IS-FUSION: Instance-Scene Collaborative Fusion for Multimodal 3D Object Detection

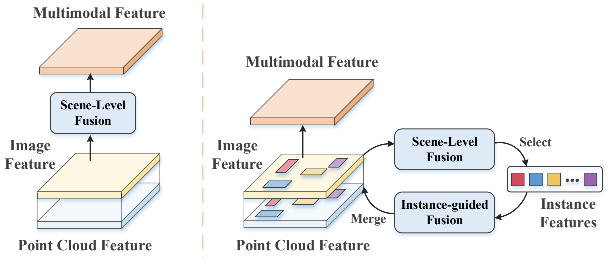
Junbo Yin[1], Jianbing Shen[2], Runnan Chen[3], Wei Li[4*], Ruigang Yang[4], Pascal Frossard[5], Wenguan Wang[6*]
[1]School of Computer Science and Technology, Beijing Institute of Technology [2]SKL-IOTSC, CIS, University of Macau
[3]The University of Hong Kong [4]Inceptio [5]École Polytechnique Fédérale de Lausanne (EPFL) [6]ReLER, CCAI, Zhejiang University
{yinjunbocn,wenguanwang.ai}@gmail.com

Abstract

Bird's eye view (BEV) representation has emerged as a dominant solution for describing 3D space in autonomous driving scenarios. However, objects in the BEV representation typically exhibit small sizes, and the associated point cloud context is inherently sparse, which leads to great challenges for reliable 3D perception. In this paper, we propose **IS-FUSION**, *an innovative multimodal* ***fusion*** *framework that jointly captures the* ***I****nstance- and* ***S****cene-level contextual information.* IS-FUSION *essentially differs from existing approaches that only focus on the BEV scene-level fusion by explicitly incorporating instance-level multimodal information, thus facilitating the instance-centric tasks like 3D object detection. It comprises a Hierarchical Scene Fusion (HSF) module and an Instance-Guided Fusion (IGF) module. HSF applies Point-to-Grid and Grid-to-Region transformers to capture the multimodal scene context at different granularities. IGF mines instance candidates, explores their relationships, and aggregates the local multimodal context for each instance. These instances then serve as guidance to enhance the scene feature and yield an instance-aware BEV representation. On the challenging nuScenes benchmark,* IS-FUSION *outperforms all the published multimodal works to date. Code is available at:* *https://github.com/yinjunbo/IS-Fusion*.

1. Introduction

3D object detection [23, 49, 53, 82, 88, 93] is a critical task in various applications such as autonomous driving and robotics. Over the past few years, tremendous progress has been achieved in point cloud-based 3D object detection, due to the effective 3D neural network models [5, 17, 18, 19, 55, 87]. While point clouds, typically captured by depth-aware sensors such as LiDAR, provide valuable geometric information about the 3D space, they often lack detailed texture descriptions and are sparsely distributed over long distances, *e.g.*, beyond 100 meters in outdoor scenarios like nuScenes [2]. To tackle these limitations, a recent trend is to perform multimodal 3D object detection [33, 34, 40, 46, 77] by fusing information from both point clouds and synchronized multi-view images. The image modality provides detailed texture and dense semantic information [20, 70], which complements the sparse point cloud and thus enhances the 3D perception capacity.



(a) Scene-only Fusion (b) Instance-Scene Collaborative Fusion

Figure 1. **Motivation of IS-FUSION.** (a) Previous approaches typically focus on fusion at the entire scene level during multimodal encoding. (b) In contrast, IS-FUSION places additional emphasis on the fusion at the instance level and explores the instance-to-scene collaboration to enhance the overall representation.

To handle heterogeneous data from different modalities, existing approaches [40, 42, 46] typically pre-define a unified space that is compatible with both modalities (*i.e.*, the bird's eye view (BEV) in the ego-vehicle coordinate system), and then perform feature alignment and fusion on this shared space. BEV representation simplifies the complex 3D space into a 2D plane, making it easier to understand the scene. However, performing fusion from the entire BEV scene level ignores the inherent difference between the foreground instances and background regions, which may undermine the performance. For example, object instances represented in the BEV often exhibit smaller sizes compared to those observed in natural images. Additionally, the number of BEV grid cells occupied by foreground instances is significantly lower than those occupied by back-

[*]Corresponding author.

ground ones, leading to a severe imbalance between foreground and background samples. As a result, the above approaches struggle to capture local context around the object instances, or largely rely on additional networks in the decoding stage to iteratively refine the detections [1, 77]. While a few methods [3, 71] aim to perform object-level encoding, they ignore the potential collaboration between the scene and instance features. For example, a false negative object in a scene can be potentially rectified by enhancing its feature through interactions with the instances sharing similar semantic information. Therefore, it remains an open question *how to simultaneously formulate the instance-level and scene-level context, as well as elegantly integrate them by leveraging multimodal fusion.*

In this work, we present a new multimodal detection framework, IS-FUSION, to tackle the above challenge. As shown in Fig. 1, IS-FUSION explores both the **I**nstance-level and **S**cene-level **Fusion**, as well as encourages the interaction between the instance and scene features to strengthen the overall representation. It consists of two crucial components: the Hierarchical Scene Fusion (HSF) module and the Instance-Guided Fusion (IGF) module. HSF aims to capture scene features at various granularities by utilizing Point-to-Grid and Grid-to-Region transformers. This also enables the generation of high-quality instance-level features that are vital for IGF. In IGF, the foreground instance candidates are determined by the heatmap scores of the scene feature; meanwhile, an inter-instance self-attention is employed to capture the instance relationships. These instances then aggregate essential semantic information from the multimodal context through deformable attention. Furthermore, we incorporate an Instance-to-Scene transformer attention to enforce the local instance features to collaborate with the global scene feature. This yields an enhanced BEV representation that is better suited for instance-aware tasks like 3D object detection.

In summary, IS-FUSION provides a new insight into existing multimodal 3D detection approaches that focus on scene-level fusion. By incorporating HSF and IGF, it explicitly promotes collaboration between scene- and instance-level features, thereby ensuring comprehensive representation and yielding improved detection results. Extensive experiments on the competitive nuScenes [2] dataset demonstrate that IS-FUSION attains the best performance among all the published 3D object detection works. For example, it achieves 72.8% mAP on the nuScenes validation set, outperforming prior art BEVFusion [46] by 4.3% mAP. It also surpasses concurrent works like CMT [74] and SparseFusion [71] by 2.5% and 1.8% mAP, respectively.

2. Related Work

LiDAR-based 3D Object Detection. LiDAR sensors are essential for advanced autonomous driving due to their capacity of perceiving objects in 3D space even in adverse illumination and weather environments, where they are usually more reliable than camera sensors [16, 21, 30, 37, 44, 67, 68]. Current LiDAR-based detection approaches can be broadly classified into three categories according to the various encoding formats of point cloud: point-based [15, 52, 58, 59, 78, 83, 89], voxel-based [22, 29, 32, 69, 81, 90] and point-voxel fusion networks [10, 25, 50, 57, 79]. Shi *et al.*[58] propose an early work for point-wise 3D detection by extending PointNet [54, 55] backbone with a two-stage proposal refinement network. Due to the huge computation overhead in large-scale scenes with more than 100k points, point downsampling operations [78] have to be applied. A more popular solution is to use the voxel-based representation, where the point clouds are quantified by regular grids such that standard convolutional networks can be directly applied. VoxelNet [90] is the seminal work that exploits the 3D convolutional network that is later optimized in [75] with sparse convolution. In addition, there are also some works like [57] exploring joint point-voxel representation by enhancing the region proposals with the raw points information, while they often require multiple stages to refine the 3D proposals.

These LiDAR-only 3D detectors usually operate with sparse and noisy context provided by point cloud data. However, in challenging scenarios where objects have low reflectivity, small sizes or are heavily occluded, relying solely on point cloud data may lead to inaccurate detection [4, 43, 48]. Therefore, our focus is to explore the multimodal context by incorporating the merits of both geometry-aware point clouds and semantic-rich images to guarantee advanced 3D object detection capability.

LiDAR-camera Fusion for 3D Object Detection. Multimodal 3D object detection [6, 38, 39, 72] has recently received considerable attention. It also has been proven that multimodal learning can yield a more accurate latent space representation [26] compared to the unimodal learning. Multimodal fusion approaches for 3D object detection basically comprise early fusion [12, 63, 63, 73, 85], middle fusion [34, 35, 36, 40, 46, 51, 77, 86] and late fusion [1, 6, 27, 31, 38, 76], which are categorized based on the stages at which data fusion occurs.

The works in [63, 64] are pioneering efforts of early fusion that enhance input points with corresponding image pixel features. Later, Chen *et al.* [12] propose to fuse point cloud and image features at the voxel level and aggregate information from multiple sampling points. However, the early fusion approaches are more sensitive to potential calibration errors. Late fusion approaches like [6, 27] typically fuse multimodal information at the region proposal level, where the region proposals are usually generated separately by modality-specific encoders. These approaches may result in limited interactions between modalities dur-

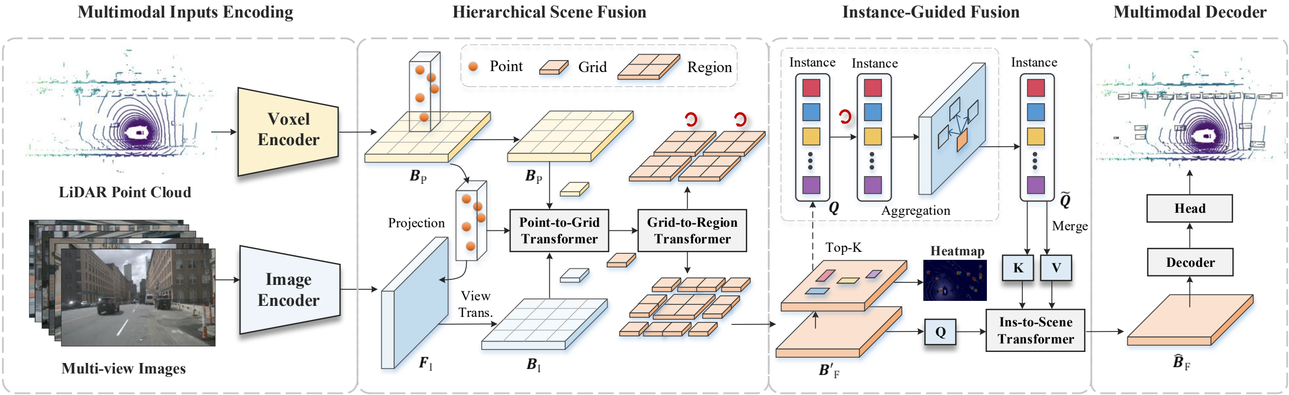


Figure 2. **Overview of our IS-Fusion framework.** Multimodal inputs including a point cloud and multi-view images are first processed by modality-specific encoders to obtain initial features. Then, the HSF module, equipped with Point-to-Grid and Grid-to-Region transformers, utilizes these features to generate a scene-level feature with hierarchical context. Furthermore, the IGF module identifies the top-K salient instances and aggregates the multimodal context for each instance. Finally, these instances are employed by the Instance-to-Scene transformer to propagate valuable information to the scene, producing the final BEV representation with improved instance awareness.

ing proposal generation [71], leading to suboptimal detection performance. By contrast, middle fusion has become increasingly popular recently as it encourages multimodal feature interaction at various representation stages, which is more robust to calibration error. Liu *et al.* [46] and Liang *et al.* [40] propose to align point cloud with multi-view image features on a unified BEV plane to simplify the fusion process. Yang *et al.* [77] further suggest fusion from both BEV and image perspective-view spaces to preserve modality-specific information.

Unlike these approaches [46, 74, 77], which primarily integrate point cloud and image features at a global scene level, our IS-Fusion investigates fusion from both local instance level and global scene level. This permits the advantages of 'hybrid fusion', which also marks an improvement over some concurrent works [3, 71] that focus only on the instances and ignore the collaboration with the scene. IS-Fusion smartly exchanges information between instances and the scene, and thus facilitates the instance-centric tasks such as 3D object detection, as shown later.

3. Methodology

We first introduce the general overview of the proposed IS-Fusion in Sec. 3.1. Next, we delve into the details of the HSF module in Sec. 3.2. After that, in Sec. 3.3, we elaborate on the crucial design steps of the IGF module.

3.1. Overall Framework

As illustrated in Fig. 2, each scene is represented by a LiDAR point cloud P, along with synchronized RGB images $I=\{I_1, I_2, \ldots, I_N\}$ captured by N cameras that are well-calibrated with the LiDAR sensor. Our goal is to devise a detection model capable of producing precise 3D bounding boxes $\boldsymbol{Y}$, given multimodal inputs (P, I). Formally, the proposed IS-Fusion model is defined by:

$$\boldsymbol{Y} = f_{\text{dec}}(f_{\text{enc}}(f_{\text{point}}(P), f_{\text{img}}(I))), \tag{1}$$

where $f_{\text{point}}(\cdot)$ and $f_{\text{img}}(\cdot)$ serve as the input encoding modules, $f_{\text{enc}}(\cdot)$ denotes the multimodal encoder (formed by HSF and IGF) and $f_{\text{dec}}(\cdot)$ is the decoder.

Multimodal Input Encoding. To handle inputs from heterogeneous modalities, we first utilize modality-specific encoders to get their respective initial representation, *i.e.*, $\boldsymbol{B}_{\text{P}} = f_{\text{point}}(P)$ and $\boldsymbol{F}_{\text{I}} = f_{\text{img}}(I)$. Following [46, 77], we instantiate $f_{\text{point}}(\cdot)$ with VoxelNet [90], and $f_{\text{img}}(\cdot)$ by Swin-Transformer [45]. This yields the point cloud BEV feature $\boldsymbol{B}_{\text{P}}$ and the image Perspective-View (PV) features $\boldsymbol{F}_{\text{I}}$. In particular, $\boldsymbol{B}_{\text{P}} \in \mathbb{R}^{W\times H\times C}$ is obtained by compressing the height dimension of the 3D voxel feature as in [90], where W and H are the numbers of BEV grid cells along the x and y axes, and C denotes the channel dimension.

Multimodal Encoder. The multimodal encoder $f_{\text{enc}}(\cdot)$ conducts cross-modality feature fusion between $\boldsymbol{B}_{\text{P}}$ and $\boldsymbol{F}_{\text{I}}$ to yield a fused BEV feature $\hat{\boldsymbol{B}}_{\text{F}} \in \mathbb{R}^{W\times H\times C}$. In contrast to previous multimodal encoders that only focus on fusion at the entire scene level [46, 77], we develop both instance-level and scene-level representations. To this end, we design $f_{\text{enc}}(\cdot)$ using two modules, namely, HSF module $f_{\text{HSF}}(\cdot)$ and IGF module $f_{\text{IGF}}(\cdot)$:

$$\hat{\boldsymbol{B}}_{\text{F}} = f_{\text{enc}}(\boldsymbol{B}_{\text{P}}, \boldsymbol{F}_{\text{I}}) = f_{\text{IGF}}(f_{\text{HSF}}(\boldsymbol{B}_{\text{P}}, \boldsymbol{F}_{\text{I}})), \tag{2}$$

where $f_{\text{HSF}}(\cdot)$ generates multi-granularity scene feature, while $f_{\text{IGF}}(\cdot)$ further integrates crucial information about foreground instances. We will elaborate on $f_{\text{HSF}}(\cdot)$ and $f_{\text{IGF}}(\cdot)$ in Sec. 3.2 and Sec. 3.3, respectively.

Multimodal Decoder. The multimodal decoder aims to yield the final 3D detections $\boldsymbol{Y}$ based on the BEV representation $\hat{\boldsymbol{B}}_{\text{F}}$, given by $\boldsymbol{Y} = f_{\text{dec}}(\hat{\boldsymbol{B}}_{\text{F}})$. In our work, $f_{\text{dec}}(\cdot)$ is

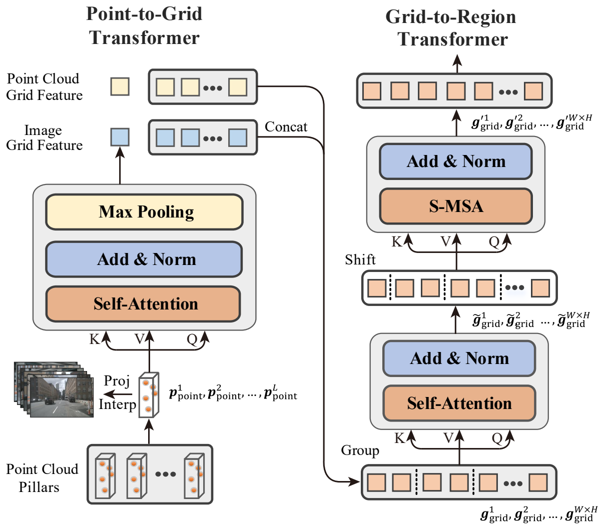

Point-to-Grid Transformer
Point Cloud Grid Feature
Image Grid Feature
Concat
Max Pooling
Add & Norm
Self-Attention
K
V
Q
Proj
Interp
$\boldsymbol{p}^1_{\text{point}}, \boldsymbol{p}^2_{\text{point}}, \cdots, \boldsymbol{p}^L_{\text{point}}$
Point Cloud Pillars
Grid-to-Region Transformer
$\boldsymbol{g}'^1_{\text{grid}}, \boldsymbol{g}'^2_{\text{grid}}, \cdots, \boldsymbol{g}'^{W\times H}_{\text{grid}}$
Add & Norm
S-MSA
K
V
Q
Shift
$\tilde{\boldsymbol{g}}^1_{\text{grid}}, \tilde{\boldsymbol{g}}^2_{\text{grid}} \cdots, \tilde{\boldsymbol{g}}^{W\times H}_{\text{grid}}$
Add & Norm
Self-Attention
K
V
Q
Group
$\boldsymbol{g}^1_{\text{grid}}, \boldsymbol{g}^2_{\text{grid}}, \cdots, \boldsymbol{g}^{W\times H}_{\text{grid}}$

Figure 3. **Illustration of HSF module.** It first aggregates the point-level features into the grid-level features with the Point-to-Grid transformer, and then explores the inter-grid and inter-region feature interaction through the Grid-to-Region transformer.

built upon a transformer architecture [80], which contains several attention layers and a feed-forward-network serving as the detection head. During training, the Hungarian algorithm [28] is applied for matching the predicted and ground-truth bounding boxes. Meanwhile, Focal loss [41] and $L1$ loss are adopted for the classification and 3D bounding box regression, respectively.

3.2. Hierarchical Scene Fusion

Given the point cloud BEV feature $\boldsymbol{B}_{\text{P}}$ and the image PV feature $\boldsymbol{F}_{\text{I}}$, we suggest a Hierarchical Scene Fusion (HSF) module $f_{\text{HSF}}(\cdot)$ to integrate $\boldsymbol{B}_{\text{P}}$ and $\boldsymbol{F}_{\text{I}}$ and obtain the fused scene representation $\boldsymbol{B}_{\text{F}} \in \mathbb{R}^{W\times H\times C}$. To be specific, $f_{\text{HSF}}(\cdot)$ comprises a Point-to-Grid transformer $f_{\text{P2G}}(\cdot)$ and Grid-to-Region transformer $f_{\text{G2R}}(\cdot)$, given by:

$$\boldsymbol{B}_{\text{F}} = f_{\text{HSF}}(\boldsymbol{F}_{\text{I}}, \boldsymbol{B}_{\text{P}}) = f_{\text{G2R}}(f_{\text{P2G}}(\boldsymbol{F}_{\text{I}}), \boldsymbol{B}_{\text{P}}). \tag{3}$$

Here, $f_{\text{P2G}}(\cdot)$ considers the inter-point/pixel correlations in each BEV grid, while $f_{\text{G2R}}(\cdot)$ further mines the inter-grid and inter-region multimodal scene context. The intuition is that different feature granularities capture scene context at different levels. For example, at the point level, each element provides detailed information about specific components of an object. In contrast, features at the grid/region level are capable of capturing the broader scene structure and the distribution of objects. HSF fully leverages various representation granularities, as illustrated in Fig. 3.

Point-to-Grid Transformer. Let us denote $G = \{g^1_{\text{grid}}, g^2_{\text{grid}}, \ldots, g^{W\times H}_{\text{grid}}\}$ as the BEV grid cells obtained by discretizing the point cloud scene P into pillars following [29]. Each grid cell $g_{\text{grid}} \in G$ is a pillar containing L points $\{p^1_{\text{point}}, p^2_{\text{point}}, \ldots, p^L_{\text{point}}\}$. The Point-to-Grid transformer assigns each point with its corresponding image feature and aggregates them into a BEV grid-wise feature.

Specifically, we project the L points within a pillar $[p^1_{\text{point}}, p^2_{\text{point}}, \ldots, p^L_{\text{point}}] \in \mathbb{R}^{L\times 3}$ onto the image feature map $\boldsymbol{F}_{\text{I}}$ and retrieve their pixel-level features:

$$\begin{aligned} &[u^1, u^2, \cdots, u^L] = f_{\text{proj}}([p^1_{\text{point}}, p^2_{\text{point}}, \ldots, p^L_{\text{point}}]), \\ &[\boldsymbol{p}^1_{\text{point}}, \boldsymbol{p}^2_{\text{point}}, \ldots, \boldsymbol{p}^L_{\text{point}}] = f_{\text{interp}}(\boldsymbol{F}_{\text{I}}, [u^1, u^2, \cdots, u^L]), \end{aligned} \tag{4}$$

where $f_{\text{proj}}(\cdot)$ indicates the projection process from point cloud to multi-view images that yields 2D coordinates $[u^1, u^2, \cdots, u^L]$ on the image plane, and $f_{\text{interp}}(\cdot)$ is the bilinear interpolation function computing features at non-integer coordinates. In this way, we get the point-wise features $[\boldsymbol{p}^1_{\text{point}}, \boldsymbol{p}^2_{\text{point}}, \ldots, \boldsymbol{p}^L_{\text{point}}] \in \mathbb{R}^{L\times C}$.

To handle the potential calibration noise between LiDAR and cameras, our Point-to-Grid transformer compares all the points within a pillar. This enables each point to consider a larger receptive field and implicitly rectifies noisy points. Afterwards, we merge the point-wise information with a max pooling operation $f_{\max}(\cdot)$:

$$\boldsymbol{g}_{\text{grid}} = f_{\max}(f_{\text{MSA}}([\boldsymbol{p}^1_{\text{point}}, \boldsymbol{p}^2_{\text{point}}, \ldots, \boldsymbol{p}^L_{\text{point}}])) \in \mathbb{R}^{1\times C}, \tag{5}$$

where $f_{\text{MSA}}(\cdot)$ is the multi-head self-attention [62], and $\boldsymbol{g}_{\text{grid}}$ is a grid-wise feature that will be assigned to the image BEV feature $\boldsymbol{B}_{\text{I}} \in \mathbb{R}^{W\times H\times C}$. Then, we compute the multimodal BEV feature $\boldsymbol{B}_{\text{F}}$ by combing $\boldsymbol{B}_{\text{I}}$ with the point cloud BEV feature $\boldsymbol{B}_{\text{P}} \in \mathbb{R}^{W\times H\times C}$:

$$\boldsymbol{B}_{\text{F}} = f_{\text{conv}}([\boldsymbol{B}_{\text{I}}, \boldsymbol{B}_{\text{P}}]) \in \mathbb{R}^{W\times H\times C}, \tag{6}$$

where $[\cdot,\cdot]$ denotes the concatenation, and $f_{\text{conv}}(\cdot)$ is implemented by a 3×3 convolutional layer.

Grid-to-Region Transformer. In addition to the Point-to-Grid transformer that models the inter-point dependencies, we further explore the inter-grid and inter-region relationships via the Grid-to-Region transformer to capture the global scene context. This can be denoted as $\boldsymbol{B}'_{\text{F}} = f_{\text{G2R}}(\boldsymbol{B}_{\text{F}})$, where $\boldsymbol{B}'_{\text{F}}$ is the enhanced BEV feature.

Intuitively, $f_{\text{G2R}}(\cdot)$ can be achieved by applying global self-attention to all the grid-wise features $\{\boldsymbol{g}^1_{\text{grid}}, \boldsymbol{g}^2_{\text{grid}}, \cdots, \boldsymbol{g}^{W\times H}_{\text{grid}}\} \in \boldsymbol{B}_{\text{F}}$. However, this can be computationally expensive due to the large number of grid cells. Hence, we choose to group these grid features into different regions following [14]. Each region is a subset described by $M\times M$ grid cells $\{g^1_{\text{grid}}, g^2_{\text{grid}}, \cdots, g^{M^2}_{\text{grid}}\}$. Next, we view each region as a whole and exchange information between the grids in a region through inter-grid attention. More concretely, this is realized by the multi-head attention $f_{\text{MSA}}(\cdot)$ operating on a set of grid-wise features $[\boldsymbol{g}^1_{\text{grid}}, \boldsymbol{g}^2_{\text{grid}}, \cdots, \boldsymbol{g}^{M^2}_{\text{grid}}] \in \mathbb{R}^{M^2\times C}$:

$$[\tilde{\boldsymbol{g}}^1_{\text{grid}}, \tilde{\boldsymbol{g}}^2_{\text{grid}}, \cdots, \tilde{\boldsymbol{g}}^{M^2}_{\text{grid}}] = f_{\text{MSA}}([\boldsymbol{g}^1_{\text{grid}}, \boldsymbol{g}^2_{\text{grid}}, \cdots, \boldsymbol{g}^{M^2}_{\text{grid}}]), \tag{7}$$

where $[\tilde{\boldsymbol{g}}_{\text{grid}}^{1}, \tilde{\boldsymbol{g}}_{\text{grid}}^{2}, \cdots, \tilde{\boldsymbol{g}}_{\text{grid}}^{M^2}]$ are the attentive grid cells.

Then, we capture the interactions between different regions with inter-region attention. To this end, we shift each region by $(M/2, M/2)$ grid cells and conduct self-attention on each shifted region containing $M \times M$ grid-wise features (using padding if necessary). This is given by:

$$[\boldsymbol{g}'^{\triangle_1}_{\text{grid}}, \boldsymbol{g}'^{\triangle_2}_{\text{grid}}, \cdots, \boldsymbol{g}'^{\triangle_{M^2}}_{\text{grid}}] = f_{\text{S-MSA}}([\tilde{\boldsymbol{g}}^{\triangle_1}_{\text{grid}}, \tilde{\boldsymbol{g}}^{\triangle_2}_{\text{grid}}, \cdots, \tilde{\boldsymbol{g}}^{\triangle_{M^2}}_{\text{grid}}]), \tag{8}$$

where $f_{\text{S-MSA}}(\cdot)$ indicates the shifted-window self-attention in [45], and $\{\triangle_1, \triangle_2, \cdots, \triangle_{M^2}\}$ represent the new grid indices after the shift. This allows each grid to interact with the grids coming from various regions before the shift, thus capturing long-range dependencies. Then, we rearrange all the attentive grid features $\{\boldsymbol{g}'^{1}_{\text{grid}}, \boldsymbol{g}'^{2}_{\text{grid}}, \cdots, \boldsymbol{g}'^{W \times H}_{\text{grid}}\}$ to get the enriched BEV feature map $\boldsymbol{B}'_{\text{F}} \in \mathbb{R}^{W \times H \times C}$.

By exploiting the hierarchical representation, HSF enables the propagation of information from individual points to different BEV regions. This facilitates the integration of both local and global multimodal scene contexts.

3.3. Instance-Guided Fusion

The basic idea of IGF is to mine the multimodal context around each object instance (*e.g.*, the lanes beside the vehicles), meanwhile integrating essential instance-level information into the scene feature. For example, if an object is incorrectly categorized as part of the background in the scene feature, we can rectify this by comparing it with all the relevant instances. Formally, given the scene feature $\boldsymbol{B}'_{\text{F}}$ produced by HSF, $f_{\text{IGF}}(\cdot)$ in Eq. (2) is formulated as:

$$\textit{instance selection:} \quad \boldsymbol{Q} = f_{\text{sel}}(\boldsymbol{B}'_{\text{F}}) \in \mathbb{R}^{K \times C}, \tag{9}$$

$$\textit{context aggregation:} \quad \tilde{\boldsymbol{Q}} = f_{\text{agg}}(\boldsymbol{Q}, \boldsymbol{B}'_{\text{F}}) \in \mathbb{R}^{K \times C}, \tag{10}$$

$$\textit{instance-to-scene:} \quad \hat{\boldsymbol{B}}_{\text{F}} = f_{\text{I2S}}(\boldsymbol{B}'_{\text{F}}, \tilde{\boldsymbol{Q}}) \in \mathbb{R}^{W \times H \times C}, \tag{11}$$

where $f_{\text{sel}}(\cdot)$ selects the top-K salient instance features $\boldsymbol{Q} = [\boldsymbol{q}^1_{\text{ins}}, \boldsymbol{q}^2_{\text{ins}}, \ldots, \boldsymbol{q}^K_{\text{ins}}]$, $f_{\text{agg}}(\cdot)$ aggregates the multimodal context for each instance and $f_{\text{I2S}}(\cdot)$ merges the augmented instance features $\tilde{\boldsymbol{Q}} = [\tilde{\boldsymbol{q}}^1_{\text{ins}}, \tilde{\boldsymbol{q}}^2_{\text{ins}}, \ldots, \tilde{\boldsymbol{q}}^K_{\text{ins}}]$ to the BEV scene feature $\boldsymbol{B}'_{\text{F}}$. We present the overall pipeline of IGF in Fig. 4, and explain $f_{\text{sel}}(\cdot)$, $f_{\text{agg}}(\cdot)$ and $f_{\text{I2S}}(\cdot)$ as follows.

Instance Candidates Selection. To efficiently generate instance features, we implement $f_{\text{sel}}(\cdot)$ following [84] that applies a keypoint detection head on the scene feature $\boldsymbol{B}'_{\text{F}}$ to predict the centerness of instances. During training, a 2D Gaussian distribution is defined for each instance as the target, and the peak location is determined by the BEV projection of the 3D center of the ground truth. Focal loss is employed to optimize this prediction head. During inference, we keep the top-K object with the highest centerness scores to represent the corresponding instances. Meanwhile, an additional linear layer is employed to embed each instance, yielding a set of instance features $\{\boldsymbol{q}^1_{\text{ins}}, \boldsymbol{q}^2_{\text{ins}}, \ldots, \boldsymbol{q}^K_{\text{ins}}\}$.

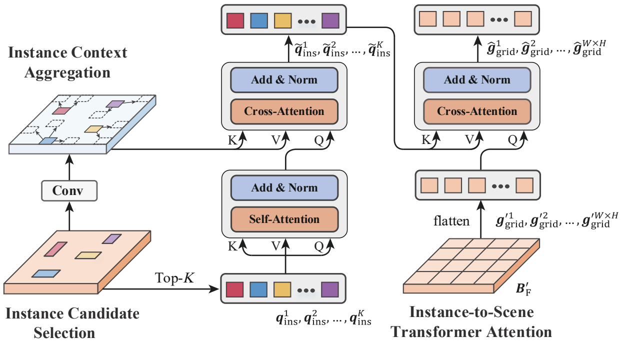

Instance Context Aggregation
Conv
Instance Candidate Selection
Top-K
$\boldsymbol{q}^1_{\text{ins}}, \boldsymbol{q}^2_{\text{ins}}, \ldots, \boldsymbol{q}^K_{\text{ins}}$
K
V
Q
Add & Norm
Self-Attention
Add & Norm
Cross-Attention
$\tilde{\boldsymbol{q}}^1_{\text{ins}}, \tilde{\boldsymbol{q}}^2_{\text{ins}}, \ldots, \tilde{\boldsymbol{q}}^K_{\text{ins}}$
$\hat{\boldsymbol{g}}^1_{\text{grid}}, \hat{\boldsymbol{g}}^2_{\text{grid}}, \ldots, \hat{\boldsymbol{g}}^{W \times H}_{\text{grid}}$
flatten
$\boldsymbol{g}'^1_{\text{grid}}, \boldsymbol{g}'^2_{\text{grid}}, \ldots, \boldsymbol{g}'^{W \times H}_{\text{grid}}$
$\boldsymbol{B}'_{\text{F}}$
Instance-to-Scene Transformer Attention

Figure 4. **Illustration of IGF module.** Instance candidates are first initialized based on the BVE heatmap. Then, we perform reasoning on these instances, meanwhile aggregating rich semantic context from the image features. Finally, these instances transfer contextual information to the BEV scene feature through an Instance-to-Scene transformer attention mechanism.

Instance Context Aggregation. We design $f_{\text{agg}}(\cdot)$ to count for both the instance-to-instance and instance-to-context interactions. In typical driving scenarios, it is often observed that pedestrians tend to appear in groups or clusters, and vehicles commonly co-exist along the roadside. Thus, it is crucial to investigate the correlations between instances. To this end, we employ self-attention $f_{\text{MSA}}(\cdot)$ on the selected instances $[\boldsymbol{q}^1_{\text{ins}}, \boldsymbol{q}^2_{\text{ins}}, \ldots, \boldsymbol{q}^K_{\text{ins}}] \in \mathbb{R}^{K \times C}$:

$$[\boldsymbol{q}'^1_{\text{ins}}, \boldsymbol{q}'^2_{\text{ins}}, \ldots, \boldsymbol{q}'^K_{\text{ins}}] = f_{\text{MSA}}([\boldsymbol{q}^1_{\text{ins}}, \boldsymbol{q}^2_{\text{ins}}, \ldots, \boldsymbol{q}^K_{\text{ins}}]), \tag{12}$$

Furthermore, we aim to mine the semantic context for each instance. This is achieved by comparing each instance $\boldsymbol{q}'_{\text{ins}}$ and the corresponding part in the multimodal feature $\boldsymbol{B}'_{\text{F}}$. Specifically, only on a small set of neighbor locations (*e.g.*, D grid cells) around $\boldsymbol{q}'_{\text{ins}}$ are considered to save computation costs, following the deformable attention in [92]:

$$\tilde{\boldsymbol{q}}_{\text{ins}} = f_{\text{DeformAtt}}(\boldsymbol{q}'_{\text{ins}}, f_{\text{conv}}(\boldsymbol{B}'_{\text{F}})), \tag{13}$$

where $f_{\text{conv}}(\cdot)$ is a 3×3 convolution operation to align the feature space between $\boldsymbol{q}'_{\text{ins}}$ and $\boldsymbol{B}'_{\text{F}}$, and $\tilde{\boldsymbol{q}}_{\text{ins}} \in \{\tilde{\boldsymbol{q}}^1_{\text{ins}}, \tilde{\boldsymbol{q}}^2_{\text{ins}}, \ldots, \tilde{\boldsymbol{q}}^K_{\text{ins}}\}$ is the enriched instance features.

Instance-to-Scene Transformer. Finally, $f_{\text{I2S}}(\cdot)$ enables each BEV grid feature to acquire valuable information from potentially relevant instances. To this end, we build $f_{\text{I2S}}(\cdot)$ with a transformer cross-attention mechanism. Specifically, after flattening $\boldsymbol{B}'_{\text{F}} \in \mathbb{R}^{W \times H \times C}$ into a set of grid features $\{\boldsymbol{g}'^1_{\text{grid}}, \boldsymbol{g}'^2_{\text{grid}}, \ldots, \boldsymbol{g}'^{W \times H}_{\text{grid}}\}$, we employ each grid $\boldsymbol{g}'_{\text{grid}}$ as a query to attend to the instance-level features $[\tilde{\boldsymbol{q}}^1_{\text{ins}}, \tilde{\boldsymbol{q}}^2_{\text{ins}}, \ldots, \tilde{\boldsymbol{q}}^K_{\text{ins}}] \in \mathbb{R}^{K \times C}$:

$$\hat{\boldsymbol{g}}_{\text{grid}} = f_{\text{MCA}}(\boldsymbol{g}'_{\text{grid}}, [\tilde{\boldsymbol{q}}^1_{\text{ins}}, \tilde{\boldsymbol{q}}^1_{\text{ins}}, \ldots, \tilde{\boldsymbol{q}}^K_{\text{ins}}]), \tag{14}$$

where $f_{\text{MCA}}(\cdot)$ indicates the multi-head cross-attention and $\hat{\boldsymbol{g}}_{\text{grid}}$ is an attentive grid cell. After applying $f_{\text{MCA}}(\cdot)$ on

Method	Modality	mAP	NDS	Car	Truck	C.V.	Bus	T.L.	B.R.	M.T.	Bike	Ped.	T.C.
CenterPoint [84] [CVPR 21]	L	58.0	65.5	84.6	51.0	17.5	60.2	53.2	70.9	53.7	28.7	83.4	76.7
Focals Conv [8] [CVPR 22]	L	63.8	70.0	86.7	56.3	23.8	67.7	59.5	74.1	64.5	36.3	87.5	81.4
VoxelNeXt [9] [CVPR 23]	L	64.5	70.0	84.6	53.0	28.7	64.7	55.8	74.6	73.2	45.7	85.8	79.0
TransFusion-L [1] [CVPR 22]	L	65.5	70.2	86.2	56.7	28.2	66.3	58.8	78.2	68.3	44.2	86.1	82.0
PillarNet-34 [56] [ECCV 22]	L	66.0	71.4	87.6	57.5	27.9	63.6	63.1	77.2	70.1	42.3	87.3	83.3
FocalFormer3D [11] [ICCV 23]	L	68.7	72.6	87.2	57.1	34.4	69.6	64.9	77.8	76.2	49.6	88.2	82.3
MVP [85] [NeurIPS 21]	L+C	66.4	70.5	86.8	58.5	26.1	67.4	57.3	74.8	70.0	49.3	89.1	85.0
GraphAlign [61] [ICCV 23]	L+C	66.5	70.6	87.6	57.7	26.1	66.2	57.8	74.1	72.5	49.0	87.2	86.3
PointAug. [64] [CVPR 21]	L+C	66.8	71.1	87.5	57.3	28.0	65.2	60.7	72.6	74.3	50.9	87.9	83.6
UVTR [34] [NeurIPS 22]	L+C	67.1	71.1	87.5	56.0	33.8	67.5	59.5	73.0	73.4	54.8	86.3	79.6
AutoAlignV2 [12] [ECCV 22]	L+C	68.4	72.4	87.0	59.0	33.1	69.3	59.3	-	72.9	52.1	87.6	-
TransFusion-LC [1] [CVPR 22]	L+C	68.9	71.7	87.1	60.0	33.1	68.3	60.8	78.1	73.6	52.9	88.4	86.7
BEVFusion [40] [NeurIPS 22]	L+C	69.2	71.8	88.1	60.9	34.4	69.3	62.1	78.2	72.2	52.2	89.2	85.5
BEVFusion [46] [ICRA 23]	L+C	70.2	72.9	88.6	60.1	39.3	69.8	63.8	80.0	74.1	51.0	89.2	86.5
DeepInteraction [77] [NeurIPS 22]	L+C	70.8	73.4	87.9	60.2	37.5	70.8	63.8	80.4	75.4	54.5	91.7	87.2
UniTR [66] [ICCV 23]	L+C	70.9	74.5	87.9	60.2	39.2	72.2	65.1	76.8	75.8	52.2	89.4	89.7
ObjectFusion [3] [ICCV 23]	L+C	71.0	73.3	89.4	59.0	40.5	71.8	63.1	80.0	78.1	53.2	90.7	87.7
MSMDFusion [3] [CVPR 23]	L+C	71.5	74.0	88.4	61.0	35.2	71.4	64.2	80.7	76.9	58.3	90.6	88.1
FocalFormer3D [11] [ICCV 23]	L+C	71.6	73.9	88.5	61.4	35.9	71.7	66.4	79.3	80.3	57.1	89.7	85.3
SparseFusion [71] [ICCV 23]	L+C	72.0	73.8	88.0	60.2	38.7	72.0	64.9	79.2	78.5	59.8	90.9	87.9
CMT [74] [ICCV 23]	L+C	72.0	74.1	88.0	63.3	37.3	75.4	65.4	78.2	79.1	60.6	87.9	84.7
IS-FUSION (Ours)	L+C	73.0	75.2	88.3	62.7	38.4	74.9	67.3	78.1	82.4	59.5	89.3	89.2
IS-FUSION† (Ours)	L+C	**76.5**	**77.4**	**89.8**	**67.8**	**44.5**	**77.6**	**68.3**	**81.8**	**85.3**	**65.6**	**93.4**	**91.1**

Table 1. **3D Object Detection Performance on the nuScenes test set.** 'L' is the LiDAR and 'C' denotes the camera. 'C.V.', 'T.L.', 'B.R.', 'M.T.', 'Ped.', and 'T.C.' indicate the construction vehicle, trailer, barrier, motorcycle, pedestrian, and traffic cone, respectively. '†' denotes the model with test-time augmentation and model ensemble techniques. The best results in each column are marked in bold font. IS-FUSION achieves superior performance compared to all the other published 3D detection works.

all the grid cells, we rearrange the obtained grid features $\{\hat{\boldsymbol{g}}^1_{\text{grid}}, \hat{\boldsymbol{g}}^2_{\text{grid}}, \ldots, \hat{\boldsymbol{g}}^{W\times H}_{\text{grid}}\}$ back into a BEV feature $\hat{\boldsymbol{B}}_{\text{F}} \in \mathbb{R}^{W\times H\times C}$, which will be employed in the subsequent decoding stage to produce the final 3D detections.

4. Experiments

4.1. Experimental Setup

Dataset. We evaluate the 3D object detection performance of the proposed IS-FUSION by comparing it with other state-of-the-art approaches on the nuScenes benchmark [2]. nuScenes is a very challenging large-scale autonomous driving dataset that is widely used for evaluating multi-modality 3D object detectors. It provides 700, 150, and 150 scene sequences for training, validation, and testing, respectively. Each sequence in the dataset consists of approximately 40 frames of annotated LiDAR point cloud data, and each point cloud data is accompanied by six calibrated image data covering 360° field of view. It requires detecting 10 object categories that are commonly observed in driving scenarios. The evaluation of 3D object detection is based on two key metrics: mean Average Precision (mAP) and nuScenes detection scores (NDS). In particular, NDS is a comprehensive metric that consolidates object translation, scale, orientation, velocity and attribute.

Network Architecture. Our implementation follows the open-source framework MMDetection3D [13]. Specifically, the point cloud covers [-54m, 54m] along the X and Y axes, and [-5m, 3m] along the Z axis, with a voxel size of (0.075m, 0.075m, 0.2m). In the Point-to-Grid transformer, we set the pillar size to (0.6m, 0.6m, 8.0m). The input resolution of multi-view images is set to 384×1056. The BEV feature map is of size 180×180. In HSF, we define the point number L as 20 and the region size M as 6. In IGF, the number of instance candidates K is set to 200. The number of sampling locations D on the multimodal feature is set to 16. For the model ensemble, multiple models are utilized with voxel sizes ranging from (0.05m, 0.05m, 0.2m) to (0.125m, 0.125m, 0.2m) with intervals of 0.025m. For the test-time augmentation, we apply double flipping and rotations (*i.e.*, $\{0°, \pm 22.5°, \pm 180°\}$) on the input point clouds.

Training. The image encoder is pre-trained on the nuImage dataset [2] following current approaches [1, 46, 77]. The full model is trained end-to-end for 10 epochs with the AdamW optimizer [47]. Meanwhile, the once-cycle learning policy [60] is employed with a maximum learning rate of $1e^{-3}$. The class-balanced sampling strategy from CBGS [91] and the cross-modal data augmentation from AutoAlignV2 [12] are adopted during training. The design of the 3D decoder follows the common practices of leading approaches, such as TransFusion-L [1] and BEVFusion [46], where we decode the top 200 bounding boxes.

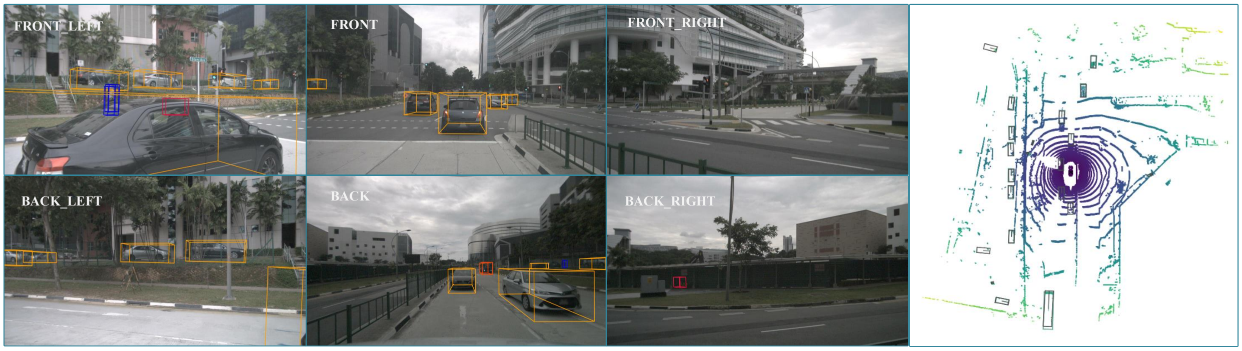


Figure 5. **Examples of 3D object detections** on nuScenes validation set. We visualize the 3D bounding boxes of car, pedestrian and bicycle with **orange**, **blue** and **red** colors in the multi-view images. In the point cloud, the predictions are in **gray** and GTs are in **green**.

Method	Image Encoder	mAP	NDS	FPS
FUTR3D [7] [CVPRW 23]	ResNet-101	64.2	68.0	2.3
TransFusion-LC [1] [CVPR 22]	ResNet-50	67.5	71.3	3.2
BEVFusion [46] [ICRA 23]	Swin-T	68.5	71.4	4.2
DeepInteraction [77] [NeurIPS 22]	Swin-T	69.9	72.6	2.6
CMT [74] [ICCV 23]	VoV-99	70.3	72.9	3.8
SparseFusion [71] [ICCV 23]	Swin-T	71.0	73.1	**5.3**
IS-Fusion (Ours)	Swin-T	**72.8**	**74.0**	3.2

Table 2. **Performance comparison on the nuScenes validation set.** IS-Fusion achieves superior 3D detection performance while maintaining a comparable inference speed.

	Baseline-L	Baseline-LC	HSF	IGF	mAP	NDS
(a)	✓				65.4	70.1
(b)		✓			69.4	71.6
(c)		✓	✓		71.6	73.2
(d)		✓		✓	70.9	72.8
(e)		✓	✓	✓	**72.8**	**74.0**

Table 3. **Ablation studies for each module in IS-Fusion** on the nuScenes validation set. Baseline-L indicates the LiDAR-only baseline, while Baseline-LC refers to a simple variant of IS-Fusion without employing the HSF or IGF modules.

4.2. Performance Benchmarking

In Table 1, we benchmark the performance of our model against current leading LiDAR-based (indicated by 'L') and multimodal (indicated by 'L+C') 3D object detectors on the nuScenes test set. It demonstrates that IS-Fusion outperforms all existing state-of-the-art (SOTA) 3D detection algorithms. Specifically, the LiDAR-only baseline of IS-Fusion is built upon TransFusion-L [1]. By exploring instance-scene collaborative fusion, IS-Fusion significantly improves it by 7.5% in mAP and 5.0% in NDS, respectively. Furthermore, IS-Fusion demonstrates superior performance compared to some very recent multimodal detection works such as FocalFormer3D [11], SparseFusion [71] and CMT [74], outperforming them by 1.4%, 1.0% and 1.0% in mAP, respectively. Notably, IS-Fusion obtains the highest results in some categories with fewer labeled instances, *i.e.*, motorcycle and trailers (constituting only 1.08% and 2.13% of the dataset). This suggests that IS-Fusion captures essential information even from limited instances. By applying test-time augmentation and model ensemble, IS-Fusion† achieves a new SOTA on the highly competitive nuScenes leaderboard.

As shown in Table 2, IS-Fusion also obtains the best detection accuracy on the nuScenes validation set, meanwhile keeping a comparable inference speed. In particular, it significantly surpasses the SOTA detectors like CMT and SparseFusion by 2.5% and 1.8% in mAP, respectively. In Fig. 5, we additionally present some qualitative detection results on the nuScenes validation set to showcase the performance of IS-Fusion. The visualization reveals that IS-Fusion is capable of accurately detecting objects of various classes, even at distant ranges and with varying scales. Overall, the promising performance of IS-Fusion can be attributed to the joint modeling of the multimodal instance-level and scene-level contexts, as well as their effective collaboration in enhancing the BEV representation.

4.3. Ablation Studies

4.3.1 Component-wise Ablation

In this section, we investigate the contribution of each component in our model. We begin by introducing the baseline frameworks of IS-Fusion. Concretely, our LiDAR-only baseline derives from Transfusion-L [1], which is reimplemented here as Baseline-L. For the multimodal baseline, denoted as Baseline-LC, we adopt a straightforward approach that combines the point cloud and image BEV features via a convolutional layer (see Eq. (6)). To obtain the image BEV features, we summarize the point features in each pillar through summation operation, where the point features are determined by the image features as introduced in Eq. (4). According to Table 3 (a)-(b), this intuitive fusion solution achieves 69.4% mAP and 71.6% NDS, outperforming Baseline-L by 4.0% in mAP and 1.5% in NDS. By leveraging HSF to enhance the scene feature, it improves 2.2% in

Image Encoder	Resolution	mAP	NDS
ResNet-50 [24]	320 × 800	71.3	72.8
CSPNet [65]	384 × 1056	71.7	73.1
Swin-T [45]	256 × 704	72.4	73.7
Swin-T [45]	384 × 1056	**72.8**	**74.0**

(a) Performance with different image encoders

Components	mAP	NDS
Baseline-LC	69.4	71.6
+ Point-to-Grid Attn.	69.9	71.9
+ Grid-to-Region Attn.	71.2	72.8
Full HSF module	**71.6**	**73.2**

(b) Component-wise ablation studies on HSF

Hyper-parameter	mAP	NDS
K=64; D=16	69.8	72.0
K=200; D=16	**70.9**	**72.8**
K=200; D=32	70.6	72.7
K=300; D=16	70.4	72.5

(c) Number of instances and neighbors in IGF

Table 4. **Design choices of IS-FUSION.** We explore the impact of various components in HSF and the optimal hyper-parameters in IGF.

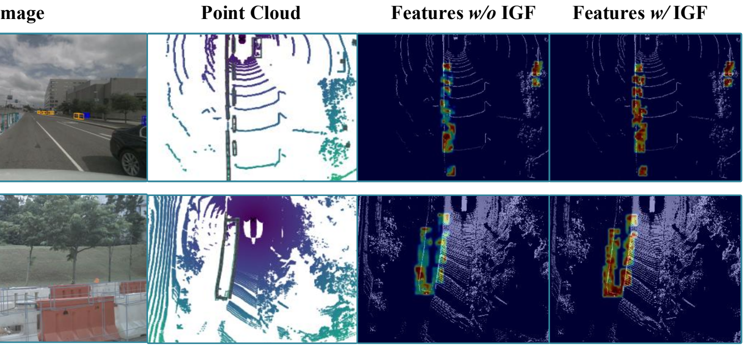


Figure 6. **Visualization of the BEV features** in challenging scenarios with traffic cones. We show the BEV features of models *w/* and *w/o* IGF. It demonstrates that IGF can yield instance representation with higher responses and more complete patterns.

mAP and 1.6% in NDS in terms of Table 3 (b)-(c). To verify the effect of instance-level modeling (*i.e*., Table 3 (b)-(d)), it shows that IGF outperforms Baseline-LC by 1.5% in mAP and 1.2% in NDS. This highlights the crucial role of instance representation. In Table 3 (e), our full model, utilizing both HSF and IGF, achieves the best performance of 72.8% mAP and 74.0% NDS, demonstrating the effect of instance-scene collaboration.

Additionally, in Table 4(a), we explore the impact of different image encoders and input resolutions. It shows that Swin-T [45] outperforms other image encoders, such as ResNet-50 [24] and CSPNet [65]. It indicates that utilizing more powerful image encoders can potentially enhance the detection performance of IS-FUSION. Furthermore, using a larger input image resolution (e.g., 384 × 1056) also leads to a slight performance improvement.

4.3.2 Analysis of HSF

The HSF module is designed to hierarchically extract multimodal features at various granularities, facilitating a comprehensive description of the scene context. Therefore, we examine the effectiveness of using different feature granularities in HSF. According to Table 4(b), the Point-to-Grid transformer, focusing on point-wise and grid-wise features, shows an improvement over Baseline-LC by 0.5% in mAP and 0.3% in NDS. The Grid-to-Region transformer improves Baseline-LC by 1.8% in mAP and 1.2% in NDS, by exploring the inter-grid and inter-region features. It suggests that a larger receptive field is more crucial for 3D object detection. The full HSF results in an improvement of 2.2% in mAP and 1.6% in NDS, highlighting the benefits of feature integration across different granularities.

4.3.3 Analysis of IGF

The IGF module aggregates the local multimodal feature around each instance, and incorporates necessary instance-level information into the BEV scene feature. In IGF, there are two hyper-parameters that need to be determined, namely, the instance number (K) and the sampled neighbor number (D) in the multimodal feature. According to Table 4(c), we found that setting $K = 200$ and $D = 16$ yields better performance, achieving 70.9% mAP and 72.8% NDS. Further increasing K or D does not lead to additional improvement, which suggests that the self-attention between instances has effectively explored a suitable receptive field. Additionally, we provide visualization of the BEV feature maps for models with and without IGF. As shown in Fig. 6, the feature maps without IGF tend to exhibit incomplete patterns and lower responses, while the IGF module significantly enhances the quality of the feature map, due to the interactive collaboration with the instance-level feature.

5. Conclusions

This work presents an innovative fusion framework, IS-FUSION, for multimodal 3D object detection. It consists of two essential modules, *i.e*., the Hierarchical Scene Fusion (HSF) module and the Instance-Guided Fusion (IGF) module. In particular, Point-to-Grid and Grid-to-Region transformer attentions are designed in HSF to capture hierarchical scene context. Furthermore, IGF is introduced to mine instances, explore inter-instance relationships and incorporate rich multimodal context around instance. We also propose an Instance-to-Scene transformer attention to encourage the collaboration between the instance and scene representations. IS-FUSION achieves superior performance on the competitive nuScenes benchmark. It provides a fresh perspective to current BEV-based perception models by emphasizing instance-level context, which is potentially beneficial to a spectrum of instance-centric tasks.

Acknowledgements. This work was supported in part by the FDCT grants 0154/2022/A3, 0102/2023/RIA2, and 001/2024/SKL, the MYRG-CRG2022-00013-IOTSC-ICI grant, the SRG2022-00023-IOTSC grant, the General Program of the National Natural Science Foundation of China (No. 62372405) and CCF-Tencent Open Fund.

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. 2, 6, 7

[2] 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

[3] Qi Cai, Yingwei Pan, Ting Yao, Chong-Wah Ngo, and Tao Mei. Objectfusion: Multi-modal 3d object detection with object-centric fusion. In *ICCV*, 2023. 2, 3, 6

[4] Runnan Chen, Youquan Liu, Lingdong Kong, Nenglun Chen, Xinge Zhu, Yuexin Ma, Tongliang Liu, and Wenping Wang. Towards label-free scene understanding by vision foundation models. In *NeurIPS*, 2023. 2

[5] Runnan Chen, Youquan Liu, Lingdong Kong, Xinge Zhu, Yuexin Ma, Yikang Li, Yuenan Hou, Yu Qiao, and Wenping Wang. Clip2scene: Towards label-efficient 3d scene understanding by clip. In *CVPR*, 2023. 1

[6] Xiaozhi Chen, Huimin Ma, Ji Wan, Bo Li, and Tian Xia. Multi-view 3d object detection network for autonomous driving. In *CVPR*, 2017. 2

[7] Xuanyao Chen, Tianyuan Zhang, Yue Wang, Yilun Wang, and Hang Zhao. Futr3d: A unified sensor fusion framework for 3d detection. In *CVPRW*, 2023. 7

[8] Yukang Chen, Yanwei Li, Xiangyu Zhang, Jian Sun, and Jiaya Jia. Focal sparse convolutional networks for 3d object detection. In *CVPR*, 2022. 6

[9] Yukang Chen, Jianhui Liu, Xiangyu Zhang, Xiaojuan Qi, and Jiaya Jia. Voxelnext: Fully sparse voxelnet for 3d object detection and tracking. In *CVPR*, 2023. 6

[10] Yilun Chen, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Fast point r-cnn. In *ICCV*, 2019. 2

[11] Yilun Chen, Zhiding Yu, Yukang Chen, Shiyi Lan, Animashree Anandkumar, Jiaya Jia, and Jose Alvarez. Focalformer3d: Focusing on hard instance for 3d object detection. In *ICCV*, 2023. 6, 7

[12] Zehui Chen, Zhenyu Li, Shiquan Zhang, Liangji Fang, Qinhong Jiang, and Feng Zhao. Autoalignv2: Deformable feature aggregation for dynamic multi-modal 3d object detection. In *ECCV*, 2022. 2, 6

[13] MMDetection3D Contributors. MMDetection3D: OpenMMLab next-generation platform for general 3D object detection. https://github.com/open-mmlab/mmdetection3d, 2020. 6

[14] 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. In *ICLR*, 2021. 4

[15] Tuo Feng, Ruijie Quan, Xiaohan Wang, Wenguan Wang, and Yi Yang. Interpretable3d: An ad-hoc interpretable classifier for 3d point clouds. In *AAAI*, 2024. 2

[16] Tuo Feng, Wenguan Wang, Fan Ma, and Yi Yang. Lsknet: Towards effective and efficient 3d perception with large sparse kernels. In *CVPR*, 2024. 2

[17] Tuo Feng, Wenguan Wang, Xiaohan Wang, Yi Yang, and Qinghua Zheng. Clustering based point cloud representation learning for 3d analysis. In *ICCV*, 2023. 1

[18] Justin Gilmer, Samuel S Schoenholz, Patrick F Riley, Oriol Vinyals, and George E Dahl. Neural message passing for quantum chemistry. In *ICML*, 2017. 1

[19] Benjamin Graham, Martin Engelcke, and Laurens Van Der Maaten. 3d semantic segmentation with submanifold sparse convolutional networks. In *CVPR*, 2018. 1

[20] Wencheng Han, Junbo Yin, Xiaogang Jin, Xiangdong Dai, and Jianbing Shen. Brnet: Exploring comprehensive features for monocular depth estimation. In *ECCV*, 2022. 1

[21] Wencheng Han, Junbo Yin, and Jianbing Shen. Self-supervised monocular depth estimation by direction-aware cumulative convolution network. In *ICCV*, 2023. 2

[22] 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. 2

[23] Chenhang He, Ruihuang Li, Yabin Zhang, Shuai Li, and Lei Zhang. Msf: Motion-guided sequential fusion for efficient 3d object detection from point cloud sequences. In *CVPR*, 2023. 1

[24] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 8

[25] Jordan SK Hu, Tianshu Kuai, and Steven L Waslander. Point density-aware voxels for lidar 3d object detection. In *CVPR*, 2022. 2

[26] Yu Huang, Chenzhuang Du, Zihui Xue, Xuanyao Chen, Hang Zhao, and Longbo Huang. What makes multi-modal learning better than single (provably). In *NeurIPS*, 2021. 2

[27] Jason Ku, Melissa Mozifian, Jungwook Lee, Ali Harakeh, and Steven L Waslander. Joint 3d proposal generation and object detection from view aggregation. In *IROS*, 2018. 2

[28] Harold W Kuhn. The hungarian method for the assignment problem. *Naval research logistics quarterly*, 2(1-2):83–97, 1955. 4

[29] 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. 2, 4

[30] Peiliang Li, Xiaozhi Chen, and Shaojie Shen. Stereo r-cnn based 3d object detection for autonomous driving. In *CVPR*, 2019. 2

[31] Xin Li, Tao Ma, Yuenan Hou, Botian Shi, Yucheng Yang, Youquan Liu, Xingjiao Wu, Qin Chen, Yikang Li, Yu Qiao, et al. Logonet: Towards accurate 3d object detection with local-to-global cross-modal fusion. In *CVPR*, 2023. 2

[32] Xiang Li, Junbo Yin, Wei Li, Cheng-Zhong Xu, Ruigang Yang, and Jianbing Shen. Di-v2x: Learning domain-invariant representation for vehicle-infrastructure collaborative 3d object detection. In *AAAI*, 2024. 2

[33] Xiang Li, Junbo Yin, Botian Shi, Yikang Li, Ruigang Yang, and Jianbing Shen. Lwsis: Lidar-guided weakly supervised instance segmentation for autonomous driving. In *AAAI*, 2023. 1

[34] 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. 1, 2, 6

[35] Yanwei Li, Xiaojuan Qi, Yukang Chen, Liwei Wang, Zeming Li, Jian Sun, and Jiaya Jia. Voxel field fusion for 3d object

detection. In *CVPR*, 2022. 2

[36] 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. 2

[37] 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. 2

[38] Ming Liang, Bin Yang, Yun Chen, Rui Hu, and Raquel Urtasun. Multi-task multi-sensor fusion for 3d object detection. In *CVPR*, 2019. 2

[39] Ming Liang, Bin Yang, Shenlong Wang, and Raquel Urtasun. Deep continuous fusion for multi-sensor 3d object detection. In *ECCV*, 2018. 2

[40] 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. In *NeurIPS*, 2022. 1, 2, 3, 6

[41] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *ICCV*, 2017. 4

[42] Rui Liu, Xiaohan Wang, Wenguan Wang, and Yi Yang. Bird's-eye-view scene graph for vision-language navigation. In *ICCV*, 2023. 1

[43] Youquan Liu, Lingdong Kong, Jun Cen, Runnan Chen, Wenwei Zhang, Liang Pan, Kai Chen, and Ziwei Liu. Segment any point cloud sequences by distilling vision foundation models. In *NeurIPS*, 2023. 2

[44] Yingfei Liu, Tiancai Wang, Xiangyu Zhang, and Jian Sun. Petr: Position embedding transformation for multi-view 3d object detection. In *ECCV*, 2022. 2

[45] 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, 5, 8

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

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

[48] Yuhang Lu, Qi Jiang, Runnan Chen, Yuenan Hou, Xinge Zhu, and Yuexin Ma. See more and know more: Zero-shot point cloud segmentation via multi-modal visual data. In *ICCV*, 2023. 2

[49] Qinghao Meng, Wenguan Wang, Tianfei Zhou, Jianbing Shen, Yunde Jia, and Luc Van Gool. Towards a weakly supervised framework for 3d point cloud object detection and annotation. *IEEE TPAMI*, 44(8):4454–4468, 2021. 1

[50] Zhenwei Miao, Jikai Chen, Hongyu Pan, Ruiwen Zhang, Kaixuan Liu, Peihan Hao, Jun Zhu, Yang Wang, and Xin Zhan. Pvgnet: A bottom-up one-stage 3d object detector with integrated multi-level features. In *CVPR*, 2021. 2

[51] AJ Piergiovanni, Vincent Casser, Michael S Ryoo, and Anelia Angelova. 4d-net for learned multi-modal alignment. In *ICCV*, 2021. 2

[52] Charles R Qi, Or Litany, Kaiming He, and Leonidas J Guibas. Deep hough voting for 3d object detection in point clouds. In *ICCV*, 2019. 2

[53] Charles R Qi, Wei Liu, Chenxia Wu, Hao Su, and Leonidas J Guibas. Frustum pointnets for 3d object detection from rgb-d data. In *CVPR*, 2018. 1

[54] 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. 2

[55] 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. 1, 2

[56] Guangsheng Shi, Ruifeng Li, and Chao Ma. Pillarnet: Real-time and high-performance pillar-based 3d object detection. In *ECCV*, 2022. 6

[57] 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. 2

[58] Shaoshuai Shi, Xiaogang Wang, and Hongsheng Li. Pointrcnn: 3d object proposal generation and detection from point cloud. In *CVPR*, 2019. 2

[59] Weijing Shi and Raj Rajkumar. Point-gnn: Graph neural network for 3d object detection in a point cloud. In *CVPR*, 2020. 2

[60] Leslie N Smith. Cyclical learning rates for training neural networks. In *WACV*, 2017. 6

[61] Ziying Song, Haiyue Wei, Lin Bai, Lei Yang, and Caiyan Jia. Graphalign: Enhancing accurate feature alignment by graph matching for multi-modal 3d object detection. In *ICCV*, 2023. 6

[62] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 4

[63] Sourabh Vora, Alex H Lang, Bassam Helou, and Oscar Beijbom. Pointpainting: Sequential fusion for 3d object detection. In *CVPR*, 2020. 2

[64] Chunwei Wang, Chao Ma, Ming Zhu, and Xiaokang Yang. Pointaugmenting: Cross-modal augmentation for 3d object detection. In *CVPR*, 2021. 2, 6

[65] Chien-Yao Wang, Hong-Yuan Mark Liao, Yueh-Hua Wu, Ping-Yang Chen, Jun-Wei Hsieh, and I-Hau Yeh. Cspnet: A new backbone that can enhance learning capability of cnn. In *CVPR*, 2020. 8

[66] Haiyang Wang, Hao Tang, Shaoshuai Shi, Aoxue Li, Zhenguo Li, Bernt Schiele, and Liwei Wang. Unitr: A unified and efficient multi-modal transformer for bird's-eye-view representation. In *ICCV*, 2023. 6

[67] Li Wang, Liang Du, Xiaoqing Ye, Yanwei Fu, Guodong Guo, Xiangyang Xue, Jianfeng Feng, and Li Zhang. Depth-conditioned dynamic message propagation for monocular 3d object detection. In *CVPR*, 2021. 2

[68] Yan Wang, Wei-Lun Chao, Divyansh Garg, Bharath Hariharan, Mark Campbell, and Kilian Q Weinberger. Pseudo-lidar from visual depth estimation: Bridging the gap in 3d object detection for autonomous driving. In *ECCV*, 2019. 2

[69] Yan Wang, Junbo Yin, Wei Li, Pascal Frossard, Ruigang Yang, and Jianbing Shen. Ssda3d: Semi-supervised domain adaptation for 3d object detection from point cloud. In *AAAI*, 2023. 2

[70] Hai Wu, Chenglu Wen, Shaoshuai Shi, Xin Li, and Cheng

Wang. Virtual sparse convolution for multimodal 3d object detection. In *CVPR*, 2023. 1

[71] Yichen Xie, Chenfeng Xu, Marie-Julie Rakotosaona, Patrick Rim, Federico Tombari, Kurt Keutzer, Masayoshi Tomizuka, and Wei Zhan. Sparsefusion: Fusing multi-modal sparse representations for multi-sensor 3d object detection. In *ICCV*, 2023. 2, 3, 6, 7

[72] Danfei Xu, Dragomir Anguelov, and Ashesh Jain. Pointfusion: Deep sensor fusion for 3d bounding box estimation. In *CVPR*, 2018. 2

[73] Shaoqing Xu, Dingfu Zhou, Jin Fang, Junbo Yin, Zhou Bin, and Liangjun Zhang. Fusionpainting: Multimodal fusion with adaptive attention for 3d object detection. In *IEEE International Intelligent Transportation Systems Conference (ITSC)*, 2021. 2

[74] Junjie Yan, Yingfei Liu, Jianjian Sun, Fan Jia, Shuailin Li, Tiancai Wang, and Xiangyu Zhang. Cross modal transformer via coordinates encoding for 3d object dectection. In *ICCV*, 2023. 2, 3, 6, 7

[75] Yan Yan, Yuxing Mao, and Bo Li. Second: Sparsely embedded convolutional detection. *Sensors*, 18(10):3337, 2018. 2

[76] Honghui Yang, Zili Liu, Xiaopei Wu, Wenxiao Wang, Wei Qian, Xiaofei He, and Deng Cai. Graph r-cnn: Towards accurate 3d object detection with semantic-decorated local graph. In *ECCV*, 2022. 2

[77] Zeyu Yang, Jiaqi Chen, Zhenwei Miao, Wei Li, Xiatian Zhu, and Li Zhang. Deepinteraction: 3d object detection via modality interaction. In *NeurIPS*, 2022. 1, 2, 3, 6, 7

[78] Zetong Yang, Yanan Sun, Shu Liu, and Jiaya Jia. 3dssd: Point-based 3d single stage object detector. In *CVPR*, 2020. 2

[79] Zetong Yang, Yanan Sun, Shu Liu, Xiaoyong Shen, and Jiaya Jia. Std: Sparse-to-dense 3d object detector for point cloud. In *ICCV*, 2019. 2

[80] Zhuyu Yao, Jiangbo Ai, Boxun Li, and Chi Zhang. Efficient detr: improving end-to-end object detector with dense prior. *arXiv preprint arXiv:2104.01318*, 2021. 4

[81] Junbo Yin, Jin Fang, Dingfu Zhou, Liangjun Zhang, Cheng-Zhong Xu, Jianbing Shen, and Wenguan Wang. Semi-supervised 3d object detection with proficient teachers. In *ECCV*, 2022. 2

[82] Junbo Yin, Jianbing Shen, Xin Gao, David J Crandall, and Ruigang Yang. Graph neural network and spatiotemporal transformer attention for 3d video object detection from point clouds. *IEEE TPAMI*, 45(8):9822–9835, 2021. 1

[83] Junbo Yin, Dingfu Zhou, Liangjun Zhang, Jin Fang, Cheng-Zhong Xu, Jianbing Shen, and Wenguan Wang. Proposalcontrast: Unsupervised pre-training for lidar-based 3d object detection. In *ECCV*, 2022. 2

[84] Tianwei Yin, Xingyi Zhou, and Philipp Krahenbuhl. Center-based 3d object detection and tracking. In *CVPR*, 2021. 5, 6

[85] Tianwei Yin, Xingyi Zhou, and Philipp Krähenbühl. Multimodal virtual point 3d detection. In *NeurIPS*, 2021. 2, 6

[86] Yanan Zhang, Jiaxin Chen, and Di Huang. Cat-det: Contrastively augmented transformer for multi-modal 3d object detection. In *CVPR*, 2022. 2

[87] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *ICCV*, 2021. 1

[88] Chao Zhou, Yanan Zhang, Jiaxin Chen, and Di Huang. Octr: Octree-based transformer for 3d object detection. In *CVPR*, 2023. 1

[89] Dingfu Zhou, Jin Fang, Xibin Song, Liu Liu, Junbo Yin, Yuchao Dai, Hongdong Li, and Ruigang Yang. Joint 3d instance segmentation and object detection for autonomous driving. In *CVPR*, 2020. 2

[90] Yin Zhou and Oncel Tuzel. Voxelnet: End-to-end learning for point cloud based 3d object detection. In *CVPR*, 2018. 2, 3

[91] Benjin Zhu, Zhengkai Jiang, Xiangxin Zhou, Zeming Li, and Gang Yu. Class-balanced grouping and sampling for point cloud 3d object detection. *arXiv preprint arXiv:1908.09492*, 2019. 6

[92] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. In *ICLR*, 2021. 5

[93] Ziyue Zhu, Qiang Meng, Xiao Wang, Ke Wang, Liujiang Yan, and Jian Yang. Curricular object manipulation in lidar-based object detection. In *CVPR*, 2023. 1