing 3D object detection tasks across diverse scales and distances, as well as BEV segmentation tasks—even under challenging conditions when objects present irregular or extreme shapes. Despite these promising results, there are instances where our methodology encounters difficulties. Notably, our model may struggle in scenarios where vehicles are densely clustered, as illustrated by the red circles in Figure 6. Such failure cases are areas of focus for future improvements in our model’s performance.

Furthermore, as depicted in Figure 7, the depth branch

integrated within our S-D Encoder demonstrates the capability to accurately estimate depth values, offering valuable 3D positional priors for 2D-to-3D reasoning. However, as indicated by the red circles, there is a tendency for inaccurate depth detection when dealing with distant objects, primarily due to the sparse nature of depth supervision. By incorporating dense semantic priors into our initial queries, we can greatly enhance the recognition of distant classes, and this improvement will be thoroughly demonstrated in Table IX.

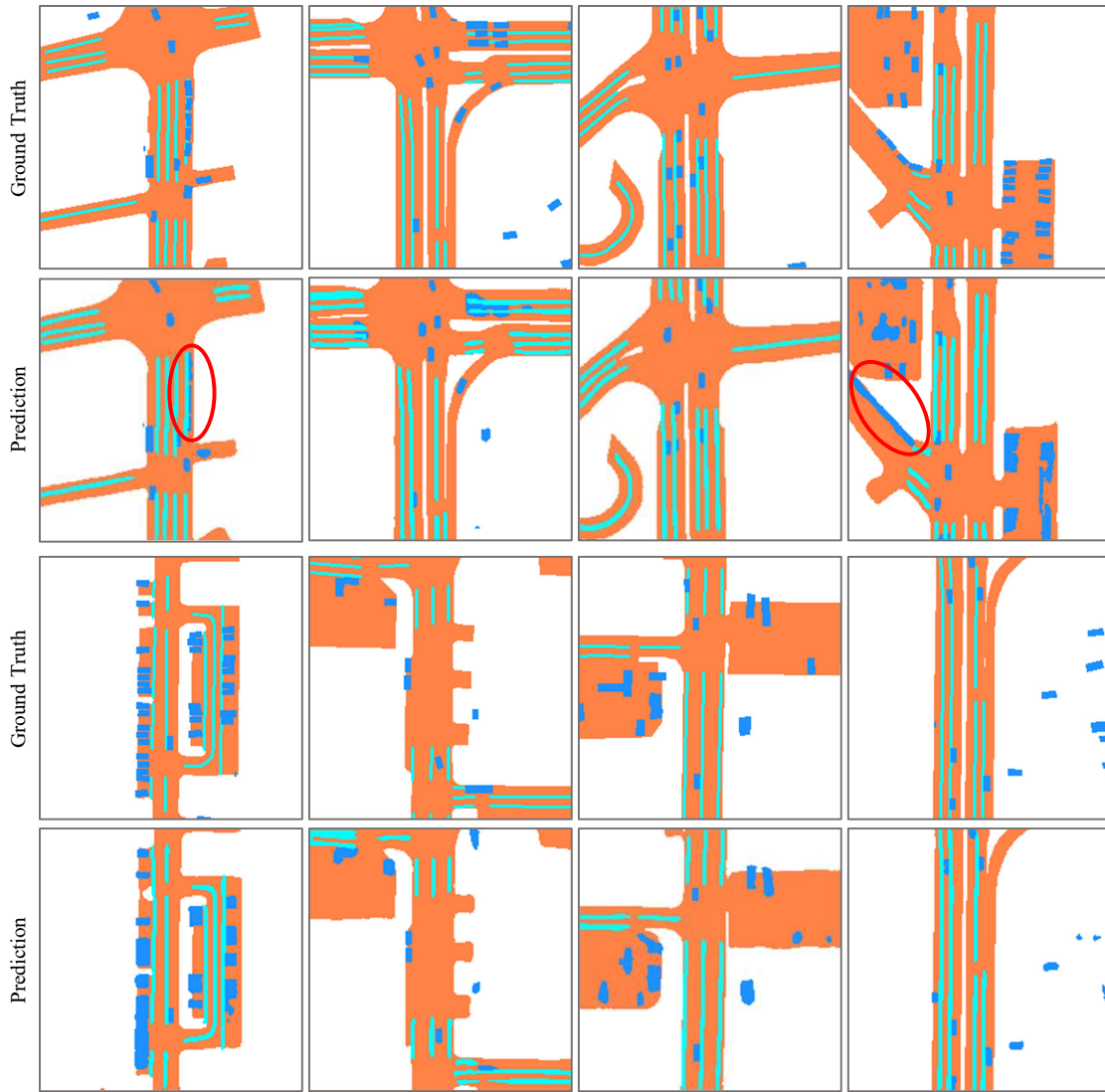


Fig. 6. **Visualization results for BEV segmentation.** Classes of vehicle, drivable area, and lane segmentation are filled with blue, orange, and cyan, respectively.

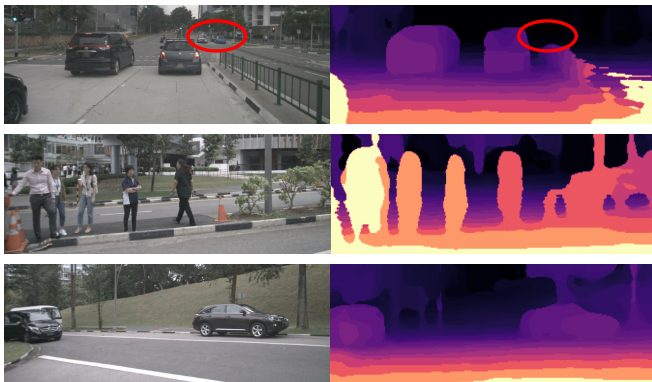


Fig. 7. **Visualization of the depth predictions.**

B. Ablation Studies

To further verify the effectiveness of the proposed modules, we conduct ablation studies from six aspects. All experiments

are conducted on the nuScenes *val* set and R50 is utilized as the backbone network if not specified.

1) *Effectiveness of Proposed Modules:* To evaluate the effectiveness of each component in our proposed method, we began with a baseline network and incrementally added the proposed modules. Table V summarizes our results. First, it can be seen that the baseline network yielded a NDS and mAP score of 36.0% and 31.2%, respectively. Adding the 2D segmentation (2D Seg) branch led to an improvement in NDS and mAP (0.7% and 0.9%, respectively), with a negligible increase in computational cost. Our results also indicate that the 2D Seg branch reduced classification and position estimation errors (mAAE, mATE, and mAOE), emphasizing the role of semantic clues as priors for both tasks. Further, combining the 2D Seg and depth estimation (DE) branches led to a more substantial improvement in NDS and mAP (1.7% and 1.5%, respectively), demonstrating the effectiveness of decoupling 2D segmentation and depth estimation from

TABLE V

ABLATION STUDIES OF PROPOSED MODULES. “2D SEG” AND “DE” REPRESENT THE 2D SEGMENTATION AND DEPTH ESTIMATION BRANCHES IN THE S-D ENCODER, RESPECTIVELY. WE MEASURE THE INCREASED COMPUTATION COMPLEXITY (MEASURED BY THE NUMBER OF FLOPS) AND GPU MEMORY USAGE INTRODUCED BY THE PROPOSED MODULES WITHOUT COUNTING THE COMPLEXITY FROM THE BASELINE. THE FPS (FRAMES PER SECOND) IS MEASURED ON A SINGLE 2080TI GPU.

2D Seg	DE	PQB	NDS↑	mAP↑	mATE↓	mAOE↓	mAAE↓	FLOPs	Memory	FPS
			0.360	0.312	0.834	0.659	0.237	-	-	5.1
✓			0.367	0.321	0.816	0.644	0.230	30.7G	38M	5.0
✓		✓	0.373	0.326	0.810	0.628	0.221	30.7G	38M	5.0
✓	✓		0.377	0.327	0.807	0.617	0.237	138.2G	648M	4.6
✓	✓	✓	0.384	0.331	0.799	0.616	0.212	138.2G	648M	4.6

TABLE VI

ABLATION STUDIES OF LOSS COMBINATIONS. WE ADOPT DIFFERENT LOSS WEIGHT COMBINATIONS TO DIRECTLY REFLECT THE EFFECTS OF AUXILIARY SUPERVISION.

γ_{seg}	γ_{dep}	NDS↑	mAP↑
1.0	1.0	0.378	0.329
2.0	1.0	0.380	0.331
3.0	1.0	0.384	0.331
4.0	1.0	0.379	0.328

TABLE VII

INTERSECTION OVER UNION SCORES OF OBJECT CATEGORIES. V2-99 IS UTILIZED AS THE BACKBONE NETWORK.

	Car	Truck	Construction vehicle	Bus	Trailer	Barrier	Motorcycle	Bicycle	Pedestrian	Traffic cone	mIoU
SDTR	0.455	0.360	0.132	0.453	0.332	0.330	0.223	0.210	0.208	0.222	0.292

TABLE VIII

ANALYSIS ON MULTI-TASK JOINT LEARNING. WE APPLY MULTIPLE LOSS WEIGHT COMBINATIONS TO EXPLORE THE EFFECT OF JOINT LEARNING, WHERE V2-99 IS UTILIZED AS THE BACKBONE NETWORK.

Task Head		3D Detection		BEV Segmentation		
Det	BEV	NDS↑	mAP↑	Drive↑	Lane↑	Vehicle↑
1.0	0.0	0.465	0.417	-	-	-
0.0	1.0	-	-	0.827	0.427	0.421
1.0	1.0	0.462	0.412	0.815	0.415	0.413
1.0	2.0	0.453	0.399	0.820	0.424	0.423

TABLE IX

ABLATION STUDIES ON SMALL DISTANT CATEGORIES. “TRANS.”, “VEL.”, AND “ATTR.” REPRESENT THE TRANSLATION, VELOCITY, AND ATTRIBUTE ERRORS IN 3D DETECTION, RESPECTIVELY. WE USE THE NOTATION “w/o” TO DENOTE THE MODEL WITHOUT THE PQB MODULE, AND “w” TO DENOTE THE MODEL WITH THE PQB MODULE.

Class	Trans.↓		Vel.↓		Attr.↓	
	w/o	w	w/o	w	w/o	w
Motor	0.734	0.726	1.615	1.517	0.148	0.126
Bike	0.725	0.714	0.484	0.427	0.028	0.013
Ped.	0.727	0.722	0.799	0.781	0.352	0.298

2D-to-3D transformation. Given the inherently ill-posed challenge of monocular depth estimation, we empirically utilize a larger quantity of Residual Units in the depth branch. As a consequence, the depth estimation branch necessitated higher computational resources compared to the 2D segmentation branch. Notably, the integration of PQB allowed our SDTR to achieve superior overall performance, achieving a NDS of 38.4% and a mAP of 33.1%. Remarkably, this performance enhancement was achieved without necessitating any additional computational or memory expenditures. The reason for constant computational complexity in our PQB is primarily due to the fact that the only trainable parameter it utilizes is a class-specific weight $S^w \in \mathbb{R}^{C_s}$, where $C_s = 10$ or 3 in 3D detection and BEV segmentation respectively. And such a trainable parameter is significantly smaller in size compared to even the simplest 1x1 convolution layer. Consequently, the increased computational complexity introduced by the PQB is too trivial to be measured or quantified. These findings suggest that PQB can provide effective guidance for query updating, thereby mutually enhancing 3D perception in a resource-efficient manner.

Moreover, to further illustrate the practicality of our SDTR model in terms of inference speed, we have included FPS results in Table V. As observed, the computational burden for inference is marginally increased, due to the substantial DE branch only being present during the training stage.

2) *Combination of Losses:* In practice, it has been observed that the depth loss typically exhibits significantly higher numerical values compared to those of the semantics loss. To determine the optimal ratio between their respective combination weights and achieve a balance between these two losses, we conducted a series of experiments, as presented in Table VI. Based on empirical results, we find that setting the ratio of semantics loss to depth loss at 3 results in the highest accuracy. Further increasing the weight of semantics loss beyond this ratio does not yield a corresponding improvement in performance.

3) *Results of Per-object Categories:* We present Intersection over Union (IoU) metrics for all object categories in Table VII, which serve to validate the effectiveness of our approach in segmenting objects, particularly small and distant objects. It’s worth noting that a direct comparison with existing literature on per-object segmentation encounters methodology constraints, primarily because they adopt a train/validation split divergent from ours. In spite of this, our findings still demonstrates that the proposed SDTR maintains robust performance, notably in localizing small, distant categories.

4) *Analysis on Multi-task Learning:* We also investigate the impact of multi-task learning within the proposed framework by varying the loss weight assigned to each task. As presented

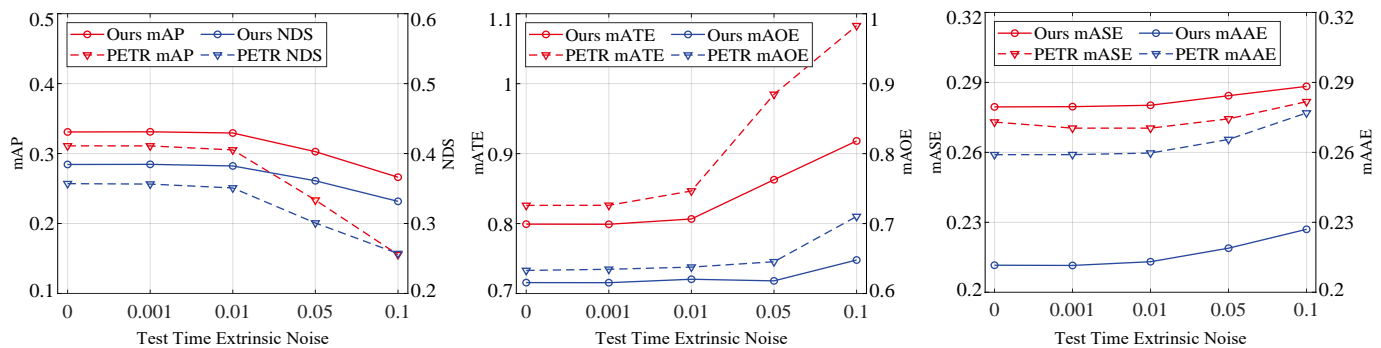


Fig. 8. **The detection results on the nuScenes val set with different extrinsic noises.** In particular, random Gaussian noise is added to the camera extrinsic during testing. The findings suggest that our SDTR model exhibits superior robustness under large extrinsic noise.

in Table VIII, our single-task model demonstrates higher accuracy across the majority of tasks, with the exception of vehicle segmentation. It is widely acknowledged that joint learning often yields a slightly lower precision than single-task learning [72], [73]. However, in the context of our study, it is possible that the improved precision in vehicle segmentation achieved through joint learning can be attributed to the 3D position information of vehicles acquired from 3D detection. Furthermore, our findings suggest that assigning a higher loss weight to the BEV segmentation task than the detection task in joint learning can lead to a performance increase of approximately 0.5% to 1.2%.

5) *Analysis on the Prior-guided Query Builder:* To assess the effectiveness of our proposed PQB in detecting small distant objects, we present the results of our experiments on translation, velocity, and attribute errors across three distant classes using different model configurations: without PQB and with PQB. The findings, as illustrated in Table IX, confirm the substantial improvement our PQB module contributes to the detection and accurate localization of these typically challenging distant objects. By visibly reducing false negatives, the PQB has proven instrumental in enhancing the overall precision of our model, particularly in scenarios traditionally prone to detection errors.

6) *Model Robustness:* The reliability of autonomous vehicles significantly hinges on the robustness of sensors. To evaluate the robustness of our model against potential sensor damage, we introduce two common types of sensor error during the testing phase. Figure 8 illustrates a comparison of our SDTR model and the PETR model under various degrees of extrinsic noise in the matrix. The findings suggest that our SDTR model demonstrates superior robustness against extrinsic noise as compared to the PETR model, as observed across all performance metrics. Differing from PETR, which heavily relies on 3D coordinates computed by camera parameters, our framework leverages depth predictions to perceive precise 3D positions. In addition, the input-dependent queries in our model facilitate a comprehensive global perception that remains largely unaffected by camera parameters. Furthermore, we conduct a test where we randomly remove several camera images from each sample to evaluate the robustness of our model to camera dropout. As shown in Figure 9, our SDTR

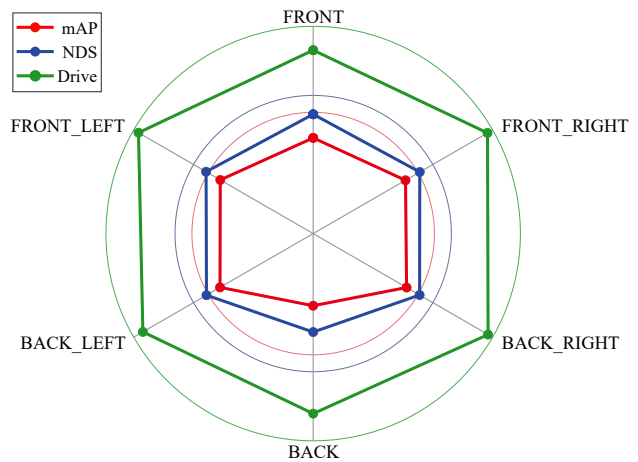


Fig. 9. **The performance on the nuScenes val set with camera drops.** For each metric, the points and the circle denote the results w/ and w/o camera drop. The closer to the center, the greater the degradation.

model exhibits a relatively high level of accuracy even in the absence of training on such sensor errors. However, it is worth noting that the performance degradation caused by the loss of the BACK camera, which has a wider field of view, is more significant.

VI. CONCLUSION

In this paper, we propose a unified framework, named SDTR, for addressing the challenges associated with multi-camera 3D object detection and BEV segmentation. In contrast to conventional techniques that rely on a tightly coupled learning process to extract categorical and 3D positional information, our approach first adopts S-D Encoders to explicitly extract semantic and depth priors. This decoupling enables greater flexibility and efficiency in our approach. Furthermore, we propose a Prior-guided Query Builder that transforms input-independent queries into input-aware ones. Experiments on the nuScenes and Lyft benchmarks demonstrate that our SDTR significantly improves the recovery of semantic information and 3D positions of objects, leading to better performance in both tasks.

REFERENCES

- [1] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2020, pp. 194–210.
- [2] Y. Liu, T. Wang, X. Zhang, and J. Sun, "Petr: Position embedding transformation for multi-view 3d object detection," *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 531–548, 2022.
- [3] H. Caesar, V. Bankiti, A. H. Lang, S. Vora, V. E. Liong, Q. Xu, A. Krishnan, Y. Pan, G. Baldan, and O. Beijbom, "nuscenes: A multi-modal dataset for autonomous driving," *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11 621–11 631, 2020.
- [4] G. N. DeSouza and A. C. Kak, "Vision for mobile robot navigation: A survey," *IEEE transactions on pattern analysis and machine intelligence*, vol. 24, no. 2, pp. 237–267, 2002.
- [5] J. U. Kim, H.-I. Kim, and Y. M. Ro, "Stereoscopic vision recalling memory for monocular 3d object detection," *IEEE Transactions on Image Processing*, 2023.
- [6] L. Xie, G. Xu, D. Cai, and X. He, "X-view: non-egocentric multi-view 3d object detector," *IEEE Transactions on Image Processing*, vol. 32, pp. 1488–1497, 2023.
- [7] J. Huang, G. Huang, Z. Zhu, and D. Du, "Bevdet: High-performance multi-camera 3d object detection in bird-eye-view," *arXiv preprint arXiv:2112.11790*, 2021.
- [8] Y. Wang, V. C. Guizilini, T. Zhang, Y. Wang, H. Zhao, and J. Solomon, "Detr3d: 3d object detection from multi-view images via 3d-to-2d queries," in *Conference on Robot Learning*. PMLR, 2022, pp. 180–191.
- [9] S. Schuster, M. Zhai, N. Jacobs, and M. Chandraker, "Learning to look around objects for top-view representations of outdoor scenes," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 787–802.
- [10] K. Mani, S. Daga, S. Garg, S. S. Narasimhan, M. Krishna, and K. M. Jatavallabhula, "Monolayout: Amodal scene layout from a single image," *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 1689–1697, 2020.
- [11] B. Liu, B. Zhuang, and M. Chandraker, "Weakly but deeply supervised occlusion-reasoned parametric road layouts," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17 000–17 009, 2022.
- [12] B. Liu, B. Zhuang, S. Schuster, P. Ji, and M. Chandraker, "Understanding road layout from videos as a whole," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4414–4423, 2020.
- [13] A. Saha, O. Mendez, C. Russell, and R. Bowden, "Enabling spatio-temporal aggregation in birds-eye-view vehicle estimation," in *IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 5133–5139.
- [14] C. Reading, A. Harakeh, J. Chae, and S. L. Waslander, "Categorical depth distribution network for monocular 3d object detection," 2021, pp. 8555–8564.
- [15] A. Hu, Z. Murez, N. Mohan, S. Dudas, J. Hawke, V. Badrinarayanan, R. Cipolla, and A. Kendall, "Fiery: Future instance prediction in bird's-eye view from surround monocular cameras," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 15 273–15 282.
- [16] Y. Zhang, Z. Zhu, W. Zheng, J. Huang, G. Huang, J. Zhou, and J. Lu, "Beverse: Unified perception and prediction in birds-eye-view for vision-centric autonomous driving," *arXiv preprint arXiv:2205.09743*, 2022.
- [17] Z. Li, W. Wang, H. Li, E. Xie, C. Sima, T. Lu, Y. Qiao, and J. Dai, "Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2022, pp. 1–18.
- [18] L. Peng, Z. Chen, Z. Fu, P. Liang, and E. Cheng, "Bevsegformer: Bird's eye view semantic segmentation from arbitrary camera rigs," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023, pp. 5935–5943.
- [19] K. He, X. Zhang, S. Ren, and J. Sun, "Identity mappings in deep residual networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 630–645.
- [20] F. Yu, V. Koltun, and T. Funkhouser, "Dilated residual networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 472–480.
- [21] X. Zhou, D. Wang, and P. Krähenbühl, "Objects as points," *arXiv preprint arXiv:1904.07850*, 2019.
- [22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems (NIPS)*, vol. 30, 2017.
- [23] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.
- [24] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 213–229, 2020.
- [25] C. Yang, Y. Chen, H. Tian, C. Tao, X. Zhu, Z. Zhang, G. Huang, H. Li, Y. Qiao, L. Lu *et al.*, "Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision," *arXiv preprint arXiv:2211.10439*, 2022.
- [26] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia, "Pyramid scene parsing network," *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2881–2890, 2017.
- [27] J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 3431–3440.
- [28] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.
- [29] X. He, J. Liu, W. Wang, and H. Lu, "An efficient sampling-based attention network for semantic segmentation," *IEEE Transactions on Image Processing*, vol. 31, pp. 2850–2863, 2022.
- [30] H. Tian, S. Qu, and P. Payeur, "A prototypical knowledge oriented adaptation framework for semantic segmentation," *IEEE Transactions on Image Processing*, vol. 31, pp. 149–163, 2021.
- [31] R. Ke, A. I. Aviles-Rivero, S. Pandey, S. Reddy, and C.-B. Schönlieb, "A three-stage self-training framework for semi-supervised semantic segmentation," *IEEE Transactions on Image Processing*, vol. 31, pp. 1805–1815, 2022.
- [32] C.-C. Lin and M.-S. Wang, "A vision based top-view transformation model for a vehicle parking assistant," *Sensors (Basel, Switzerland)*, vol. 12, pp. 4431 – 4446, 2012.
- [33] S. Ammar Abbas and A. Zisserman, "A geometric approach to obtain a bird's eye view from an image," *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pp. 0–0, 2019.
- [34] T. Sämann, K. Amende, S. Milz, C. Witt, M. Simon, and J. Petzold, "Efficient semantic segmentation for visual bird's-eye view interpretation," in *International Conference on Intelligent Autonomous Systems*. Springer, 2018, pp. 679–688.
- [35] C. Lu, M. van de Molengraft, and G. Dubbelman, "Monocular semantic occupancy grid mapping with convolutional variational encoder–decoder networks," *IEEE Robotics and Automation Letters (RA-L)*, vol. 4, pp. 445–452, 2019.
- [36] B. Pan, J. Sun, H. Y. T. Leung, A. Andonian, and B. Zhou, "Cross-view semantic segmentation for sensing surroundings," *IEEE Robotics and Automation Letters (RA-L)*, vol. 5, pp. 4867–4873, 2020.
- [37] T. Roddick and R. Cipolla, "Predicting semantic map representations from images using pyramid occupancy networks," *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11 138–11 147, 2020.
- [38] N. Gosala and A. Valada, "Bird's-eye-view panoptic segmentation using monocular frontal view images," *IEEE Robotics and Automation Letters (RA-L)*, vol. 7, no. 2, pp. 1968–1975, 2022.
- [39] A. Saha, O. Mendez, C. Russell, and R. Bowden, "the pedestrian next to the lamppost" adaptive object graphs for better instantaneous mapping," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 19 528–19 537.
- [40] B. Zhou and P. Krähenbühl, "Cross-view transformers for real-time map-view semantic segmentation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022, pp. 13 760–13 769.
- [41] W. Yang, Q. Li, W. Liu, Y. Yu, Y. Ma, S. He, and J. Pan, "Projecting your view attentively: Monocular road scene layout estimation via cross-view transformation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 15 536–15 545.
- [42] D. Eigen and R. Fergus, "Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture," in

- Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2015, pp. 2650–2658.
- [43] H. Fu, M. Gong, C. Wang, K. Batmanghelich, and D. Tao, “Deep ordinal regression network for monocular depth estimation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018, pp. 2002–2011.
- [44] C. Godard, O. Mac Aodha, and G. J. Brostow, “Unsupervised monocular depth estimation with left-right consistency,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 270–279.
- [45] C. Godard, O. Mac Aodha, M. Firman, and G. J. Brostow, “Digging into self-supervised monocular depth estimation,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019, pp. 3828–3838.
- [46] S. F. Bhat, I. Alhashim, and P. Wonka, “Adabins: Depth estimation using adaptive bins,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 4009–4018.
- [47] Y. Li, H. Bao, Z. Ge, J. Yang, J. Sun, and Z. Li, “Bevstereo: Enhancing depth estimation in multi-view 3d object detection with dynamic temporal stereo,” *arXiv preprint arXiv:2209.10248*, 2022.
- [48] J. Park, C. Xu, S. Yang, K. Keutzer, K. Kitani, M. Tomizuka, and W. Zhan, “Time will tell: New outlooks and a baseline for temporal multi-view 3d object detection,” *arXiv preprint arXiv:2210.02443*, 2022.
- [49] Y. Li, Z. Ge, G. Yu, J. Yang, Z. Wang, Y. Shi, J. Sun, and Z. Li, “Bevdepth: Acquisition of reliable depth for multi-view 3d object detection,” *arXiv preprint arXiv:2206.10092*, 2022.
- [50] I. Dwivedi, S. Malla, Y.-T. Chen, and B. Dariush, “Bird’s eye view segmentation using lifted 2d semantic features,” in *British Machine Vision Conference (BMVC)*, 2021, pp. 6985–6994.
- [51] H. Zhang, H. Li, X. Liao, F. Li, S. Liu, L. M. Ni, and L. Zhang, “Da-bev: Depth aware bev transformer for 3d object detection,” *arXiv preprint arXiv:2302.13002*, 2023.
- [52] Z. Chen, Z. Li, S. Zhang, L. Fang, Q. Jiang, F. Zhao, B. Zhou, and H. Zhao, “Autoalign: Pixel-instance feature aggregation for multi-modal 3d object detection,” *International Joint Conferences on Artificial Intelligence (IJCAI)*, 2022.
- [53] V. A. Sindagi, Y. Zhou, and O. Tuzel, “Mvx-net: Multimodal voxelnet for 3d object detection,” in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 7276–7282.
- [54] E. Xie, Z. Yu, D. Zhou, J. Philion, A. Anandkumar, S. Fidler, P. Luo, and J. M. Alvarez, “M² bev: Multi-camera joint 3d detection and segmentation with unified birds-eye view representation,” *arXiv preprint arXiv:2204.05088*, 2022.
- [55] Y. Zhang, W. Zheng, Z. Zhu, G. Huang, J. Zhou, and J. Lu, “A simple baseline for multi-camera 3d object detection,” *arXiv preprint arXiv:2208.10035*, 2022.
- [56] D. Feng and F. Ferroni, “Priors are powerful: Improving a transformer for multi-camera 3d detection with 2d priors,” *arXiv preprint arXiv:2301.13592*, 2023.
- [57] T. Wang, X. Zhu, J. Pang, and D. Lin, “Fcos3d: Fully convolutional one-stage monocular 3d object detection,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 913–922.
- [58] T. Wang, Z. Xinge, J. Pang, and D. Lin, “Probabilistic and geometric depth: Detecting objects in perspective,” in *Conference on Robot Learning*. PMLR, 2022, pp. 1475–1485.
- [59] J. Lu, Z. Zhou, X. Zhu, H. Xu, and L. Zhang, “Learning ego 3d representation as ray tracing,” in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2022, pp. 129–144.
- [60] R. Kesten, M. Usman, J. Houston, T. Pandya, K. Nadhamuni, A. Ferreira, M. Yuan, B. Low, A. Jain, P. Ondruska *et al.*, “Lyft level 5 av dataset 2019,” [urlhttps://level5.lyft.com/dataset](https://level5.lyft.com/dataset), 2019.
- [61] A. Paszke, S. Gross, S. Chintala, G. Chanan, E. Yang, Z. DeVito, Z. Lin, A. Desmaison, L. Antiga, and A. Lerer, “Automatic differentiation in pytorch,” 2017.
- [62] Y. Lee and J. Park, “Centermask: Real-time anchor-free instance segmentation,” in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 13 906–13 915.
- [63] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” *arXiv preprint arXiv:1711.05101*, 2017.
- [64] —, “Sgdr: Stochastic gradient descent with warm restarts,” *arXiv preprint arXiv:1608.03983*, 2016.
- [65] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, “Focal loss for dense object detection,” in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 2980–2988.
- [66] B. Zhu, Z. Jiang, X. Zhou, Z. Li, and G. Yu, “Class-balanced grouping and sampling for point cloud 3d object detection,” *arXiv preprint arXiv:1908.09492*, 2019.
- [67] Z. Chen, Z. Li, S. Zhang, L. Fang, Q. Jiang, and F. Zhao, “Graph-detr3d: rethinking overlapping regions for multi-view 3d object detection,” in *Proceedings of the 30th ACM International Conference on Multimedia (ACMMM)*, 2022, pp. 5999–6008.
- [68] D. Park, R. Ambrus, V. Guizilini, J. Li, and A. Gaidon, “Is pseudo-lidar needed for monocular 3d object detection?” in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021, pp. 3142–3152.
- [69] N. Hendy, C. Sloan, F. Tian, P. Duan, N. Charchut, Y. Xie, C. Wang, and J. Philbin, “Fishing net: Future inference of semantic heatmaps in grids,” *arXiv preprint arXiv:2006.09917*, 2020.
- [70] A. Saha, O. Mendez, C. Russell, and R. Bowden, “Translating images into maps,” in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 9200–9206.
- [71] T. Roddick, A. Kendall, and R. Cipolla, “Orthographic feature transform for monocular 3d object detection,” in *British Machine Vision Conference (BMVC)*, 2019.
- [72] C. Fifty, E. Amid, Z. Zhao, T. Yu, R. Anil, and C. Finn, “Efficiently identifying task groupings for multi-task learning,” in *Advances in Neural Information Processing Systems (NIPS)*, 2021, pp. 27 503–27 516.
- [73] M. Crawshaw, “Multi-task learning with deep neural networks: A survey,” *arXiv preprint arXiv:2009.09796*, 2020.