UniTR: A Unified and Efficient Multi-Modal Transformer for Bird's-Eye-View Representation

Haiyang Wang[1*]   Hao Tang[1,4*]   Shaoshuai Shi[2†]   Aoxue Li[3]

Zhenguo Li[3]   Bernt Schiele[2]   Liwei Wang[1†]

[1]Peking University   [2]Max Planck Institute for Informatics   [3]Huawei, China   [4]Pazhou Laboratory

{wanghaiyang@stu, tanghao@stu, wanglw@cis}.pku.edu.cn

{sshi, schiele}@mpi-inf.mpg.de {liaoxue2, Li.Zhenguo}@huawei.com

Abstract

Jointly processing information from multiple sensors is crucial to achieving accurate and robust perception for reliable autonomous driving systems. However, current 3D perception research follows a modality-specific paradigm, leading to additional computation overheads and inefficient collaboration between different sensor data. In this paper, we present an efficient multi-modal backbone for outdoor 3D perception named UniTR, which processes a variety of modalities with unified modeling and shared parameters. Unlike previous works, UniTR introduces a modality-agnostic transformer encoder to handle these view-discrepant sensor data for parallel modal-wise representation learning and automatic cross-modal interaction without additional fusion steps. More importantly, to make full use of these complementary sensor types, we present a novel multi-modal integration strategy by both considering semantic-abundant 2D perspective and geometry-aware 3D sparse neighborhood relations. UniTR is also a fundamentally task-agnostic backbone that naturally supports different 3D perception tasks. It sets a new state-of-the-art performance on the nuScenes benchmark, achieving +1.1 NDS higher for 3D object detection and +12.0 higher mIoU for BEV map segmentation with lower inference latency. Code will be available at https://github.com/Haiyang-W/UniTR.

1. Introduction

Perceiving the physical world in 3D space is critical for reliable autonomous driving systems [2, 54]. As self-driving sensors become more advanced, integrating the complementary signals captured from different sensors (*e.g.*, Cameras, LiDAR, and Radar) in a unified manner is

---

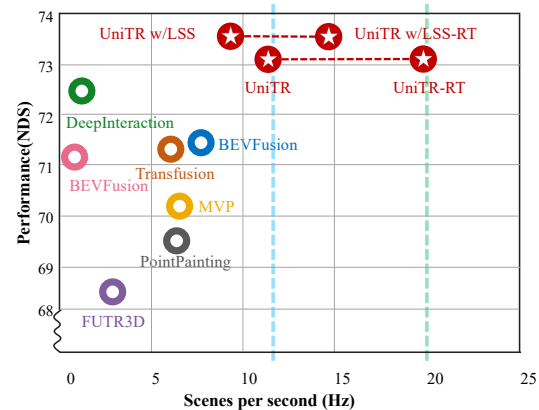[*]Equal contribution.

[†]Corresponding author.

Figure 1. 3D object detection performance (NDS) vs speed (Hz) on nuScenes [3] validation set. Latency is measured on an A100 GPU with AMD EPYC 7513 CPU. Blue and green lines are the operating frequency of the camera and LiDAR in nuScenes.

essential. To achieve this goal, we propose UniTR, a unified yet efficient multi-modal transformer backbone that can process both 3D sparse point clouds and 2D multi-view dense images in parallel to learn the unified bird's-eye-view (BEV) representations for boosting 3D outdoor perception.

Data obtained from multi-sensory systems are represented in fundamentally different modalities: *e.g.*, cameras capture visually rich perspective images, while LiDARs acquire geometry-sensitive point clouds in 3D space. Integrating these complementary sensors is an ideal solution for achieving robust 3D perception. However, due to the view discrepancy in raw data representations, developing an effective fusion approach is non-trivial. Previous works can be broadly classified into point-based, proposal-based, and BEV-based fusion methods. Point-based [51, 52, 22, 70, 53] and proposal-based approaches [4, 72, 1, 30, 5, 66, 28] enrich the LiDAR points and object proposals with semantic features from 2D images separately. BEV-based methods [36, 32] unify the representations of camera and lidar into a shared BEV space and fuse them with subsequent

2D convolutions. Though performant, these fusion schemes generally require modality-specific encoders that process different sensor data in a sequential manner, leading to increased inference latency and hindering their real-world applicability. Thus, developing a modality-agnostic encoder has the potential to efficiently align the multi-modal features while facilitating the learning of generic representations for better 3D scene understanding.

Transformers have emerged as a powerful tool for multi-modal processing in various research fields [6, 25, 47, 77]. However, its application to image-LiDAR data presents unique challenges due to the view discrepancy (*i.e.*, 2D dense images and 3D sparse point clouds). Existing transformer-based fusion strategies [1, 5, 27] rely on modality-specific encoders followed by additional query-based late fusion, incurring non-negligible computational overheads. Hence, building an efficient and unified multi-modal backbone that can automatically learn the intra- and inter-modal representations from both image and LiDAR data is the main challenge we aim to address in this paper.

In this paper, we introduce UniTR, a unified multi-modal transformer backbone for outdoor 3D perception. As shown in Figure 2, unlike previous modality-specific encoders, our UniTR processes the data from multi-sensors in parallel with a modal-shared transformer encoder and integrates them automatically without additional fusion steps. To achieve these goals, we design two major transformer blocks extended on DSVT [56] (*i.e.*, a powerful yet flexible transformer architecture for sparse data). One is the intra-modal block that facilitates parallel computation of modal-wise representation learning for the data from each sensor, and the other one is an inter-modal block to perform cross-modal feature interaction by considering both 2D perspective and 3D geometric neighborhood relations.

Specifically, given single- or multi-view images and point clouds, we first convert them into unified token sequences with lightweight modality-specific tokenizers, *i.e.*, 2D Convolution [23] for images and voxel feature encoding layer [75] for point clouds. To jointly process the modal-wise representation learning of each sensor, these sequences are then partitioned to size-equivalent local sets in their corresponding modal space separately, which are then assigned to different samples in the same batch for parallel computation by several modality-agnostic DSVT blocks. This strategy maximizes the parallel computing capabilities of modern GPU, which greatly reduces the inference latency (about 2× faster) while also achieving better performance by sharing weights among different modalities (see Table 4).

Secondly, we present a powerful yet efficient cross-modal transformer block for camera-lidar fusion. To resolve the view discrepancy, previous methods adopt two view transformations, *i.e.*, LiDAR-to-camera [1, 66, 5, 30] or camera-to-LiDAR projection [36, 32, 41], to unify multi-
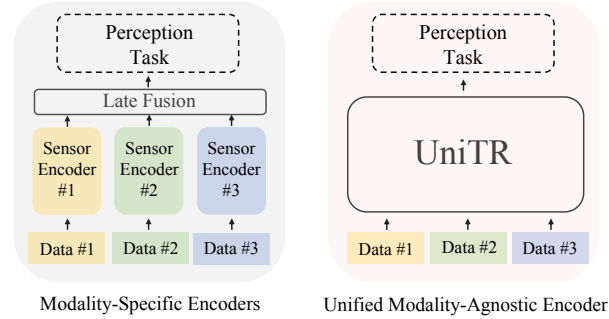

Figure 2. Comparison between sequential modality-specific encoders and our proposed UniTR, which processes various modalities in parallel with a single model and shared parameters.

modal features in a shared space. However, these two fusion spaces are actually complementary due to their distinct neighborhood relations among the inputs, *i.e.*, 2D camera view preserves dense semantic relations, while 3D lidar space provides sparse geometric structures. Combining them can benefit the performance of both geometric- and semantic-oriented 3D perception. More importantly, the time cost brought by additional late fusion behind the modality-specific encoders is normally considerable. To address these problems, we design an inter-modal transformer block to bridge different modalities according to 2D and 3D structural relations. Equipped with this block, our UniTR can integrate multi-modal features with alternative 2D-3D neighborhood configurations during backbone processing, and the performance gains (see Table 3) demonstrate its effectiveness. Notably, this block is also built upon DSVT and can be seamlessly inserted into our multi-modal backbone.

In a nutshell, our contribution is four-fold. 1) We present a weight-sharing intra-modal transformer block for efficient modal-wise representation learning in parallel. 2) To bridge sensors with disparate views, a powerful cross-modal transformer block is designed for integrating different modalities by considering both 2D perspective and 3D geometric structural relations. 3) With the above designs, we introduce a novel multi-modal backbone for outdoor 3D perception named UniTR, which processes a variety of modalities with shared parameters in a unified manner. 4) Our UniTR achieves state-of-the-art performance on nuScenes [3] benchmark of various 3D perception tasks, *i.e.*, 3D Object Detection (+1.1) and BEV Map Segmentation (+12.0), with lower latency, as shown in Figure 1.

We hope the observed strong performance of UniTR can serve as an encouraging benchmark for future efforts toward integrating visual and geometric signals with unified architectures for more generic outdoor 3D perception.

2. Related Work

LiDAR-Based 3D Perception. LiDAR sensors have become indispensable devices providing abundant geometric

information for reliable autonomous driving. Existing lidar-based 3D perception research can be classified into two lines in terms of representations, *i.e.*, point-based and voxel-based methods. Point-based approaches [49, 42, 57, 69, 9] adopt PointNet [43] and its variants [44, 33] to extract features from point clouds. Voxel-based methods [64, 46, 70, 11, 48, 50, 55, 76] first convert point clouds into regular 3D voxels and then process them with sparse convolution [19, 10] or sparse voxel transformer [56, 37, 15, 40, 12, 68, 20]. **Camera-Based 3D Perception.** Perceiving 3D objects only with cameras has been heavily investigated due to the high cost of LiDAR sensors. Previous methods [59, 60, 34, 41, 31] extract 3D information from camera data by adding extra 3D regression branches to 2D detectors [59], designing DETR-based heads with learnable 3D object queries [60, 34] or converting image features from perspective view to bird's-eye view (BEV) using view transformers [41, 21, 31]. However, camera-based methods often face challenges such as limited depth information and occlusion, which require sophisticated modeling and post-processing. **Multi-Sensor 3D Perception.** LiDAR and camera are complementary signals for achieving reliable autonomous driving, which has been well-explored recently. Previous multi-sensor 3D perception approaches can be coarsely classified into three types by fusion, *i.e.*, point-, proposal- and BEV-based methods. Point-based [51, 52, 22, 70, 53, 61] and proposal-based approaches[4, 72, 1, 30, 66, 58, 39] typically leverage image features to augment LiDAR points or 3D object proposals. BEV-based methods [36, 32, 17] efficiently unify the representations of camera and lidar into BEV space, and fuse them with 2D convolution for various 3D perception tasks. However, these successors often rely on sequential processing with modality-specific encoders, followed by additional late fusion steps, which slows down the inference speed and limits real-world applications. To address these challenges, we present a unified and weight-sharing backbone that can efficiently learn intra- and inter-modal representations in a fully parallel manner, enabling faster inference and broader real-world applicability.

3. Revisiting DSVT

Dynamic Sparse Voxel Transformer (DSVT) [56] is a window-based voxel transformer backbone for outdoor 3D perception from point clouds. In order to efficiently handle sparse data in a fully parallel manner, they reformulate the standard window attention [35] as parallel computing self-attention within a series of window-bounded and size-equivalent subsets. To allow the cross-set connection, they design a rotated set partition strategy that alternates between two partition configurations in consecutive attention layers. **Dynamic set partition.** Given the sparse tokens and window partition, they further divide a series of local regions in each window based on their sparsity. As for a specific

window, it has T tokens, $\mathcal{T} = \{t_i\}_{i=1}^T$. S is the required number of sub-sets, which dynamically varies with the window sparsity. Then they evenly distribute T tokens into S sets, (*e.g.*, $\mathcal{Q}_j = \{q_k^j\}_{k=0}^{\tau-1}$ is the token indices of j-th set). The whole process can be formulated as,

$$\{\mathcal{Q}_j\}_{j=0}^{S-1} = \text{DSP}(\mathcal{T}, L \times W \times H, \tau), \qquad (1)$$

where $L \times W \times H$ is the window shape and τ is the size of each set, regardless of T, which enables the attention computation can be efficiently performed on all sets in parallel.

After obtaining the partition $\mathcal{Q}_j$ of j-th set, corresponding token features and coordinates will be collected based on the predefined token inner-window id $\mathcal{I} = \{I_i\}_{i=1}^T$ as,

$$\mathcal{F}_j, \mathcal{C}_j = \text{INDEX}(\mathcal{T}, \mathcal{Q}_j, \mathcal{I}), \qquad (2)$$

where $\text{INDEX}(\cdot_{\text{voxels}}, \cdot_{\text{partition}}, \cdot_{\text{ID}})$ is the index operation, $\mathcal{F}_j \in \mathbb{R}^{\tau \times C}$ and $\mathcal{C}_j \in \mathbb{R}^{\tau \times 3}$ are the respective token features and spatial coordinates (x, y, z) of this set. In this way, they obtain some non-overlapped and size-equivalent subsets for the following parallel attention computation. **Rotated set attention.** To bridge the tokens across these non-overlapping sets, DSVT proposes the rotated-set attention that alternates between two spatially rotated partitioning configurations in successive attention layers as

$$\begin{aligned}
\mathcal{F}^l, \mathcal{C}^l &= \text{INDEX}(\mathcal{T}^{l-1}, \{\mathcal{Q}_j\}_{j=0}^{S-1}, \mathcal{I}_x), \\
\mathcal{T}^l &= \text{MHSA}(\mathcal{F}^l, \text{PE}(\mathcal{C}^l)), \\
\mathcal{F}^{l+1}, \mathcal{C}^{l+1} &= \text{INDEX}(\mathcal{T}^l, \{\mathcal{Q}_j\}_{j=0}^{S-1}, \mathcal{I}_y), \\
\mathcal{T}^{l+1} &= \text{MHSA}(\mathcal{F}^{l+1}, \text{PE}(\mathcal{C}^{l+1})),
\end{aligned} \qquad (3)$$

where $\mathcal{I}_x$ and $\mathcal{I}_y$ are the inner-window voxel index sorted in X- and Y-Axis main order respectively. $\text{MHSA}(\cdot)$ and $\text{PE}(\cdot)$ denote the Multi-head Self-Attention layer and positional encoding. $\mathcal{F} \in \mathbb{R}^{S \times \tau \times C}$ and $\mathcal{C} \in \mathbb{R}^{S \times \tau \times 3}$ are the indexed voxel features and coordinates of all sets. In this way, the original sparse window attention will be approximatively reformulated as several set attention, which can be processed in the same batch in parallel. Notably, the set partition configuration used in DSVT is generalized and flexible to be adapted to different data structures and modalities.

4. Methodology

In this section, we will describe our unified architecture for various modalities (*i.e.*, multi-view cameras and LiDAR) and tasks (*i.e.*, detection and segmentation). Figure 3 illustrates the architecture. Given different sensory inputs, the model first converts them into token sequences with modality-specific tokenizers. A modality-agnostic Transformer backbone is then employed to perform single- and cross-modal representation learning (§4.1-§4.2) for boosting various 3D perception tasks (§4.3).
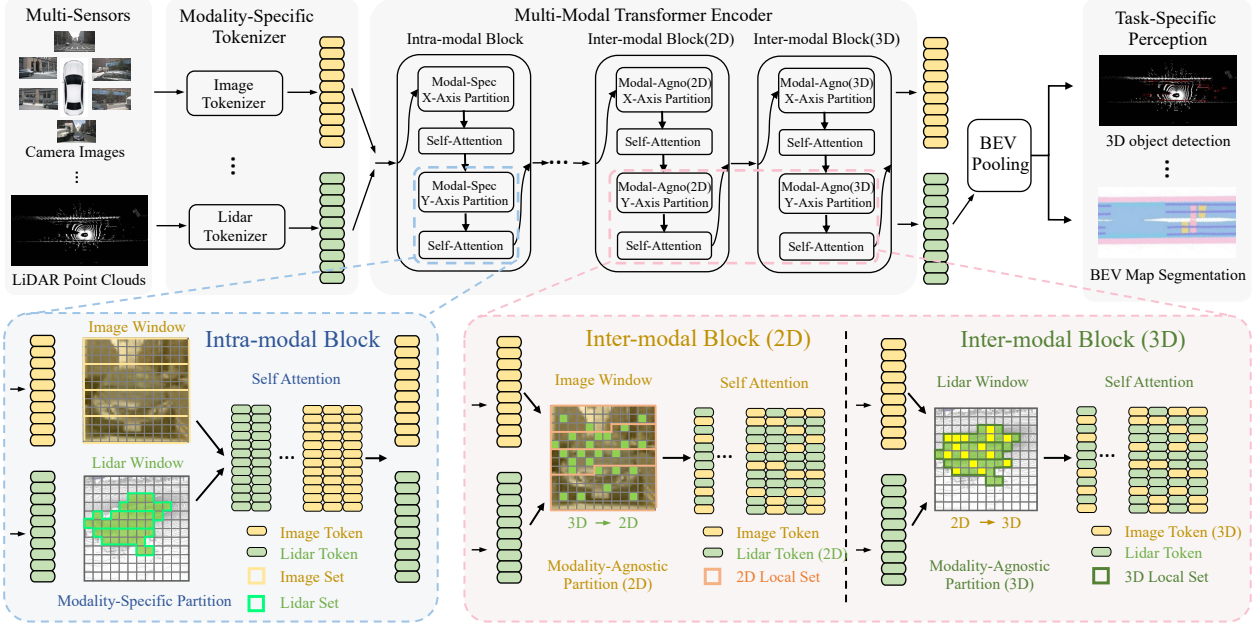
Figure 3. An illustration of our UniTR. Given different sensory inputs, the model first converts them into token sequences with modality-specific tokenizers. A multi-modal transformer encoder is then employed to perform single- and cross-modal representation learning and efficiently pools the semantically enriched lidar tokens into a BEV space for boosting various 3D perception tasks.

4.1. Single-Modal Representation Learning

Perceiving 3D scenes in autonomous driving scenarios requires reliable representations of multiple modalities, *e.g.*, multi-view images and sparse point clouds. Due to their discrepant representations, previous approaches [27, 5, 36, 32] encode features of each modality by separate encoders that are generally processed sequentially, slowing down the inference speed and limiting their real-world applications. To tackle these problems, we propose to process intra-modal representation learning of each sensor with a unified architecture in parallel, whose parameters are shared for all modalities. Our approach begins by converting different modal inputs into token sequences using modality-specific tokenizers, followed by several modal-shared DSVT blocks that enable parallel modal-wise feature encoding.

Tokenization. Given the raw images X^I captured from $\mathcal{B}$ cameras and point clouds X^P obtained from LiDAR, modality-specific tokenizers are applied to generate the input token sequences for the succeeding transformer encoder. Specifically, we use the image patch tokenizer [13] for image modality and the dynamic voxel feature encoding tokenizer [75] for point cloud modality. As illustrated in Figure 3, by adopting these two tokenizers, the input sequence $\mathcal{T} \in \mathbb{R}^{(M+N) \times C}$ of the following intra-modal transformer block can be composed of N point cloud tokens $\mathcal{T}^P \in \mathbb{R}^{N \times C}$ and M image tokens $\mathcal{T}^I \in \mathbb{R}^{M \times C}$. More details of these tokenizers are described in Appendix.

Modality-specific set attention. To efficiently process intra-modal representation learning among the given multi-modal tokens in parallel, we first perform the dynamic set partition introduced in [56] for each sensor individually.

Specifically, after obtaining the lidar and image tokens from a specific scene, *i.e.*,

$$\mathcal{T}^P = \{t_i^P | t_i^P = [(x_i^P, y_i^P, z_i^P); f_i^P]\}_{i=1}^N,$$
$$\mathcal{T}^I = \{t_i^I | t_i^I = [(x_i^I, y_i^I, b_i^I); f_i^I]\}_{i=1}^M, \quad (4)$$

where $(x_i^{(*)}, y_i^{(*)}, z_i^{(*)}) \in \mathbb{R}^3$ and $f_i^{(*)} \in \mathbb{R}^C$ denote the coordinates and features of lidar and image tokens. $b_i^I \in [0, \mathcal{B} - 1]$ is the corresponding view ID of the i-th image patch. We compute the token indices of each local set in its respective modal space as follows,

$$\{\mathcal{Q}_n^P\}_{n=0}^{\mathcal{N}} = \text{DSP}(\mathcal{T}^P, L^P \times W^P \times H^P, \tau),$$
$$\{\mathcal{Q}_m^I\}_{m=0}^{\mathcal{M}} = \text{DSP}(\mathcal{T}^I, L^I \times W^I \times 1, \tau), \quad (5)$$

where $\text{DSP}(\cdot_{\text{input}}, \cdot_{\text{window size}}, \cdot_{\text{token number of each set}})$ is the standard dynamic set partitioning strategy introduced in DSVT [56]. $(L^{(*)}, W^{(*)}, H^{(*)})$ and τ are the window size and the token number of each set used in this step. $\mathcal{N}$ and $\mathcal{M}$ are the number of partitioned subsets for lidar and image respectively. Note that image window size $(L^I \times W^I \times 1)$ indicates that the image set partition is only performed within each camera view individually.

Given the modal-wise image-lidar partition, $\{\mathcal{Q}_n^P\}_{n=0}^{\mathcal{N}}$ and $\{\mathcal{Q}_m^I\}_{m=0}^{\mathcal{M}}$, we then collect the token features and co-ordinates assigned to each set and perform parallel attention computation for each modality as,

$$\mathcal{F}_l, \mathcal{C}_l = \text{INDEX}([\mathcal{T}_{l-1}^P, \mathcal{T}_{l-1}^I], [\{\mathcal{Q}_n^P\}_{n=0}^{\mathcal{N}}, \{\mathcal{Q}_m^I\}_{m=0}^{\mathcal{M}}]),$$
$$\widetilde{\mathcal{T}}_l^P, \widetilde{\mathcal{T}}_l^I = \text{MHSA}(\mathcal{F}_l, \text{PE}(\mathcal{C}_l)), \quad (6)$$

where INDEX is the index function described in Eq. (2) (we remove the inner-window ID here for simplicity) and $[\cdot, \cdot]$ means concatenation operation. $\mathcal{F} \in \mathbb{R}^{(\mathcal{M}+\mathcal{N}) \times \tau \times C}$ and $\mathcal{C} \in \mathbb{R}^{(\mathcal{M}+\mathcal{N}) \times \tau \times 3}$ are the multi-modal token features and spatial coordinates. MHSA$(\cdot)$ denotes the standard Multi-head Self-Attention layer with FFN and Layer Normalization, propagating information among τ tokens in each local set. PE$(\cdot)$ stands for the positional encoding function. In this way, the modal-wise representation learning will be performed by a series of modality-specific set attention in parallel. Notably, our model shares self-attention weights across modalities, enabling parallel computation and making it more computationally efficient (about $2\times$ faster) than processing each modality separately (see Table 4).

4.2. Cross-Modal Representation Learning

To effectively integrate the view-discrepant information from multiple sensors in autonomous driving scenarios, existing methods typically involve designing separate deep models for each sensor and fusing the information via post-processing approaches, such as augmenting 3D object proposals [1] or lidar-only BEV maps [36, 32] with semantic features from image views. Moreover, prior to the fusion step, all the sensor data are usually converted into a unified representation space (*i.e.*, 3D/BEV space [36, 32] or 2D perspective space [1, 66]), which are either semantic-lossy or geometry-diminished. To allow the efficient cross-modal connection and make full use of these two complementary representations, we design two partitioning configurations alternated in consecutive modality-agnostic DSVT blocks, which can automatically fuse the multi-modal data by considering both the 2D and 3D structural relations.

Image perspective space. To bridge the multi-sensor data in semantic-aware 2D image space, we first transform all the lidar tokens to the image plane by utilizing the camera's intrinsic and extrinsic parameters. To create a one-to-one projection for the succeeding unified perspective partition, we only take the first view hit of 3D lidar tokens and place them in their respective 2D positions as follows,

$$\mathcal{T}^P \to \mathcal{T}^P_{2D} : (x^P, y^P, z^P) \to (x^P_{2D}, y^P_{2D}, b^P_{2D}), \quad (7)$$

where b^P_{2D} is the identified view ID of each lidar token.

After unifying both the image and lidar tokens into camera perspective space, we use the conventional dynamic set partition to generate $\widetilde{\mathcal{M}}$ cross-modal 2D local sets in a modality-agnostic manner, *i.e.*,

$$\{\mathcal{Q}^{2D}_m\}^{\widetilde{\mathcal{M}}}_{m=0} = \text{DSP}([\mathcal{T}^P_{2D}, \mathcal{T}^I], L^I \times W^I \times 1, \tau), \quad (8)$$

where $[\cdot, \cdot]$ denotes concatenation. This step employs a modality-agnostic manner to group multi-modal tokens into the same local sets based on their positions in camera perspective space. Then these modality-mixed subsets are used by several DSVT blocks for 2D cross-modal interaction.

3D geometric space. To unify the multi-modal inputs in 3D space, an efficient and robust view projection is required to uniquely map the image patches into 3D space. However, depth-degraded camera-to-lidar transformation is an ill-posed problem due to the inherent ambiguity of the depth associated with each image pixel. Although previous learnable depth estimators [18, 16] can predict a depth image with acceptable accuracy, they require an additional computation-intensive prediction module and suffer from poor generalization ability. To overcome these limitations, inspired by MVP [71], we present a non-learnable and totally pre-computable approach to efficiently transform image patches into 3D space for the succeeding partition.

Specifically, we first sample a group of pseudo 3D grid points, $\mathcal{V}^P \in \mathbb{R}^{L^S \times W^S \times H^S \times 3}$, where L^S, W^S, H^S are the spatial shape of the 3D space divided by predefined grid size. Then we project all the pseudo lidar points, $\mathcal{V}^P$, into their corresponding virtual image coordinates (denote as $\mathcal{V}^I = \{v_k | (x_k, y_k); b_k; d_k)\}^\nu_{k=0}$), where b_k and d_k are the view ID and associated depth. Notably, only the projected points that fall within the view images will be considered, so the number of valid image points $\nu \leq |\mathcal{V}^P| \times \mathcal{B}$.

With these pseudo image points $\mathcal{V}^I$, we retrieve the depth estimate of each image token from its nearest image virtual neighbor as follows,

$$\{v^{\text{nearest}}_i | (x_i, y_i); b_i; d_i\}^M_{i=0} = \text{Nearest}(\mathcal{V}^I, \mathcal{T}^I), \quad (9)$$

where $\text{Nearest}(\cdot_{\text{virtual points}}, \cdot_{\text{image tokens}})$ is the nearest neighbor function and d_i represents the computed depth of i-th image token, which is the same as its nearest neighbor. Then we unproject the image tokens $\mathcal{T}^I$ back to 3D space and generate offset features based on their 2D distance, *i.e.*,

$$
\begin{aligned}
&\mathcal{T}^I \to \mathcal{T}^I_{3D} : (x^I, y^I, b^I) \to (x^I_{3D}, y^I_{3D}, z^I_{3D}), \\
&f^I \to f^I + \text{MLP}(||v^{\text{nearest}}(x, y) - (x^I, y^I)||).
\end{aligned} \quad (10)
$$

In this way, we unify the image and lidar tokens in 3D space by a pre-calculable view projection, which can be totally cached during inference. Finally, a dynamic set partition module is applied to generate 3D cross-modal local sets as

$$\{\mathcal{Q}^{3D}_n\}^{\widetilde{\mathcal{N}}}_{n=0} = \text{DSP}([\mathcal{T}^P, \mathcal{T}^I_{3D}], L^P \times W^P \times H^P, \tau), \quad (11)$$

which is then processed by a DSVT block in a unified manner for cross-modal interaction in 3D lidar space.

Our cross-modal transformer block introduces connections between different modalities by considering both 2D and 3D structural relations and is found to be effective in multi-modal 3D perception, as shown in Table 3. Importantly, it shares the same fundation (DSVT) of intra-modal block and can be integrated into our backbone seamlessly.

4.3. Perception Task Setup

UniTR can be adapted to most 3D perception tasks. To demonstrate its versatility, we evaluate it on two important

Methods	Present at	Modality	NDS (*val*)	mAP (*val*)	NDS (*test*)	mAP (*test*)	Latency (*ms*)
BEVFormer [31]	ECCV'22	C	-	-	56.9	48.1	–
BEVDepth [29]	AAAI'23	C	-	-	60.0	50.3	–
BEVFormer v2 [65]	CVPR'23	C	-	-	63.4	55.6	–
SECOND [64]	Sensors'18	L	63.0	52.6	63.3	52.8	53.2
PointPillars [24]	CVPR'19	L	61.3	52.3	–	–	28.1
CenterPoint [70]	CVPR'21	L	66.8	59.6	67.3	60.3	62.7
PointPainting [52]	CVPR'20	C+L	69.6	65.8	-	-	151.8
PointAugmenting [53]	CVPR'21	C+L	-	-	71.0†	66.8†	188.4
MVP [71]	NeurIPS'21	C+L	70.0	66.1	70.5	66.4	148.1
FusionPainting [63]	ITSC'21	C+L	70.7	66.5	71.6	68.1	-
FUTR3D [5]	ArXiv'22	C+L	68.3	64.5	-	-	302.6
AutoAlign [8]	IJCAI'22	C+L	71.1	66.6	-	-	-
TransFusion [1]	CVPR'22	C+L	71.3	67.5	71.6	68.9	164.6
AutoAlignV2 [7]	ECCV'22	C+L	71.2	67.1	72.4	68.4	207.0
UVTR [27]	NeurIPS'22	C+L	70.2	65.4	71.1	67.1	-
BEVFusion (PKU) [32]	NeurIPS'22	C+L	71.0	67.9	71.8	69.2	1231.0
DeepInteraction [67]	NeurIPS'22	C+L	72.6	69.9	73.4	**70.8**	541.1
BEVFusion (MIT) [36]	ICRA'23	C+L	71.4	68.5	72.9	70.2	130.5
UniTR (Ours)	ICCV'23	C+L	**73.1**	**70.0**	**74.1**	70.5	**88.7 (50.2‡)**
UniTR w/ LSS (Ours)	ICCV'23	C+L	**73.3**	**70.5**	**74.5**	**70.9**	**107.5 (69.1‡)**

Table 1. Performance of 3D detection on nuScenes (val and test) dataset [3]. Notion of modality: Camera (C), LiDAR (L). $\dagger$: with test-time augmentation. $\ddagger$: deployed by TensorRT). We highlight the top-2 entries with **bold** font in each column.

tasks: 3D object detection and BEV map segmentation.

Detection. Without loss of generality, we follow the same detection framework adopted in BEVFusion [36]. Notably, we only switch its multiple modality-specific encoders to ours and remove the redundant late fusion module while all other settings remain unchanged. To fully exploit geometrically enhanced image features, we also present an augmented variant further strengthened by an additional LSS-based BEV fusion step [36, 32]. More implementation details are described in previous 3D detection papers [36, 1].

Segmentation. We adopt the same framework as our 3D object detection task, with the exception of segmentation head and evaluation protocol used in BEVFusion [36].

5. Experiments

In this section, we conduct experiments on 3D object detection and BEV map segmentation tasks using nuScenes dataset [3], covering both geometric- and semantic-oriented 3D perceptions. Actually, owing to the inclusive 2D perspective and 3D geometric spaces, our unified backbone also offers versatility and seamless extensibility to accommodate other sensor types, *e.g.*, radars and ultrasonic, as well as various other 3D perception tasks, *e.g.*, 3D object tracking [70] and motion prediction [14].

5.1. Implementation Details

Backbone. Our UniTR starts with two modality-specific tokenizers, a DVFE layer [75] for point clouds voxelization with grid size (0.3m, 0.3m, 8m) of detection and (0.4m, 0.4m, 8m) of segmentation, and an image patch tokenizer [13] for $8\times$ downsampling camera images to 32×88. Then one weight-sharing intra-modal UniTR block is followed for processing modal-wise representation learning in parallel. Finally, three inter-modal UniTR blocks cross the boundaries of different modalities and provide connection among them by alternating between 2D and 3D partitioning configurations consecutively. The block configuration adopted in this paper is {intra, inter$_{2D}$, inter$_{2D}$, inter$_{3D}$}. More elaborated details are in Appendix.

Dataset. We conduct our experiments on nuScenes [3], a challenging large-scale outdoor benchmark that provides diverse annotations for various tasks, (*e.g.*, 3D object detection [70] and BEV map segmentation [36, 26]). It contains 40,157 annotated samples, each with six monocular camera images covering a 360-degree FoV and a 32-beam LiDAR.

Training. Unlike the dominant two-step training strategies [36, 32, 1] that contain separate single-modal pre-training and joint multi-modal post-training, our UniTR is trained by a simpler one-step training scheme in an end-to-end manner with aligned multi-modal data augmentations. All the experiments are trained by AdamW optimizer [38] on 8 A100 GPUs. See appendix for more details.

5.2. 3D Object Detection

Setting. We benchmark UniTR on nuScenes [3] dataset, which contains 1.4 million annotated 3D bounding boxes for 10 classes. Without loss of generality, we follow the framework of BEVFusion [36] and place our UniTR before its final multi-modal BEV encoder. The nuScenes detection score (NDS) and mean average precision (mAP) are

Methods	Modality	Drivable	Ped. Cross.	Walkway	Stop Line	Carpark	Divider	Mean IoU
OFT [45]	C	74.0	35.3	45.9	27.5	35.9	33.9	42.1
LSS [41]	C	75.4	38.8	46.3	30.3	39.1	36.5	44.4
CVT [74]	C	74.3	36.8	39.9	25.8	35.0	29.4	40.2
M^{2}BEV [62]	C	77.2	–	–	–	–	40.5	–
BEVFusion [36]	C	81.7	54.8	58.4	47.4	50.7	46.4	56.6
PointPillars [24]	L	72.0	43.1	53.1	29.7	27.7	37.5	43.8
CenterPoint [70]	L	75.6	48.4	57.5	36.5	31.7	41.9	48.6
UniTR (Ours)	L	**89.6**	**71.4**	**75.8**	**64.7**	**64.4**	**61.5**	**71.2**
PointPainting [52]	C+L	75.9	48.5	57.1	36.9	34.5	41.9	49.1
MVP [71]	C+L	76.1	48.7	57.0	36.9	33.0	42.2	49.0
BEVFusion [36]	C+L	85.5	60.5	67.6	52.0	57.0	53.7	62.7
UniTR (Ours)	C+L	**90.4**	**73.1**	**78.2**	**66.6**	**67.3**	**63.8**	**73.2**
UniTR w/ LSS (Ours)	C+L	**90.5**	**73.8**	**79.1**	**68.0**	**72.7**	**64.0**	**74.7**

Table 2. UniTR outperforms the state-of-the-art multi-sensor fusion methods on BEV map segmentation on nuScenes (val), which demonstrates the effectiveness of our unified backbone for semantic-oriented 3D perception tasks. All baselines are reported in BEVFusion [36].

Modality	3D Space (L)	2D Space (C)	BEVLSS	NDS	mAP
L				70.5	65.9
C+L	✓			72.0	68.5
C+L		✓		72.5	69.0
C+L	✓	✓		73.1	70.0
C+L	✓	✓	✓	73.3	70.5

Table 3. Effect of camera image space fusion, 3D lidar geometric space fusion and BEV unifier on nuScenes (val). 1st row is the lidar-only variant of our model. Camera (C), LiDAR (L).

Modality	Serial	Parallel	Latency(ms)	NDS	mAP
C			28	36.2	31.4
L			26	70.5	65.9
C+L	✓		51	72.2	68.5
C+L		✓	33	72.4	69.0

Table 4. Effect of parallel intra-modal transformer block.

reported. We also measure the latency of all open-source methods on the same workstation with an A100 GPU. Notably, we only use a single model without any test-time augmentation for both validation and test results.

Results. As shown in Table 1, our UniTR outperforms all previous LiDAR-camera fusion methods with much lower inference latency (88.7ms). It achieves state-of-the-art performance, 73.1 and 74.1 in *val* and *test* NDS, surpassing BEVFusion [36] by +1.7 and +1.2 separately. After incorporating an additional LSS-based BEV fusion step, our UniTR achieves an enhanced performance, reaching 73.3 and 74.5 in *val* and *test* NDS, respectively. Notably, our model is built upon DSVT [56] and inherits its deployment-friendly characteristic, making it easily accelerated by well-optimized deployment tools (*i.e.*, TensorRT) to lower inference latency (50.2ms). To the best of our knowledge, our method is the first to achieve state-of-the-art performance with close-to-real-time running speed (about 20 *Hz*).

5.3. BEV Map Segmentation

Setting. To further demonstrate the generalizability of our approach, we follow BEVFusion [36] and evaluate our

model on BEV Map Segmentation task. This segmentation variant is adapted from our multi-modal detection approach, and replaced by the perception head and evaluation protocol used in BEVFusion. The per-class and averaged mean IoU scores are reported for our evaluation metric.

Results. We report the comparison results with state-of-the-art methods in Table 2. Our unitr and enhanced variant achieve 73.2 and 74.7 mIoU on nuScenes validation set respectively, which outperforms previous best result [36] by +10.5 and +12.0, demonstrating its effectiveness of semantic-oriented 3D perception tasks.

5.4. Ablation Studies and Discussions

We conduct comprehensive ablation studies on the validation set of nuScenes 3D object detection to analyze individual components of our proposed method. Notably, all the experiments are performed on our BEV enhanced variant.

Effect of 2D & 3D fusion. We first ablate the effects of our 2D & 3D cross-modal representation learning blocks in Table 3. The base competitor (1st row) is our lidar-only variant that abandons image information. To make a fair comparison, we only switch the fusion algorithm while all other settings remain unchanged (such as the number of transformer blocks). See appendix for more details. As evidenced in the 1st, 2nd and 3rd rows, processing cross-modal interaction by considering the perspective 2D image space and 3D sparse space separately brings considerable performance gains of our model, *i.e.*, $70.5 \rightarrow 72.5$, $70.5 \rightarrow 72.0$ on NDS. Combining both of them (4th rows), our model can achieve a significant improvement, *i.e.*, $70.5 \rightarrow 73.1$, $65.9 \rightarrow 70.0$ on NDS and mAP. That verifies our motivation that the semantic-rich 2D perspective view and geometric-sensitive 3D view are two complementary spaces for boosting 3D perception performance.

Effect of parallel intra-modal transformer block. Table 4 ablates the effectiveness of our parallel intra-modal transformer block. The 1st and 2nd rows are the pure-image

Approach	Clean		Missing F		Preserve F		Stuck	
	mAP	NDS	mAP	NDS	mAP	NDS	mAP	NDS
DETR3D* [60]	34.9	43.4	25.8	39.2	3.3	20.5	17.3	32.3
PointAugmenting [53]	46.9	55.6	42.4	53.0	31.6	46.5	42.1	52.8
MVX-Net [51]	61.0	66.1	47.8	59.4	17.5	41.7	48.3	58.8
TransFusion [1]	66.9	70.9	65.3	70.1	64.4	69.3	65.9	70.2
BEVFusion [32]	67.9	71.0	65.9	70.7	65.1	69.9	66.2	70.3
UniTR (Ours)	70.5	73.3	68.5	72.4	66.5	71.2	68.1	71.8

Table 5. Results on robustness setting of camera failure cases. F denotes the front camera, and * means camera-only inputs. All the experiments are on nuScenes validation set.

Aug	Metrics	LiDAR	BEVFusion[32]	Ours
	mAP	31.3	40.2 (+8.9)	38.3(+7.0)
	NDS	50.7	54.3 (+3.6))	55.6(+4.9)
✓	mAP	-	54.0(+22.7)	60.2(+28.9)
✓	NDS	-	61.6(+10.9)	66.0(+15.3)

Table 6. Results on robustness setting of object failure cases. We refer readers to [32] for more details.

Modality	Camera	Lidar	Serial	Parallel	Block config	NDS	mAP
C+L(1-beam)	36.2	42.0	57.6	**59.5**	3D → 2D	73.0	70.0
C+L(4-beam)	36.2	62.1	67.3	**68.5**	2D → 3D	**73.3**	**70.5**
C+L(16-beam)	36.2	69.1	71.6	**72.2**	Inter → Intra	72.9	69.8
C+L(32-beam)	36.2	70.5	73.2	**73.3**	Intra → Inter	**73.3**	**70.5**

Table 7. Ablation of low beam setting with NDS evaluation metric.   Table 8. Results of different block configurations.

Models	Latency (ms)	NDS	mAP
PointPainting [52]	151.8	69.6	65.8
TransFusion [1]	164.6	71.3	67.5
BEVFusion (PKU) [32]	1231.0	71.0	67.9
BEVFusion (MIT) [36]	130.5	71.4	68.5
UniTR	**88.7**	**73.1**	**70.0**
UniTR (TensorRT)	**50.2**	**73.1**	**70.0**

Table 9. The latency and performance on nuScenes val set.

and pure-lidar baselines of our method. To better evaluate the effectiveness of this weight-sharing manner, we remove the powerful 2D&3D fusion algorithm except for the final late fusion module and restrict backbone to only carry out intra-modal representation learning within each modality. By comparing the 3rd and 4th rows, we find that weight-sharing backbone can achieve slightly better performance than previous sequential manner with much lower inference latency, i.e, 51ms → 33ms. We argue that the unified encoder is naturally suitable for aligning representations of different modalities, especially for the closely related and complementary image-lidar pairs, which encourages our model to overcome the modality gap and learn generic representations for 3D outdoor perception more easily.

As for the faster running speed, attention computation is very fast, so involving more tokens in this operation doesn't significantly slow down the inference, e.g., lidar-only (26ms) vs parallel image-lidar (33ms). However, calling attention frequently will introduce some additional overheads, (e.g., memory access), and increase the latency. **Effect of different block configurations.** All the blocks in UniTR are independent and can be flexibly configured. Table 8 presents the results for various block setups. The "Inter → Intra" configuration exhibits inferior performance, possibly due to the challenges encountered when fusing modalities for shallow features. Similarly, the "3D → 2D" configuration shows slightly lower performance. We argue that it is more advantageous for the last block to perform fusion in the 3D space, as it aligns with the necessity of converting features into 3D for perception.

5.5. Robustness Against Sensor Failure

We follow the same evaluation protocols adopted in BEVFusion [32] to demonstrate the robustness of our UniTR for lidar and camera malfunctioning. We refer readers to [73, 32] for more implementation details. All the experiments are conducted on nuScenes validation set.

As shown in Table 5 and 6, our method outperforms all the single- and multi-modal methods regardless of any Li-DAR or camera malfunction scenarios, which demonstrates the robustness of UniTR against sensors failure cases. To further illustrate the motivation behind unification, the combination of cameras with low-beam LiDAR has been conducted. We followed the same setting adopted in BEVFusion [36]. As depicted in Table 7, our unified modeling can better leverage the complementary information among different sensors, particularly benefiting the sparser LiDAR and leading to enhanced performance robustness.

5.6. Inference Speed

We present a comparison with other state-of-the-art methods on both inference latency and performance accuracy in Table 9. Our model significantly outperforms BEV-Fusion [36] while achieving a lower latency, i.e., 88.7 ms vs 130.5 ms. Thanks to its deployment-friendly characteristic, our model can reach a close-to-real-time running speed (about 20Hz) after being deployed by TensorRT. All the experiments are evaluated on the same workstation.

6. Conclusion

In this paper, we propose a unified backbone that processes various modalities with a single model and shared parameters for outdoor 3D perception. With the specially designed modality-agnostic transformer blocks for intra- and inter-modal representation learning, our method can achieve state-of-the-art performance with remarkable gains of various 3D perception tasks on challenging nuScenes dataset. It is our belief that UniTR can provide a strong foundation for facilitating the development of more efficient and generic outdoor 3D perception systems.

References

[1] Xuyang Bai, Zeyu Hu, Xinge Zhu, Qingqiu Huang, Yilun Chen, Hongbo Fu, and Chiew-Lan Tai. Transfusion: Robust lidar-camera fusion for 3d object detection with transformers. In *CVPR*, 2022. 1, 2, 3, 5, 6, 8

[2] Mayank Bansal, Alex Krizhevsky, and Abhijit Ogale. Chauffeurnet: Learning to drive by imitating the best and synthesizing the worst. 2018. 1

[3] 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 *CVPR*, 2020. 1, 2, 6

[4] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *CVPR*, 2017. 1, 3

[5] Xuanyao Chen, Tianyuan Zhang, Yue Wang, Yilun Wang, and Hang Zhao. Futr3d: A unified sensor fusion framework for 3d detection. *arXiv preprint arXiv:2203.10642*, 2022. 1, 2, 4, 6

[6] Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. Uniter: Universal image-text representation learning. In *ECCV*, 2020. 2

[7] Zehui Chen, Zhenyu Li, Shiquan Zhang, Liangji Fang, Qinhong Jiang, and Feng Zhao. Autoalignv2: Deformable feature aggregation for dynamic multi-modal 3d object detection. *arXiv preprint arXiv:2207.10316*, 2022. 6

[8] Zehui Chen, Zhenyu Li, Shiquan Zhang, Liangji Fang, Qinghong Jiang, Feng Zhao, Bolei Zhou, and Hang Zhao. Autoalign: Pixel-instance feature aggregation for multimodal 3d object detection. In *IJCAI*, 2022. 6

[9] Bowen Cheng, Lu Sheng, Shaoshuai Shi, Ming Yang, and Dong Xu. Back-tracing representative points for voting-based 3d object detection in point clouds. In *CVPR*, 2021. 3

[10] Christopher Choy, JunYoung Gwak, and Silvio Savarese. 4d spatio-temporal convnets: Minkowski convolutional neural networks. In *CVPR*, 2019. 3

[11] Jiajun Deng, Shaoshuai Shi, Peiwei Li, Wen gang Zhou, Yanyong Zhang, and Houqiang Li. Voxel r-cnn: Towards high performance voxel-based 3d object detection. In *AAAI*, 2021. 3

[12] Shaocong Dong, Lihe Ding, Haiyang Wang, Tingfa Xu, Xinli Xu, Jie Wang, Ziyang Bian, Ying Wang, and Jianan Li. Mssvt: Mixed-scale sparse voxel transformer for 3d object detection on point clouds. In *NeurIPS*, 2022. 3

[13] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *ICLR*, 2021. 4, 6

[14] Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles R Qi, Yin Zhou, et al. Large scale interactive motion forecasting for autonomous driving: The waymo open motion dataset. In *ICCV*, 2021. 6

[15] 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 *CVPR*, 2022. 3

[16] Huan Fu, Mingming Gong, Chaohui Wang, Kayhan Batmanghelich, and Dacheng Tao. Deep ordinal regression network for monocular depth estimation. In *CVPR*, 2018. 5

[17] Yulu Gao, Chonghao Sima, Shaoshuai Shi, Shangzhe Di, Si Liu, and Hongyang Li. Sparse dense fusion for 3d object detection. *IROS*, 2023. 3

[18] Clément Godard, Oisin Mac Aodha, and Gabriel J Brostow. Unsupervised monocular depth estimation with left-right consistency. In *CVPR*, 2017. 5

[19] Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *CVPR*, 2018. 3

[20] Chenhang He, Ruihuang Li, Shuai Li, and Lei Zhang. Voxel set transformer: A set-to-set approach to 3d object detection from point clouds. In *CVPR*, 2022. 3

[21] Junjie Huang, Guan Huang, Zheng Zhu, and Dalong Du. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*, 2021. 3

[22] Tengteng Huang, Zhe Liu, Xiwu Chen, and Xiang Bai. Epnet: Enhancing point features with image semantics for 3d object detection. In *ECCV*, 2020. 1, 3

[23] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 2017. 2

[24] Alex H Lang, Sourabh Vora, Holger Caesar, Lubing Zhou, Jiong Yang, and Oscar Beijbom. Pointpillars: Fast encoders for object detection from point clouds. In *CVPR*, 2019. 6, 7

[25] Liunian Harold Li, Mark Yatskar, Da Yin, Cho-Jui Hsieh, and Kai-Wei Chang. Visualbert: A simple and performant baseline for vision and language. *arXiv preprint arXiv:1908.03557*, 2019. 2

[26] Qi Li, Yue Wang, Yilun Wang, and Hang Zhao. Hdmapnet: An online hd map construction and evaluation framework. In *ICRA*, 2022. 6

[27] Yanwei Li, Yilun Chen, Xiaojuan Qi, Zeming Li, Jian Sun, and Jiaya Jia. Unifying voxel-based representation with transformer for 3d object detection. In *NeurIPS*, 2022. 2, 4, 6

[28] Yingyan Li, Lue Fan, Yang Liu, Zehao Huang, Yuntao Chen, Naiyan Wang, Zhaoxiang Zhang, and Tieniu Tan. Fully sparse fusion for 3d object detection. *arXiv preprint arXiv:2304.12310*, 2023. 1

[29] Yinhao Li, Zheng Ge, Guanyi Yu, Jinrong Yang, Zengran Wang, Yukang Shi, Jianjian Sun, and Zeming Li. Bevdepth: Acquisition of reliable depth for multi-view 3d object detection. 2023. 6

[30] Yingwei Li, Adams Wei Yu, Tianjian Meng, Ben Caine, Jiquan Ngiam, Daiyi Peng, Junyang Shen, Yifeng Lu, Denny Zhou, Quoc V Le, et al. Deepfusion: Lidar-camera deep fusion for multi-modal 3d object detection. In *CVPR*, 2022. 1, 2, 3

[31] 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 *ECCV*, 2022. 3, 6

[32] Tingting Liang, Hongwei Xie, Kaicheng Yu, Zhongyu Xia, Zhiwei Lin, Yongtao Wang, Tao Tang, Bing Wang, and Zhi Tang. Bevfusion: A simple and robust lidar-camera fusion framework. 2022. 1, 2, 3, 4, 5, 6, 8

[33] Yongcheng Liu, Bin Fan, Shiming Xiang, and Chunhong Pan. Relation-shape convolutional neural network for point cloud analysis. In *CVPR*, 2019. 3

[34] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. In *ECCV*, 2022. 3

[35] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *ICCV*, 2021. 3

[36] Zhijian Liu, Haotian Tang, Alexander Amini, Xinyu Yang, Huizi Mao, Daniela Rus, and Song Han. Bevfusion: Multi-task multi-sensor fusion with unified bird's-eye view representation. In *ICRA*, 2023. 1, 2, 3, 4, 5, 6, 7, 8

[37] Zhijian Liu, Xinyu Yang, Haotian Tang, Shang Yang, and Song Han. Flatformer: Flattened window attention for efficient point cloud transformer. In *CVPR*, 2023. 3

[38] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019. 6

[39] Haoyu Lu, Yuqi Huo, Mingyu Ding, Nanyi Fei, and Zhiwu Lu. Cross-modal contrastive learning for generalizable and efficient image-text retrieval. *Machine Intelligence Research*, pages 1–14, 2023. 3

[40] Jiageng Mao, Yujing Xue, Minzhe Niu, Haoyue Bai, Jiashi Feng, Xiaodan Liang, Hang Xu, and Chunjing Xu. Voxel transformer for 3d object detection. In *ICCV*, 2021. 3

[41] Jonah Philion and Sanja Fidler. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *ECCV*, 2020. 2, 3, 7

[42] Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3d object detection in point clouds. In *ICCV*, 2019. 3

[43] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *CVPR*, 2017. 3

[44] Charles Ruizhongtai Qi, Li Yi, Hao Su, and Leonidas J Guibas. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. In *NeurIPS*, 2017. 3

[45] Thomas Roddick, Alex Kendall, and Roberto Cipolla. Orthographic feature transform for monocular 3d object detection. 2018. 7

[46] Shaoshuai Shi, Chaoxu Guo, Li Jiang, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-rcnn: Point-voxel feature set abstraction for 3d object detection. In *CVPR*, 2020. 3

[47] Shaoshuai Shi, Li Jiang, Dengxin Dai, and Bernt Schiele. Motion transformer with global intention localization and local movement refinement. In *NeurIPS*, 2022. 2

[48] Shaoshuai Shi, Li Jiang, Jiajun Deng, Zhe Wang, Chaoxu Guo, Jianping Shi, Xiaogang Wang, and Hongsheng Li. Pv-

rcnn++: Point-voxel feature set abstraction with local vector representation for 3d object detection. *IJCV*, 2022. 3

[49] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *CVPR*, 2019. 3

[50] Shaoshuai Shi, Zhe Wang, Jianping Shi, Xiaogang Wang, and Hongsheng Li. From points to parts: 3d object detection from point cloud with part-aware and part-aggregation network. *TPAMI*, 2020. 3

[51] Vishwanath A Sindagi, Yin Zhou, and Oncel Tuzel. Mvxnet: Multimodal voxelnet for 3d object detection. In *ICRA*, 2019. 1, 3, 8

[52] Sourabh Vora, Alex H Lang, Bassam Helou, and Oscar Beijbom. Pointpainting: Sequential fusion for 3d object detection. In *CVPR*, 2020. 1, 3, 6, 7, 8

[53] Chunwei Wang, Chao Ma, Ming Zhu, and Xiaokang Yang. Pointaugmenting: Cross-modal augmentation for 3d object detection. In *CVPR*, 2021. 1, 3, 6, 8

[54] Dequan Wang, Coline Devin, Qi-Zhi Cai, Philipp Krähenbühl, and Trevor Darrell. Monocular plan view networks for autonomous driving. In *IROS*, 2019. 1

[55] Haiyang Wang, Lihe Ding, Shaocong Dong, Shaoshuai Shi, Aoxue Li, Jianan Li, Zhenguo Li, and Liwei Wang. CA-Group3d: Class-aware grouping for 3d object detection on point clouds. In *NeurIPS*, 2022. 3

[56] Haiyang Wang, Chen Shi, Shaoshuai Shi, Meng Lei, Sen Wang, Di He, Bernt Schiele, and Liwei Wang. Dsvt: Dynamic sparse voxel transformer with rotated sets. *CVPR*, 2023. 2, 3, 4, 7

[57] Haiyang Wang, Shaoshuai Shi, Ze Yang, Rongyao Fang, Qi Qian, Hongsheng Li, Bernt Schiele, and Liwei Wang. Rbgnet: Ray-based grouping for 3d object detection. In *CVPR*, 2022. 3

[58] Luequan Wang, Hongbin Xu, and Wenxiong Kang. Mvcontrast: Unsupervised pretraining for multi-view 3d object recognition. *Machine Intelligence Research*, pages 1–12, 2023. 3

[59] Tai Wang, Xinge Zhu, Jiangmiao Pang, and Dahua Lin. Fcos3d: Fully convolutional one-stage monocular 3d object detection. In *ICCV*, 2021. 3

[60] Yue Wang, Vitor Campagnolo Guizilini, Tianyuan Zhang, Yilun Wang, Hang Zhao, and Justin Solomon. Detr3d: 3d object detection from multi-view images via 3d-to-2d queries. In *CoRL*, 2022. 3, 8

[61] Hai Wu, Chenglu Wen, Shaoshuai Shi, Xin Li, and Cheng Wang. Virtual sparse convolution for multimodal 3d object detection. In *CVPR*, 2023. 3

[62] Enze Xie, Zhiding Yu, Daquan Zhou, Jonah Philion, Anima Anandkumar, Sanja Fidler, Ping Luo, and Jose M Alvarez. M^2bev: Multi-camera joint 3d detection and segmentation with unified birds-eye view representation. *arXiv preprint arXiv:2204.05088*, 2022. 7

[63] Shaoqing Xu, Dingfu Zhou, Jin Fang, Junbo Yin, Zhou Bin, and Liangjun Zhang. Fusionpainting: Multimodal fusion with adaptive attention for 3d object detection. In *ITSC*, 2021. 6

[64] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 2018. 3, 6

[65] Chenyu Yang, Yuntao Chen, Hao Tian, Chenxin Tao, Xizhou Zhu, Zhaoxiang Zhang, Gao Huang, Hongyang Li, Yu Qiao, Lewei Lu, et al. Bevformer v2: Adapting modern image backbones to bird's-eye-view recognition via perspective supervision. *CVPR*, 2023. 6

[66] Hao Yang, Chen Shi, Yihong Chen, and Liwei Wang. Boosting 3d object detection via object-focused image fusion. *arXiv preprint arXiv:2207.10589*, 2022. 1, 2, 3, 5

[67] Zeyu Yang, Jiaqi Chen, Zhenwei Miao, Wei Li, Xiatian Zhu, and Li Zhang. Deepinteraction: 3d object detection via modality interaction. In *NeurIPS*, 2022. 6

[68] Zetong Yang, Li Jiang, Yanan Sun, Bernt Schiele, and Jiaya Jia. A unified query-based paradigm for point cloud understanding. In *CVPR*, 2022. 3

[69] Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 3dssd: Point-based 3d single stage object detector. In *CVPR*, 2020. 3

[70] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *CVPR*, 2021. 1, 3, 6, 7

[71] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Multimodal virtual point 3d detection. In *NeurIPS*, 2021. 5, 6, 7

[72] Jin Hyeok Yoo, Yecheol Kim, Jisong Kim, and Jun Won Choi. 3d-cvf: Generating joint camera and lidar features using cross-view spatial feature fusion for 3d object detection. In *ECCV*, 2020. 1, 3

[73] Kaicheng Yu, Tang Tao, Hongwei Xie, Zhiwei Lin, Zhongwei Wu, Zhongyu Xia, Tingting Liang, Haiyang Sun, Jiong Deng, Dayang Hao, et al. Benchmarking the robustness of lidar-camera fusion for 3d object detection. *arXiv preprint arXiv:2205.14951*, 2022. 8

[74] Brady Zhou and Philipp Krähenbühl. Cross-view transformers for real-time map-view semantic segmentation. In *CVPR*, 2022. 7

[75] Yin Zhou, Pei Sun, Yu Zhang, Dragomir Anguelov, Jiyang Gao, Tom Ouyang, James Guo, Jiquan Ngiam, and Vijay Vasudevan. End-to-end multi-view fusion for 3d object detection in lidar point clouds. In *CoRL*, 2020. 2, 4, 6

[76] Xinge Zhu, Hui Zhou, Tai Wang, Fangzhou Hong, Yuexin Ma, Wei Li, Hongsheng Li, and Dahua Lin. Cylindrical and asymmetrical 3d convolution networks for lidar segmentation. In *CVPR*, 2021. 3

[77] Xizhou Zhu, Jinguo Zhu, Hao Li, Xiaoshi Wu, Hongsheng Li, Xiaohua Wang, and Jifeng Dai. Uni-perceiver: Pretraining unified architecture for generic perception for zero-shot and few-shot tasks. In *CVPR*, 2022. 2