Camera-Based 3D Semantic Scene Completion With Sparse Guidance Network

Jianbiao Mei, Yu Yang, Mengmeng Wang, Junyu Zhu, *Member, IEEE*, Jongwon Ra, Yukai Ma, Laijian Li, and Yong Liu

***Abstract*— Semantic scene completion (SSC) aims to predict the semantic occupancy of each voxel in the entire 3D scene from limited observations, which is an emerging and critical task for autonomous driving. Recently, many studies have turned to camera-based SSC solutions due to the richer visual cues and cost-effectiveness of cameras. However, existing methods usually rely on sophisticated and heavy 3D models to process the lifted 3D features directly, which are not discriminative enough for clear segmentation boundaries. In this paper, we adopt the dense-sparse-dense design and propose a one-stage camera-based SSC framework, termed SGN, to propagate semantics from the semantic-aware seed voxels to the whole scene based on spatial geometry cues. Firstly, to exploit depth-aware context and dynamically select sparse seed voxels, we redesign the sparse voxel proposal network to process points generated by depth prediction directly with the coarse-to-fine paradigm. Furthermore, by designing hybrid guidance (sparse semantic and geometry guidance) and effective voxel aggregation for spatial geometry cues, we enhance the feature separation between different categories and expedite the convergence of semantic propagation. Finally, we devise the multi-scale semantic propagation module for flexible receptive fields while reducing the computation resources. Extensive experimental results on the SemanticKITTI and SSCBench-KITTI-360 datasets demonstrate the superiority of our SGN over existing state-of-the-art methods. And even our lightweight version SGN-L achieves notable scores of 14.80% mIoU and 45.45% IoU on SeamnticKITTI validation with only 12.5 M parameters and 7.16 G training memory. Code is available at https://github.com/Jieqianyu/SGN.**

***Index Terms*— Semantic scene completion, sparse guidance network, hybrid guidance, voxel aggregation.**

I. Introduction

IN RECENT years, there has been significant attention and rapid progress in 3D scene understanding, which constitutes the bedrock of autonomous driving systems and robotics. By precisely perceiving the occupancy and semantics of their surroundings, autonomous vehicles, and robotics can make informed decisions and navigate safely. To this end, Semantic Scene Completion (SSC) has been introduced to predict the semantic occupancy of each voxel of the entire 3D scene from limited observation. SSC helps create a more comprehensive representation of the environment, which includes filling in the gaps or missing information in the sensor data. This can be essential for agents to identify obstacles, understand the road layout, and make safe decisions. However, accurately estimating the semantics and geometry of the real world from partial observations is challenging due to the complexities presented by real-world scenarios.

SSC has attracted extensive studies due to its application prospects for downstream tasks such as mapping and planning. When working with outdoor driving scenes, LiDAR has emerged as a popular input modality for many existing methods [1], [2], [3], [4], [5], [6] to capture 3D information of surroundings, but it suffers from high-cost sensors. Recently, there has been a shift towards camera-based SSC solutions. As the pioneer, MonoScene [7] proposed the first framework for monocular 3D SSC, utilizing mapping projection to lift RGB images to 3D volumes processed with the 3D UNet. Afterward, many camera-based methods such as OccDepth [8], SurroundOcc [9], and OccFormer [10] are developed with a similar pipeline consisting of the image backbone, view transformer, and 3D model, as illustrated in Figure 1 (a). However, they rely on sophisticated and heavy 3D models to process the lifted 3D features directly, which are not discriminative enough for clear segmentation boundaries. We explain that the lifted 3D features by 3D-2D mapping projection [7] contain many ambiguities due to the assumption of the uniform depth distribution and 2D-3D methods such as LSS [11] only utilize coarse surface information from depth distribution estimation.

On the other hand, VoxFormer [12] proposed an MAE-like architecture to complete non-visible structures using constructed visible areas. It adopts the two-stage framework, with the first stage for query proposal and the second stage for densification and segmentation. By completing the 3D scene in a **sparse-to-dense** manner shown in Figure 1 (b), VoxFormer is more efficient and scalable than the dense processing with complicated 3D models mentioned above. However, it still suffers from several limitations. Firstly, the densification stage is mainly considered from the perspective of scene completion based on queries. The intra-category feature separation of queries is neglected. Besides, the second stage only considers the information from the queries that only include partial observation and are not always accurate, increasing the

Received 2 January 2024; revised 18 June 2024; accepted 11 September 2024. Date of publication 26 September 2024; date of current version 4 October 2024. This work was supported by NSFC under Grant 62088101 and in part by the Autonomous Intelligent Unmanned Systems. The associate editor coordinating the review of this article and approving it for publication was Prof. Sebastian Knorr. *(Corresponding author: Yong Liu.)*

The authors are with the Institute of Cyber-Systems and Control, Zhejiang University, Hangzhou 310027, China (e-mail: jianbiaomei@zju.edu.cn; yu.yang@zju.edu.cn; mengmengwang@zju.edu.cn; juyunzhu@zju.edu.cn; jongwonra@zju.edu.cn; yukaima@zju.edu.cn; lilaijian@zju.edu.cn; yongliu@iipc.zju.edu.cn).

Digital Object Identifier 10.1109/TIP.2024.3461989

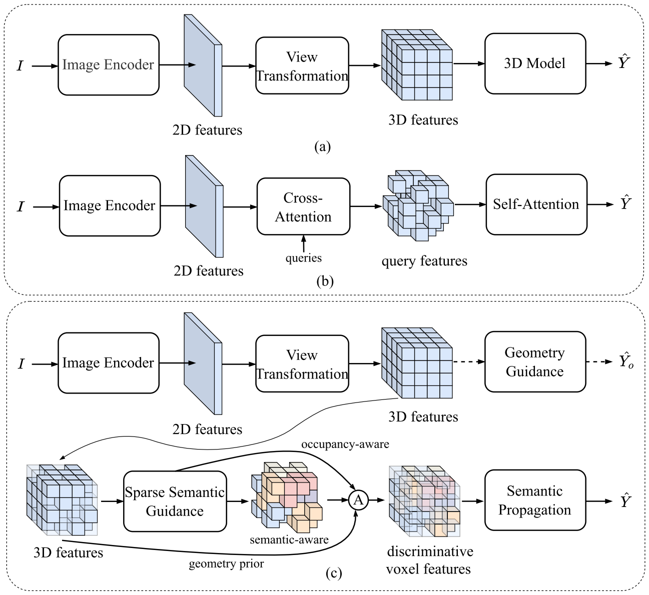

Fig. 1. (a) Fully dense processing with heavy and complex 3D model. (b) MAE-like architecture in a "sparse-to-dense" manner. (c) Our "dense-sparse-dense" design with hybrid guidance and semantic propagation. "A" denotes the voxel aggregation layer for geometry cues.

difficulty of subsequent completion and segmentation. Finally, the two-stage training and inference cannot fully consider global information due to the independent optimization of different stages. The geometry information from the first stage is also not fully utilized.

To address the above problems, we propose a novel one-stage camera-based SSC framework, **S**parse **G**uidance **N**etwork (**SGN**), to propagate semantics from the semantic-aware seed voxels to the whole scene based on spatial geometry cues, as illustrated in Figure 1 (c). Specifically, we employ the **dense-sparse-dense** design to implement the semantic propagation of semantic-aware seed features, avoiding relying on heavy 3D models to process coarse scene representations that are not discriminative enough. Firstly, to dynamically select sparse seed voxels and encode depth-aware context, we redesign the sparse voxel proposal network to directly process points generated by depth prediction with the coarse-to-fine paradigm. And by further designing hybrid guidance (sparse semantic and geometry guidance) and effective voxel aggregation for spatial geometry cues, we enhance the intra-category feature separation and expedite the convergence of the semantic propagation. We also devise the multi-scale semantic propagation module using anisotropic convolutions [13] for flexible receptive fields while reducing the computation resources. By this means, our SGN is lightweight while having a more powerful representation ability.

Extensive experiments on the challenging SemanticKITTI [14] and SSCBench-KITTI-360 [15] datasets demonstrate the superiority of our SGN over existing state-of-the-art methods. For example, on the SemanticKITTI validation set, even our lightweight version SGN-L achieves notable scores of 14.80% mIoU and 45.45% IoU with only **12.5 M** parameters and **7.16 G** memory for training, exceeding VoxFormer-T by 1.45% points in mIoU and 1.30% points in IoU while being more lightweight and less memory consumption.

Our main contributions can be summarized as follows:

- We propose a one-stage camera-based SSC framework called **SGN**, propagating semantics from the semantic- and occupancy-aware seed voxels to the whole scene based on spatial geometry cues.
- We adopt the **dense-sparse-dense** design and propose hybrid guidance and effective voxel aggregation to enhance intra-categories feature separation and expedite the convergence of the semantic propagation.
- Extensive experiments on the SemanticKITTI and SSCBench-KITTI-360 benchmarks demonstrate the effectiveness of our SGN, which is more lightweight and achieves the new state-of-the-art.

II. Related Works

A. Semantic Scene Completion

Due to the vital application of semantic occupancy prediction in autonomous driving, SSC has attracted extensive attention. After the release of the large-scale outdoor benchmark SemanticKITTI [14], many outdoor SSC methods have emerged. According to the input modality, existing outdoor methods can be mainly classified into LiDAR-based and camera-based methods.

1) LiDAR-Based Methods: consider LiDAR a primary modality to enable accurate 3D semantic occupancy prediction. Following the pioneering SSCNet [16], UDNet [17] exploits a single 3D U-Net framework to obtain predictions from the grids generated by the LiDAR points, resulting in extra computation overhead of empty voxels. Afterward, LMSCNet [1] introduces the 2D CNN for feature encoding, and SGCNet [18] uses spatial group convolutions to improve efficiency. Some solutions focus on multi-view fusion [2], local implicit functions [3], and knowledge distillation [5] for SSC. Besides, the relationships between semantic segmentation and scene completion are explored. For example, JS3C-Net [4] and SSA-SC [19] design a semantic segmentation network to assist the semantic scene completion. SSC-RS [20] design multi-branch network to fuse semantic and geometry features hierarchically.

2) Camera-Based Methods: Recently, camera-based perception such as detection [21], [22], [23], [24], [25], [26] and segmentation [24], [27], [28] is currently more attractive due to cameras' richer visual cues and cost-effectiveness. And there is also a shift towards camera-based solutions [7], [29] to SSC. MonoScene [7] first proposed to infer 3D SSC from a single monocular RGB image, which applied a classical 3D UNet network to process the voxel features projected along the line of sight. Afterward, TPVFormer [29] proposed a tri-perspective view (TPV) representation to describe the fine-grained 3D structure of a scene. VoxFormer [12] proposed an MAE-like architecture to complete non-visible structures using constructed visible areas. OccFormer [10] designed a dual-path transformer network. And SurroundOcc [9] applied 3D convolutions to upsample multi-scale voxel features progressively and devised a pipeline to generate dense SSC ground truth. Symphonize [30] modeled the scene volume with a sparse set of instance queries with context awareness. Some methods [8], [31] leveraged implicit stereo depth information

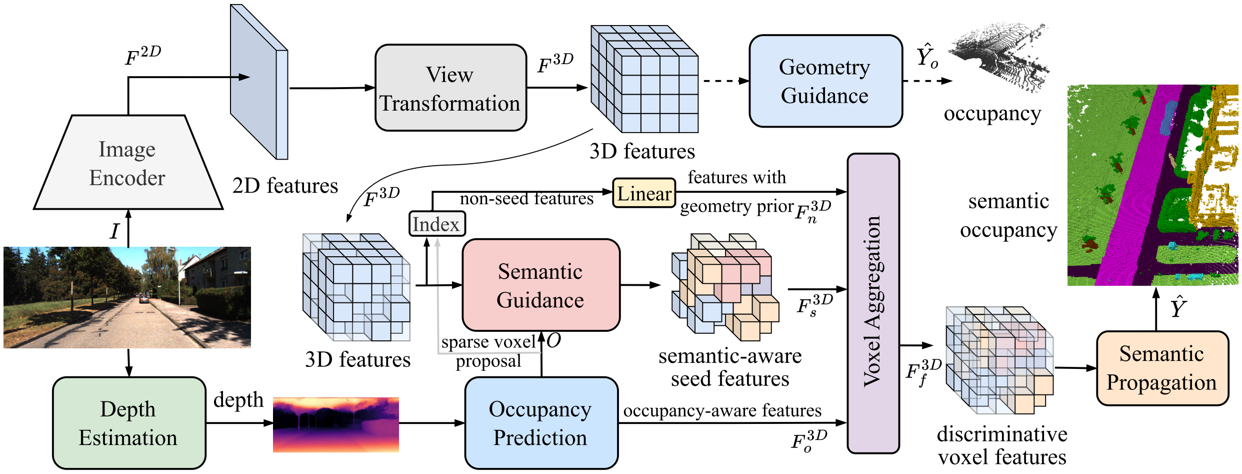

Fig. 2. Overall framework of our SGN. The image encoder extracts 2D features, establishing the foundation for the 3D features generated through view transformation. An auxiliary occupancy head is applied to provide geometry guidance. The sparse semantic guidance consists of two parts: sparse voxel proposal and semantic guidance. The depth-based occupancy prediction is designed for the sparse voxel proposal. This proposal, along with the 3D features, is fed into the subsequent semantic guidance (depicted in Figure 3) to index seed features and inject semantic context into these seed features. Afterward, the voxel aggregation layer combines the semantic-aware seed features, geometry prior from the non-seed features, and occupancy-aware features from the depth-based occupancy prediction. This forms the informative voxel features processed by the multi-scale semantic propagation for the final prediction.

and stereo matching to resolve geometry ambiguity. NDC-scene [32] extends the 2D feature map to a Normalized Device Coordinates (NDC) space to alleviate the feature ambiguity, pose ambiguity and computation imbalance. Besides, there are some SSC solutions [9], [29], [33], [34] for multi-view cameras. And multiple benchmarks [15], [35], [36] are proposed to facilitate the SSC's development.

We focus on camera-based SSC in outdoor scenarios. Compared with the existing works, our SGN proposes to propagate semantics from the semantic-aware seed voxels to the whole scene based on spatial geometry cues. SGN avoids relying on heavy and sophisticated 3D models to handle lifted voxel features with rough geometry context like many existing SSC methods [7], [8], [9], [10], [30]. Our method is built on the recent two-stage method VoxFormer [12]. However, unlike VoxFormer, our SGN is one-stage, which adopts the dense-sparse-dense design and proposes hybrid guidance and effective voxel aggregation to enhance intra-categories feature separation and expedite the convergence of the semantic propagation. Compared with VoxFormer, our SGN achieved better performance while being more lightweight and requiring less memory consumption.

3) Camera-Based 3D Perception: Camera-based 3D perception, encompassing domains such as 3D detection [21], [22], [23], [24], [25], [26], [37], [38] and segmentation [24], [27], [28], [39], [40], has gained increasing traction owing to the rich visual cues provided by cameras and their cost-effectiveness. Various monocular-based approaches have adapted 2D techniques to the 3D domain, such as FCOS3D [25] and DETR3D [22]. In recent times, a significant shift has been observed in camera-based research toward Bird's Eye View (BEV) representations [21], [24], [27], [39], [41], [42], [43], [44], facilitated by view transformation techniques such as LSS [11], OFT [45], and the cross-attention module [24]. For example, BEVDet [21] and BEVDepth [41] incorporate depth estimation to facilitate the transformation from perspective view to BEV. Additionally, BEVFormer [24] employs cross-attention to inject cues from image features to BEV queries effectively. The efficacy of BEV-based perception [20], [24], [41], [43], [46], [47] has been validated by these advancements. However, for Semantic Scene Completion (SSC) tasks, the utilization of 3D voxel representations, which encapsulate more volumetric information, becomes imperative. As such, the quest to devise discriminative 3D scene representations and to process voxel features both effectively and efficiently remains a vibrant area of ongoing research and exploration.

III. Method

A. Overview

We show the overall framework of our SGN in Figure 2. SGN adopts the dense-sparse-dense design and propagates semantics from the semantic-aware seed voxels to the whole scene based on spatial geometry cues from the non-seed features and features from the depth-based occupancy prediction. SGN takes RGB images as the input and extracts 2D features using the image encoder. Then the 3D features are obtained through the view transformation. For dynamically indexing seed voxels, we generate the sparse voxel proposal based on depth prediction. Then according to the proposal and 3D features, the hybrid guidance is designed to inject semantic and geometry cues and facilitate feature learning. Furthermore, we develop the voxel aggregation layer to form the informative voxel features, which are processed by our multi-scale semantic propagation module for the final semantic occupancy prediction.

1) Image Encoder: We use ResNet-50 [48] with FPN [49] to construct the image encoder for extracting 2D features from RGB images. The extracted features $\mathbf{F}^{2D} \in \mathbb{R}^{N_t \times C \times H \times W}$ provide a strong foundation for the subsequent voxel features, where N_t is the image number of temporal inputs, C is the feature channel and (H, W) denotes the image resolution.

2) View Transformation: Similar to MonoScene [7], we construct 3D features by sampling 2D features via 3D-2D projection mapping with camera parameters. The simple projection mapping operation provides coarse volumetric scene representation for the latter contextual modeling. And it is more convenient and concise than learnable LSS [11] and cross-attention mechanism [24].

Let $\mathbf{x} \in \mathbb{R}^{X\times Y\times Z\times 3}$ denote the centroid of $X\times Y\times Z$ voxels in world coordinates. We establish the projection mapping $\pi(\mathbf{x})$ using the camera parameters $(\mathbf{K}, \mathbf{T})$, where $\mathbf{K}$ and $\mathbf{T} = [\mathbf{R}, \mathbf{t}]$ are the cameras intrinsic and extrinsic matrices directly provided in KITTI [50]. Let p denotes a point in $\mathbf{x}$, the mapping function establishes the relationship between the point and the image pixel (u, v), which can be represented by:

$$[x_c, y_c, z_c]^T = \mathbf{R}\cdot p + \mathbf{t} \tag{1}$$

$$z_c \circ [u, v, 1]^T = \mathbf{K}\cdot [x_c, y_c, z_c]^T \tag{2}$$

where $\circ$ denote element-wise product.

We take the average of sampled features from different images for each voxel. And the features of voxels outside the field of view (FOV) are set to zero. Mathematically, the 3D features $\mathbf{F}^{3D} \in \mathbb{R}^{C\times X\times Y\times Z}$ are sampled from the 2D features $\mathbf{F}^{2D}$ as follows:

$$\mathbf{F}^{3D} = W \cdot \sum_{t=1}^{N_t} [\phi_{\pi(\mathbf{x})}(\mathbf{F}_t^{2D}) \cdot \mathbf{M}_t^{FOV}] \tag{3}$$

$$W_p = \begin{cases} 1/\delta_p, & \delta_p > 0, \\ 1, & \delta_p = 0. \end{cases} \tag{4}$$

where $\phi_a(b)$ is the sampling function that samples features b at coordinates a, $\mathbf{F}_t^{2D}$ is the 2D features of image $\mathbf{I}_t$, $\mathbf{M}_t^{FOV} \in \mathbb{R}^{1\times X\times Y\times Z}$ is the binary mask indicating the field of view of image $\mathbf{I}_t$, δ_p is the number of hit images for point p in $\mathbf{x}$, W_p is the weight value for points p in W.

B. Feature Learning With Hybrid Guidance

As discussed above, most existing methods design heavy and complicated models to directly process the 3D features $\mathbf{F}^{3D}$ produced by the view transformation module for the final semantic scene prediction. We argue that the coarse scene representation $\mathbf{F}^{3D}$ is not discriminative enough to get clear segmentation boundaries, which slows down the convergence of the model. Therefore, we propose sparse semantic guidance and geometry guidance to inject semantic and geometry cues for informative voxel features.

1) Geometry Guidance: We first attach the auxiliary 3D occupancy head as the geometry guidance after the 3D features from the view transformation module to provide coarse geometry awareness. Specifically, we construct the 3D occupancy head with an anisotropic convolution layer [13] and a linear layer. In the spirit of [51], the anisotropic convolution decomposes a 3D convolution operation into three consecutive 1D convolutions in different directions. Additionally, each of these 1D convolutions is equipped with a mixer containing distinct kernel sizes, enhancing the model's ability to learn and extract meaningful features from the input data. It can provide flexible receptive fields while alleviating resource consumption. By predicting the 3D occupancy $\hat{\mathbf{Y}}_o$ on the lifted 3D features $\mathbf{F}^{3D}$ using the auxiliary head, we apply the guidance on the coarse scene representation and provide the geometry prior for the latter seed features' semantic prediction and propagation. We optimize the occupancy probability with binary cross-entropy loss:

$$\mathscr{L}_{geo} = -\sum_i [(1-\mathbf{Y}_{o,i})\log(1-\hat{\mathbf{Y}}_{o,i}) + \mathbf{Y}_{o,i}\log(\hat{\mathbf{Y}}_{o,i})] \tag{5}$$

where i indexs the voxel of the 3D scene and $\mathbf{Y}_o$ is the occupancy ground truth. Note that the auxiliary 3D head is abandoned during inference and using geometry guidance does not introduce any extra computation.

2) Sparse Semantic Guidance: Since directly learning the semantics of all the voxels from the 3D features with coarse volumetric information is less effective and efficient, we propose propagating **semantics** from **seed** voxel to the whole scene. Specifically, we generate the sparse voxel proposal to choose seed voxels and encourage inter-category separability of seed features with semantic guidance, expediting the latter semantic propagation.

a) Sparse voxel proposal: We devise the sparse voxel proposal network (SVPN) to generate the sparse proposal for indexing seed voxels. Unlike Voxformer [12], which learns class-agnostic proposal on temporal data **offline** for voxel queries, our SVPN aims to dynamically select seed voxels **online** by occupancy probability for subsequent semantic context learning. Specifically, SVPN consists of depth estimation and coarse-to-fine occupancy prediction. Following [12] and [30], we utilize the pre-trained Mobilestereonet [53] to infer the depth prediction and calculate the scene points $\mathbf{P}$ by back-projecting the depth map into the 3D point cloud using the camera parameters $(\mathbf{K}, \mathbf{T})$. The scene points $\mathbf{P}$ imply the volumetric surface and are used for occupancy prediction. Let (u, v) denote a pixel in the depth map and p is the corresponding 3D point, the back-projecting procedure is formulated as follows:

$$p = \mathbf{R}^{-1}\cdot [\mathbf{K}^{-1}\cdot (z_c \circ [u, v, 1]^T) - \mathbf{t}] \tag{6}$$

where $\circ$ denote element-wise product, z_c is the depth value of the pixel. $\mathbf{K}$ and $\mathbf{T} = [\mathbf{R}, \mathbf{t}]$ are the intrinsic and extrinsic parameters of the camera.

Next, we generate the occupancy prediction $\mathbf{O} \in \mathbb{R}^{X\times Y\times Z}$ in a coarse-to-fine manner. Firstly, the points $\mathbf{P}$ are fed into a voxelization layer adopted from DRNet [54] for voxel-wise features. Then we apply the tiny sparse convolution network consisting of a sparse feature encoder and a sparse geometry feature encoder adopted from GASN [52] to predict the coarse occupancy probability from the voxel-wise features. Finally, the occupancy probability is further fed into a lightweight Unet-like network [1] for the final occupancy prediction $\mathbf{O}$, which is used to select the sparse voxels as explained in semantic guidance. Similar to the geometry guidance, we use the binary cross entropy loss to calculate the loss $\mathscr{L}_{occ}$ for occupancy prediction.

To further utilize the geometry information from the depth-based SVPN, we also take the occupancy-aware 3D features

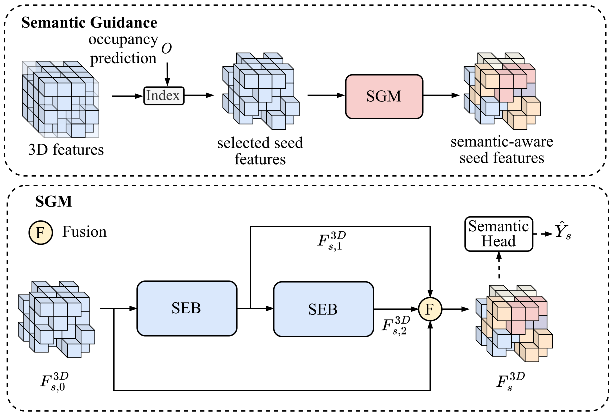

Fig. 3. Detailed architecture of the proposed semantic guidance module (SGM). The sparse encoder block (SEB) consists of a sparse feature encoder and a sparse geometry feature encoder adopted from [52].

$\mathbf{F}_o^{3D} \in \mathbb{R}^{C_o \times X \times Y \times Z}$ from the last layer of the Unet-like network for the latter voxel feature aggregation.

b) Semantic guidance: After obtaining the occupancy prediction $\mathbf{O}$ and voxel coordinates $\mathbf{V}^{3D} \in \mathbb{Z}^{3 \times X \times Y \times Z}$ of the scene, we first choose the initial seed voxel features $\mathbf{F}_{s,0}^{3D} \in \mathbb{R}^{C \times N_s}$ and seed coordinates $\mathbf{V}_s^{3D} \in \mathbb{Z}^{3 \times N_s}$ by:

$$\mathbf{V}_s^{3D} = \mathbf{V}^{3D}[:, \mathbf{O} > \theta] \tag{7}$$

$$\mathbf{F}_{s,0}^{3D} = \mathbf{F}^{3D}[:, \mathbf{O} > \theta] \tag{8}$$

where θ is the threshold to determine if the voxel is occupied and N_s is the number of non-empty voxel. Then these seed voxel features $\mathbf{F}_{s,0}^{3D}$ and corresponding voxel indices V_s^{3D} are fed into the semantic guidance module (SGM) illustrated in Figure 3 for mutual interactions. The semantic guidance module has two sparse encoder blocks (SEB), a fusion layer, and an auxiliary semantic head. Each sparse encoder block (SEB) consists of a sparse feature encoder and a sparse geometry feature encoder adopted from [52] and outputs features with multi-scale contextual information. Let $\mathbf{F}_{s,1}^{3D}, \mathbf{F}_{s,2}^{3D}$ are the outputs of the two sparse encoder blocks, the fusion feature $\mathbf{F}_s^{3D} \in \mathbb{R}^{C \times N_s}$ are obtained by:

$$\mathbf{F}_s^{3D} = \text{MLP}([\mathbf{F}_{s,0}^{3D}, \mathbf{F}_{s,1}^{3D}, \mathbf{F}_{s,2}^{3D}]) \tag{9}$$

where [.] denotes concatenate operation along feature dimension. After that, the fused features $\mathbf{F}_s^{3D}$ are fed into the auxiliary semantic head consisting of a two-layer MLP to predict the corresponding semantics $\hat{\mathbf{Y}}_s \in \mathbb{R}^{C_{class} \times N_s}$, where C_{class} is the number of classes. We calculate the cross entropy loss and lovasz loss [55] for the semantic guidance:

$$\mathscr{L}_{sem} = \mathscr{L}_{ce}(\hat{\mathbf{Y}}_s, \mathbf{Y}_s) + \mathscr{L}_{lovasz}(\hat{\mathbf{Y}}_s, \mathbf{Y}_s) \tag{10}$$

where $\mathbf{Y}_s$ is the seed voxels' semantic label indexed from the semantic scene label $\mathbf{Y}$.

By this means, we inject semantic cues into the fused seed features $\mathbf{F}_s^{3D}$ and enhance the feature separation between categories, which is the key to semantic propagation.

C. Voxel Aggregation

As shown in Figure 2, to fully exploit the geometry information in 3D features $\mathbf{F}^{3D}$ and $\mathbf{F}_o^{3D}$, we further aggregate them with the semantic-aware seed features $\mathbf{F}_s^{3D}$ to construct the final discriminative voxel features $\mathbf{F}_f^{3D} \in \mathbb{R}^{(C+C_o) \times X \times Y \times Z}$ for subsequent semantic propagation. Specifically, we leverage the coordinates of non-seed voxels to index features $\mathbf{F}_n^{3D}$ from $\mathbf{F}^{3D}$. Then the non-seed voxel features $\mathbf{F}_n^{3D}$ are fed into a linear layer and combined with the semantic-aware features $\mathbf{F}_s^{3D}$ to form the new scene representation, which contains the semantic context and geometry cues. We argue that non-seed voxel features $\mathbf{F}_n^{3D}$ are vital and can well complement the seed features since the sparse voxel proposal is not always accurate. To further utilize the geometry information from the SVPN, we also concatenate the features $\mathbf{F}_o^{3D}$ from SVPN with the new scene representation to obtain the final voxel features. The detailed procedure can be formulated as follows:

$$\mathbf{F}_f^{3D} = \text{MLP}([\text{CN}(\mathbf{F}_s^{3D}, \text{Conv1d}(\mathbf{F}_n^{3D})), \mathbf{F}_o^{3D}]) \tag{11}$$

where CN is the feature combination of seed and non-seed voxels.

D. Multi-Scale Semantic Propagation

By learning features with hybrid guidance and voxel aggregation, we obtain discriminative voxel features $\mathbf{F}_f^{3D}$ with the rich semantic context in the seed features $\mathbf{F}_s^{3D}$ and spatial geometry cues from previous 3D features $\mathbf{F}_n^{3D}$, and occupancy-aware features $\mathbf{F}_o^{3D}$. Then we design the multi-scale semantic propagation (MSSP) module to propagate the semantic information from seed features to the whole scene. The MSSP module contains three anisotropic convolutional layers [13] and the ASPP [57] module consisting of three 3D convolutions with the kernel size of 3 × 3 × 3 and dilation of 1, 2, and 4. This module is lightweight and can well capture multi-scale features of instances of different sizes through convolutional kernels with different receptive fields. Afterward, we use the head consisting of a linear layer and softmax layer to predict the final semantic scene prediction $\hat{\mathbf{Y}} \in \mathbb{R}^{C_{class} \times X \times Y \times Z}$ from the propagated voxel features.

Following MonoScene [7], we adopt the Scene-Class Affinity Loss to force the network to account for voxels within the same category as well as voxels across different categories. The Affinity Loss optimizes the class-wise derivable precision, recall, and specificity metrics simultaneously, where precision and recall evaluate the performance of voxels within the same class, while specificity assesses the performance of dissimilar voxels. Specifically, similar to [7] and [32], we apply scene-class affinity loss on both semantic and geometry results of the prediction $\hat{\mathbf{Y}}$. We integrate and optimize scene- and class-wise semantics $\mathscr{L}_{scal}^{sem}$, geometry $\mathscr{L}_{scal}^{geo}$, and cross-entropy loss $\mathscr{L}_{ce}$. The overall loss function is formulated by:

$$\mathscr{L}_{ssc} = \mathscr{L}_{scal}^{sem}(\hat{\mathbf{Y}}, \mathbf{Y}) + \mathscr{L}_{scal}^{geo}(\hat{\mathbf{Y}}, \mathbf{Y}) + \mathscr{L}_{ce}(\hat{\mathbf{Y}}, \mathbf{Y}) \tag{12}$$

E. Training Loss

Unlike VoxFormer [12] with sophisticated two-stage training, we train our SGN end-to-end. The total training loss $\mathscr{L} = \mathscr{L}_{geo} + \mathscr{L}_{occ} + \mathscr{L}_{sem} + \mathscr{L}_{ssc}$.

TABLE I

SEMANTIC SCENE COMPLETION ON SEMANTICKITTI HIDDEN TEST SET. † DENOTES THE RESULTS PROVIDED BY MONOSCENE [7]. BOLD AND UNDERLINE DENOTE THE BEST PERFORMANCE AND THE SECOND-BEST PERFORMANCE, RESPECTIVELY

Method	IoU	road (15.30%)	sidewalk (11.13%)	parking (1.12%)	otherground (0.56%)	building (14.1%)	car (3.92%)	truck (0.16%)	bicycle (0.03%)	motorcycle (0.03%)	othervehicle (0.20%)	vegetation (39.3%)	trunk (0.51%)	terrain (9.17%)	person (0.07%)	bicyclist (0.07%)	motorcyclist (0.05%)	fence (3.90%)	pole (0.29%)	trafficsign (0.08%)	mIoU
		🟪	🟪	🟪	🟥	🟨	🟦	🟪	🟦	🟦	🟦	🟩	🟫	🟩	🟥	🟪	🟪	🟧	🟨	🟥	
LMSCNet† [1]	31.38	46.70	19.50	13.50	3.10	10.30	14.30	0.30	0.00	0.00	0.00	10.80	0.00	10.40	0.00	0.00	0.00	5.40	0.00	0.00	7.07
AICNet† [13]	23.93	39.30	18.30	19.80	1.60	9.60	15.30	0.70	0.00	0.00	0.00	9.60	1.90	13.50	0.00	0.00	0.00	5.00	0.10	0.00	7.09
JS3C-Net† [4]	34.00	47.30	21.70	19.90	2.80	12.70	20.10	0.80	0.00	0.00	4.10	14.20	3.10	12.40	0.00	0.20	0.20	8.70	1.90	0.30	8.97
MonoScene [7]	34.16	54.70	27.10	24.80	5.70	14.40	18.80	3.30	0.50	0.70	4.40	14.90	2.40	19.50	1.00	1.40	0.40	11.10	3.30	2.10	11.08
TPVFormer [29]	34.25	55.10	27.20	27.40	6.50	14.80	19.20	3.70	1.00	0.50	2.30	13.90	2.60	20.40	1.10	2.40	0.30	11.00	2.90	1.50	11.26
VoxFormer [12]	42.95	53.90	25.30	21.10	5.60	19.80	20.80	3.50	1.00	0.70	3.70	22.40	7.50	21.30	1.40	2.60	0.20	11.10	5.10	4.90	12.20
OccFormer [10]	34.53	55.90	30.30	**31.50**	6.50	15.70	21.60	1.20	1.50	1.70	3.20	16.80	3.90	21.30	**2.20**	1.10	0.20	11.90	3.80	3.70	12.32
SurroundOcc [9]	34.72	56.90	28.30	30.20	6.80	15.20	20.60	1.40	1.60	1.20	4.40	14.90	3.40	19.30	1.40	2.00	0.10	11.30	3.90	2.40	11.86
NDC-scene [32]	36.19	58.12	28.05	25.31	6.53	14.90	19.13	**4.77**	**1.93**	**2.07**	**6.69**	17.94	3.49	25.01	**3.44**	**2.77**	**1.64**	12.85	4.43	2.96	12.58
SGN-S (ours)	41.88	57.80	29.20	27.70	5.20	23.90	24.90	2.70	0.40	0.30	4.00	24.20	10.00	25.80	1.10	2.50	0.30	14.20	7.40	4.40	14.01
SGN-L (ours)	43.71	57.90	29.70	25.60	5.50	27.00	25.00	1.50	0.90	0.70	3.60	26.90	12.00	26.40	0.60	0.30	0.00	14.70	9.00	6.40	14.39
SGN-T (ours)	**45.42**	**60.40**	**31.40**	28.90	**8.70**	**28.40**	**25.40**	4.50	0.90	1.60	3.70	**27.40**	**12.60**	**28.40**	0.50	0.30	0.10	**18.10**	**10.00**	**8.30**	**15.76**

IV. EXPERIMENTS

In this section, we present the datasets, evaluation metrics, and detailed implementation aspects of our approach. Subsequently, we conduct extensive experiments to establish that our proposed SGN consistently surpasses or achieves comparable performance against the state-of-the-art methods on the complex, large-scale outdoor dataset SemanticKITTI [14] as well as SSCBench-KITTI-360 [15]. Following this, we provide qualitative results to underscore the efficacy of our SGN. Moreover, we conducted detailed ablation studies to dissect the contribution of individual components of our method and various configurations, thereby offering an in-depth analysis of our approach. Additionally, we provide detailed experiments on the NYUv2 dataset [58] to demonstrate the generalization ability of our SGN on indoor scenes.

A. Dataset and Metrics

1) Dataset: For large-scale outdoor scene understanding, the KITTI odometry dataset [50] collects 22 sequences with 20 classes with a Velodyne HDL-64 laser scanner in the scenes of autonomous driving. SemanticKITTI [14] is based on the KITTI dataset and provides semantic annotation of all sequences. According to the official setting for semantic scene completion (SSC), sequences 00-07 and 09-10 (a total of 3834 scans) are for training, sequence 08 (815 scans) is for validation, and the rest (3901 scans) is for testing. SSCBench-KITTI-360 [15] offers a comprehensive benchmark for semantic scene completion, featuring nine densely annotated sequences of urban driving scenes. The dataset is meticulously partitioned, with the training set encompassing 8,487 frames across scenes 00, 02-05, 07, and 10. The validation set is carefully curated with 1,812 frames from scene 06, ensuring a robust evaluation framework. Furthermore, the testing set includes 2,566 frames exclusively from scene 09, providing a diverse and challenging environment for model assessment. The volume of interest for both two SSC benchmarks is $[0 \sim 51.2m, -25.6m \sim 25.6m, -2 \sim 4.4m]$, and the voxelization resolution s is $0.2m$. The SSC labels with resolution $256 \times 256 \times 32$ of train and validation set are provided for the users. In this work, we focus on the camera-based SSC, taking the RGB images as inputs similar to [7], [12], and [30].

2) Metrics: Following [16], we mainly report the Intersection-over-Union (IoU) for scene completion and mIoU of C_n classes (no "unlabeled" class) for semantic scene completion. The mIoU is calculated by:

$$mIoU = \frac{1}{C_n} \sum_{c=1}^{C_n} \frac{TP_c}{TN_c + FP_c + FN_c} \tag{13}$$

where TP_c, TN_c, FP_c, and FN_c denote true positive, true negative, false positive, and false negative for class c.

B. Implementation Details

We crop the input RGB images of cam2 to size 1220 × 370 for SemanticKITTI and RGB images of cam1 of 1408 × 376 for SSCBench-KITTI-360. The 2D feature maps with 1/16 of the input resolution are taken for the subsequent processing. The feature dimension C and the channel number C_o are set to 128 and 8, respectively. The size $X \times Y \times Z$ of the 3D feature volume is $128 \times 128 \times 16$. And the final predictions are upsampled to $256 \times 256 \times 32$. The threshold θ for selecting seed voxels is set to 0.5 by default. We train SGN for 40 epochs on 4 V100 GPUs with a total batch size of 4. The AdamW [59] optimizer is used with an initial learning rate of 2e-4 and a weight decay of 1e-2. Following VoxFormer [12], we design the single-image version SGN-S, taking only the current image as input and the temporal version SGN-T with the current and the previous 4 images as input. We also provide the

TABLE II

SEMANTIC SCENE COMPLETION ON SEMANTICKITTI VAL SET. † DENOTES THE RESULTS PROVIDED BY MONOSCENE. BOLD AND UNDERLINE DENOTE THE BEST PERFORMANCE AND THE SECOND-BEST PERFORMANCE, RESPECTIVELY

Method	IoU	road (15.30%) 🟪	sidewalk (11.13%) 🟪	parking (1.12%) 🟪	otherground (0.56%) 🟥	building (14.1%) 🟨	car (3.92%) 🟦	truck (0.16%) 🟪	bicycle (0.03%) 🟦	motorcycle (0.03%) 🟦	othervehicle (0.20%) 🟪	vegetation (39.3%) 🟩	trunk (0.51%) 🟫	terrain (9.17%) 🟩	person (0.07%) 🟥	bicyclist (0.07%) 🟪	motorcyclist (0.05%) 🟪	fence (3.90%) 🟧	pole (0.29%) 🟨	trafficsign (0.08%) 🟥	mIoU
LMSCNet† [1]	28.61	40.68	18.22	4.38	0.00	10.31	18.33	0.00	0.00	0.00	0.00	13.66	0.02	20.54	0.00	0.00	0.00	1.21	0.00	0.00	6.70
AICNet† [13]	29.59	43.55	20.55	11.97	0.07	12.94	14.71	4.53	0.00	0.00	0.00	15.37	2.90	28.71	0.00	0.00	0.00	2.52	0.06	0.00	8.31
JS3C-Net† [4]	38.98	50.49	23.74	11.94	0.07	15.03	24.65	4.41	0.00	0.00	6.15	18.11	4.33	26.86	0.67	0.27	0.00	3.94	3.77	1.45	10.31
MonoScene [7]	37.12	57.47	27.05	15.72	0.87	14.24	23.55	7.83	0.20	0.77	3.59	18.12	2.57	30.76	1.79	1.03	0.00	6.39	4.11	2.48	11.50
TPVFormer [29]	35.61	56.50	25.87	20.60	0.85	13.88	23.81	8.08	0.36	0.05	4.35	16.92	2.26	30.38	0.51	0.89	0.00	5.94	3.14	1.52	11.36
VoxFormer [12]	44.02	54.76	26.35	15.50	0.70	17.65	25.79	5.63	0.59	0.51	3.77	24.39	5.08	29.96	1.78	**3.32**	0.00	7.64	7.11	4.18	12.35
OccFormer [10]	36.50	58.85	26.88	19.61	0.31	14.40	25.09	**25.53**	0.81	1.19	8.52	19.63	3.93	32.62	2.78	2.82	0.00	5.61	4.26	2.86	13.46
NDC-scene [32]	37.24	59.20	28.24	**21.42**	**1.67**	14.94	26.26	14.75	**1.67**	**2.37**	7.73	19.09	3.51	31.04	**3.60**	2.74	0.00	6.65	4.53	2.73	12.70
SGN-S (ours)	43.60	**59.32**	**30.51**	18.46	0.42	21.43	31.88	13.18	0.58	0.17	5.68	25.98	7.43	34.42	1.28	1.49	0.00	9.66	9.83	4.71	14.55
SGN-L (ours)	45.45	59.00	30.11	19.35	0.21	23.95	32.51	9.74	0.39	0.15	5.19	28.29	8.48	34.91	0.78	0.20	0.00	8.83	12.13	6.95	14.80
SGN-T (ours)	**46.21**	59.10	29.41	19.05	0.33	**25.17**	**33.31**	6.03	0.61	0.46	**9.84**	**28.93**	**9.58**	**38.12**	0.47	0.10	0.00	**9.96**	**13.25**	**7.32**	**15.32**

lightweight version SGN-L, which takes temporal inputs and uses ResNet18 as the backbone with dimension $C = 64$ and 1 anisotropic convolution layer for MSSP.

C. Comparison With the State-of-the-Art

1) SemanticKITTI: Table I and Table II present the comparison results between our SGN and other state-of-the-art camera-based SSC methods on the SemanticKITTI validation and test sets, respectively. Our SGN-T achieves state-of-the-art performance on both SemanticKITTI validation and test sets. Specifically, SGN-T outperforms the second one by 1.86% points (OccFormer) and 2.19% points (VoxFormer) regarding mIoU and IoU, as shown in Table II. And compared with these fully dense processing methods with complex 3D models, such as MonoScene and OccFormer, our SGN-S also performs better in terms of mIoU and IoU. For example, SGN-S greatly boosts the MonoScene by 3.05% points in mIoU and 6.48% points in IoU, demonstrating the effectiveness of our dense-sparse-dense design equipped with hybrid guidance. Notably, SGN-S outperforms the recent VoxFormer by 2.2% points in mIoU but has a slightly lower IoU (-0.42% points). We explain that VoxFormer adopted a two-stage training approach, and the first stage was trained offline with temporal inputs, helping enhance occupancy precision. However, our SGN-S is end-to-end trained with only a single frame as input. Compared to the two-stage VoxFormer, the higher mIoU score of our one-stage SGN-S for semantic scene completion demonstrates the superiority of our framework of semantic propagation based on spatial geometry cues.

Remarkably, our lightweight version SGN-L achieves notable performance (45.45% IoU and 14.80% mIoU) on SemanticKITTI validation with only **12.5M** parameters. Compared with MonoScene, OccFormer, and VoxFormer with ~150M, ~200M, and ~60M parameters, our SGN-L performs better while being more lightweight. It demonstrates that our SGN requires no heavy 3D model and has a more powerful representation ability.

TABLE III

QUANTITATIVE COMPARISON IN DIFFERENT RANGES ON SEMANTICKITTI VALIDATION. "*" DENOTES THE RESULTS PROVIDED BY VOXFORMER

Methods	Modality	IoU (%)			mIoU (%)		
		12.8m	25.6m	51.2m	12.8m	25.6m	51.2m
SSCNet [16]	LiDAR	64.37	61.02	50.22	20.02	19.68	16.35
JS3CNet [4]	LiDAR	63.47	**63.40**	**53.09**	**30.55**	**28.12**	**22.67**
MonoScene* [7]	Camera	38.42	38.55	36.80	12.25	12.22	11.30
OccFormer [10]	Camera	56.38	47.28	36.50	20.91	17.90	13.46
VoxFormer-S [12]	Camera	65.35	57.54	44.02	17.66	16.48	12.35
VoxFormer-T [12]	Camera	65.38	57.69	44.15	21.55	18.42	13.35
SGN-S (ours)	Camera	64.21	56.20	43.60	21.53	19.60	14.55
SGN-L (ours)	Camera	70.08	61.17	45.45	24.76	21.17	14.80
SGN-T (ours)	Camera	**70.61**	61.90	46.21	25.70	22.02	15.32

Quantitative Comparison in Different Ranges: We also provide the results of different ranges in Table III. The results show that SGN-T achieves mIoU scores of 25.70% and 22.02% within 12.8 meters and 25.6 meters and performs better than VoxFormer-T by 4.15% and 3.60% points in mIoU, respectively. Additionally, our SGN-S surpassed MonoScene by 9.28% and 7.38% points in mIoU within 12.8 meters and 25.6 meters. Notably, SGN-T obtains competitive performance with LiDAR-based methods in short-range (12.8 meters) areas. For example, SGN-T outperforms SSCNet by 5.68% points in mIoU and 6.24% points in IoU within 12.8 meters, demonstrating the potential application of our camera-based method for autonomous driving.

2) SSCBench-KITTI-360: Table IV presents the comparison results between our SGN and other state-of-the-art SSC methods, including LiDAR-based methods (SSCNet, LMSCNet) and camera-based methods (MonoScene, TPVFormer,

TABLE IV

QUANTITATIVE RESULTS ON SSCBENCH-KITTI360 TEST SET. THE RESULTS FOR COUNTERPARTS ARE PROVIDED IN [15]. BOLD AND UNDERLINE DENOTE THE BEST PERFORMANCE AND THE SECOND-BEST PERFORMANCE, RESPECTIVELY

Method	IoU	Precision	Recall	mIoU	car (2.85%) ■	bicycle (0.01%) ■	motorcycle (0.01%) ■	truck (0.16%) ■	other-veh. (5.75%) ■	person (0.02%) ■	road (14.98%) ■	parking (2.31%) ■	sidewalk (6.43%) ■	other-grnd. (2.05%) ■	building (15.67%) ■	fence (0.96%) ■	vegetation (41.99%) ■	terrain (7.10%) ■	pole (0.22%) ■	traf.-sign (0.06%) ■	other-struct. (4.33%) ■	other-obj. (0.28%) ■
LiDAR-based																						
SSCNet [16]	**53.58**	69.63	**69.92**	16.95	**31.95**	0.00	0.17	10.29	0.00	0.07	**65.70**	**17.33**	**41.24**	3.22	**44.41**	6.77	**43.72**	**28.87**	0.78	0.75	8.69	0.67
LMSCNet [1]	47.35	**72.77**	57.55	13.65	20.91	0.00	0.00	0.26	0.58	0.00	62.95	13.51	33.51	0.20	43.67	0.33	40.01	26.80	0.00	0.00	3.63	0.00
Camera-based																						
MonoScene [7]	37.87	56.73	53.26	12.31	19.34	0.43	0.58	8.02	2.03	0.86	48.35	11.38	28.13	3.32	32.89	3.53	26.15	16.75	6.92	5.67	4.20	3.09
TPVFormer [29]	40.22	59.32	55.54	13.64	21.56	1.09	1.37	8.06	2.57	2.38	52.99	11.99	31.07	3.78	34.83	4.80	30.08	17.52	7.46	5.86	5.48	2.70
VoxFormer [12]	38.76	58.52	53.44	11.91	17.84	1.16	0.89	4.56	2.06	1.63	47.01	9.67	27.21	2.89	31.18	4.97	28.99	14.69	6.51	6.92	3.79	2.43
OccFormer [10]	40.27	59.70	55.31	13.81	22.58	0.66	0.26	9.89	3.82	2.77	54.30	13.44	31.53	3.55	36.42	4.80	31.00	19.51	7.77	8.51	6.95	4.60
DepthSSC [56]	40.85	60.69	55.86	14.28	21.90	2.36	4.30	11.51	4.56	2.92	50.88	12.89	30.27	2.49	37.33	5.22	29.61	21.59	5.97	7.71	5.24	3.51
Symphonize [30]	44.12	69.24	54.88	**18.58**	30.02	1.85	5.90	**25.07**	**12.06**	**8.20**	54.94	13.83	32.76	**6.93**	35.11	**8.58**	38.33	11.52	14.01	9.57	**14.44**	**11.28**
SGN-S (ours)	46.22	68.17	58.94	17.71	28.20	2.09	3.02	11.95	3.68	4.20	59.49	14.50	36.53	4.24	39.79	7.14	36.61	23.10	14.86	16.14	8.24	4.95
SGN-L (ours)	46.64	68.26	59.55	16.95	26.91	1.72	0.85	8.60	3.80	1.93	56.52	13.83	35.40	3.42	40.62	6.65	36.68	22.00	15.84	**16.49**	8.06	5.76
SGN-T (ours)	47.06	68.83	59.81	18.25	29.03	**3.43**	2.90	10.89	5.20	2.99	58.14	15.04	36.40	4.43	42.02	7.72	38.17	23.22	**16.73**	16.38	9.93	5.86

VoxFormer, OccFormer, DepthSSC, Symphonize) on the SSCBench-KITTI-360 test sets. We can see that our SGN outperforms most camera-based methods by a large margin in terms of both mIou and IoU metrics. For example, SGN-S, SGN-L, and SGN-T surpass OccFormer by 5.95%, 6.37%, 6.79% points in IoU and 3.9%, 3.14% 4.44% points in mIoU. Compared to the LiDAR-based methods, our SGN-T also archives comparable performance in IoU and performs better in mIoU, demonstrating the superiority of our SGN. Interestingly, we found that our SGN-T achieves better performance on many *thing* classes such as traffic-sign, other-object, trucks, bicycles, motorcycles, and other vehicles while performing worse on plain *stuff* classes such as road, parking, sidewalk, building, and vegetation than LiDAR-based method SSCNet. We explain that the LiDAR point cloud contains more accurate structure information, which may facilitate the occupancy prediction of plain classes, while the vision feature includes more semantic information that helps distinguish objects that belong to different classes.

D. Qualitative Visualizations

We provide the visualization results of the proposed SGN-T on SemanticKITTI validation in Figure 4. Compared to VoxFormer-T and MonoScene, our SGN-T generates more precise segmentation boundaries, especially on "plane" classes and large objects such as cars. Besides, SGN-T predicts more accurate SSC results and preserves more regional details in the short-range areas than other methods. For example, there are some wrong semantics and missing objects for VoxFormer-T and MonoScene in the short-range areas, as shown in the first and third rows in Figure 4. However, we noticed that our SGN-T also missed some distant objects that are very small in RGB images. We explain that our SGN uses image features of the 1/16 scale, which may degrade the performance of objects in distant areas. We also provide qualitative results of our SGN-S, SGN-L, and SGN-T on SemanticKITTI hidden test set in Figure 6. Our method can provide accurate semantic occupancy prediction and road layout and handle typical driving scenes such as crowded cars, shadows, tiny poles, and crossroads.

TABLE V

ABLATION ON NETWORK COMPONENTS, I.E., SEMANTIC GUIDANCE (SG), GEOMETRY GUIDANCE (GG), MULTI-SCALE SEMANTIC PROPAGATION (MSSP), AND VOXEL AGGREGATION LAYER (VA)

Variants	MSSP	SG	GG	VA		IoU (%)	mIoU (%)
				GP	OA		
baseline						41.76	10.62
1	✓					43.22	13.00
2	✓	✓				43.32	13.68
3	✓	✓	✓			43.45	13.44
4	✓	✓	✓	✓		43.14	14.39
5	✓	✓	✓	✓	✓	**43.60**	**14.55**

E. Ablation Studies

We do ablation studies on network components, training mode, depth estimator, image features, view transformation, seed voxels, model dimensions, and temporal input on SemanticKITTI validation. All experiments are conducted with our SGN-S by default.

1) Ablation on Network Components: We do ablation studies to analyze the effect of the proposed semantic guidance (SG), geometry guidance (GG), multi-scale semantic propagation (MSSP), and voxel aggregation layer (VA) in Table V. GP and OA denote geometry information from 3D features $\mathbf{F}^{3D}$ and occupancy-aware features $\mathbf{F}_o^{3D}$, respectively. Firstly, we construct a baseline that directly attaches a segmentation head after the selected seed features. As shown in the first line of Table V, the constructed baseline has already achieved 41.76% IoU and 10.62% mIoU scores, demonstrating our

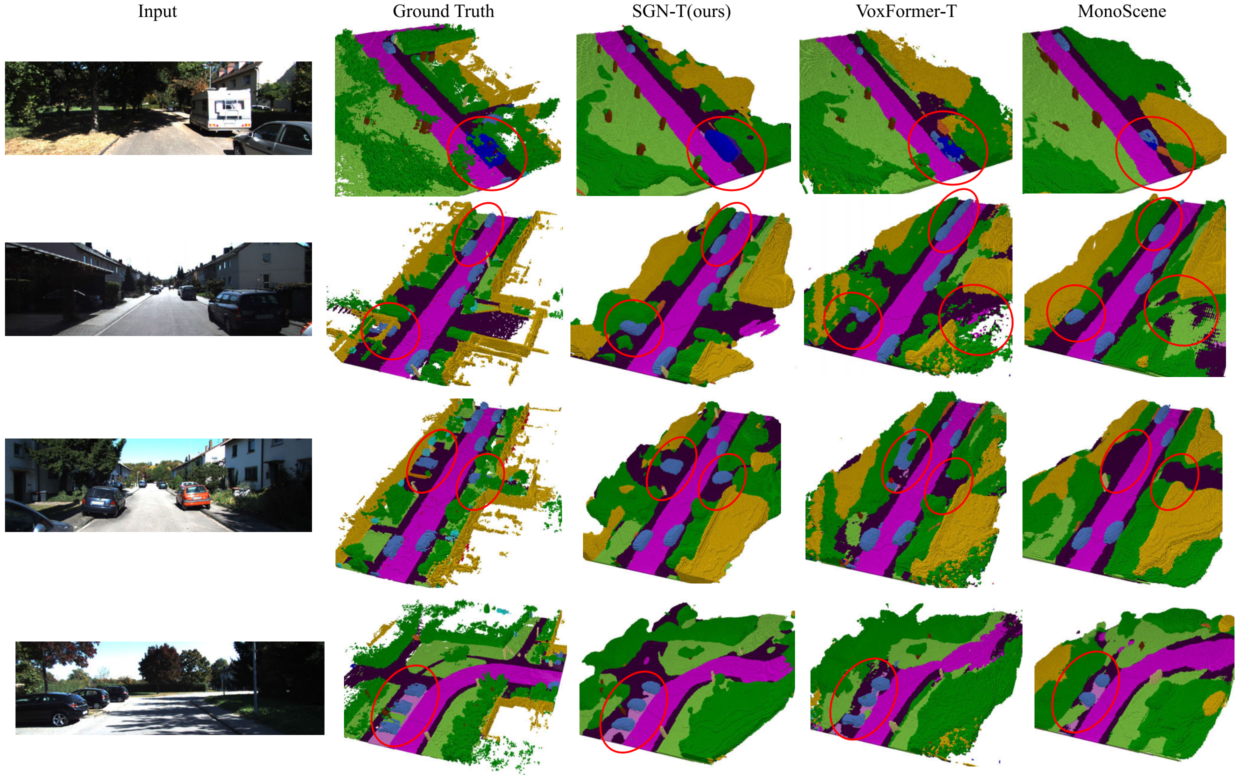

Fig. 4. Visual comparison of our SGN-T with state-of-the-art methods on SemanticKITTI validation. Compared to VoxFormer-T and MonoScene, our SGN-T generates more precise segmentation boundaries (labeled in red circles).

depth-based sparse voxel proposal network can provide effective seed voxels. When equipped with our MSSP (Variant 1), the IoU and mIoU scores are improved by 1.46% points and 2.38% points, respectively. It demonstrates the effectiveness of multi-scale information propagation. And our semantic guidance on the seed features further boosts the mIoU by 0.68% points (Variant 2 vs. Variant 1), showing the importance of intra-category separation of seed features. On the other hand, the geometry guidance brings slight improvement in terms of IoU score, while the geometry prior further boosts the mIoU score by 0.95% points (Variant 4 vs. Variant 3). Besides, introducing geometry information in features F_o^{3D} can help improve performance (Variant 5 vs Variant 4). And comparing Variant 5 with Variant 1, the mIoU score is significantly boosted (+1.55% points), demonstrating the effectiveness of our semantic propagation based on spatial geometry cues.

2) Impact of Different Training Modes: To explore the effect of different training modes, i.e., two-stage and one-stage training, we provide the detailed comparison results with VoxFormer in Table VI. Line 3 of Table VI presents the results of our SGN-S with the two-stage training strategy. Note that when equipping SGN-S with a two-stage approach, the first stage remains the same as VoxFormer, and the parameters and training memory of the second stage are calculated for a fair comparison. In the same training configuration, Our SGN-S surpasses VoxFormer-S by a large margin regarding mIoU scores (+2.58% points). Even our one-stage SGN-S outperforms two-stage VoxFormer-S by 2.2% points in mIoU. It is worth noting that the model parameters of our SGN-S are only about half of those of VoxFormer-S. For the temporal version, our SGN-T achieves 46.21 IoU and 15.32 mIoU scores, boosting VoxFormer-T by 2.06% points in IoU and 1.97% points in mIoU. Our lightweight version, SGN-L, with only 12.5M parameters, also outperforms VoxFormer-T on mIoU and IoU scores while requiring only about half the training memory (7.16 G).

TABLE VI
IMPACT OF DIFFERENT TRAINING MODES. OUR ONE-STAGE SGN SURPASSES THE TWO-STAGE VOXFORMER BY A LARGE MARGIN. MEMORY DENOTES TRAINING MEMORY

Methods	Mode	IoU (%)	mIoU (%)	Params (M)	Memory (G)
VoxFormer-S	two-stage	44.02	12.35	57.90	14.41
VoxFormer-T	two-state	44.15	13.35	57.90	16.38
SGN-S (ours)	two-stage	**44.76**	**14.93**	**27.79**	**10.92**
SGN-S (ours)	one-stage	43.60	14.55	28.16	14.21
SGN-L (ours)	one-stage	45.45	14.80	**12.50**	**7.16**
SGN-T (ours)	one-stage	**46.21**	**15.32**	28.16	15.83

3) Ablation on Depth Estimator: Our sparse voxel proposal network produces the seed voxels based on the depth map predicted by the depth estimator. The generated depth map contains 3D structure information, such as volume surfaces, which has a direct impact on the seed voxel proposal. To quantitatively analyze the impact of the depth estimator, we compare our SGN equipped with the monocular-based AdaBins [60] and stereo-based MobileStereoNet [53] with VoxFormer [12]. The results are presented in Table VII and show that using the stereo-based depth estimation brought significant performance improvements for both VoxFormer

TABLE VII

ABLATION STUDY FOR DEPTH ESTIMATOR. MONO AND STEREO DENOTE USING MONOCULAR-BASED ADABINS [60] AND STEREO-BASED MOBILESTEREONET [53] AS THE DEPTH ESTIMATOR

Methods	Depth	IoU (%)			mIoU (%)		
		12.8m	25.6m	51.2m	12.8m	25.6m	51.2m
VoxFormer-S	Mono	57.41	50.61	38.68	14.62	14.01	10.67
	Stereo	65.35	57.54	44.02	17.66	16.48	12.35
VoxFormer-T	Mono	59.03	50.47	38.08	18.67	15.42	11.27
	Stereo	65.38	57.69	44.15	21.55	18.42	13.35
SGN-S (ours)	Mono	59.82	51.43	39.35	18.13	16.65	12.51
	Stereo	64.21	56.20	43.60	21.53	19.60	14.55
SGN-L (ours)	Mono	63.79	54.99	41.36	20.77	17.69	12.64
	Stereo	70.08	61.17	45.45	24.76	21.17	14.80
SGN-T (ours)	Mono	64.74	55.55	41.87	21.43	17.94	12.91
	Stereo	**70.61**	**61.90**	**46.21**	**25.70**	**22.02**	**15.32**

TABLE VIII

ABLATION ON VIEW TRANSFORMATION. USING FLOSP CONTAINS FEWER PARAMETERS WHILE ACHIEVING COMPARABLE PERFORMANCE TO OTHER VARIANTS. MEMORY DENOTES TRAINING MEMORY

Module	IoU (%)	mIoU (%)	Params (M)	Memory (G)
FLoSP	43.60	14.55	**28.16**	14.21
LSS	42.95	**14.77**	30.56	**13.23**
Cross-Attention	**43.66**	14.05	62.15	19.01

TABLE IX

ABLATION ON IMAGE FEATURES, INCLUDING THE IMAGE BACKBONE AND SCALES. USING SCALE 16 STRIKES A BALANCE BETWEEN PERFORMANCE AND MODEL PARAMETERS

Backbone	scales				IoU (%)	mIoU (%)	Params (M)
	4	8	16	32			
ResNet50	✓				43.10	13.85	28.06
		✓			42.92	14.26	28.09
			✓		**43.60**	14.55	28.16
				✓	43.36	14.20	28.29
	✓	✓	✓	✓	42.86	**14.60**	28.96
ResNet18			✓		43.57	14.08	**15.73**

and our SGN, which means a stronger depth estimator that produces more accurate depth maps can boost the performance further. Notably, in the same configurations, our SGN consistently surpasses VoxFormer in all different ranges, demonstrating the effectiveness and superiority of our approaches.

4) Impact of Image Features: The 2D features provide a foundation for informative voxel features. We do detailed experiments in Table IX to explore the impact of the feature scale and image backbone. We see that using 2D features at a scale of 1/16 in ResNet 50 achieves the best IoU score and has comparable mIoU to other variants. And it strikes a balance between performance and model size. The results of using ResNet18 as the backbone are presented in the last line of Table IX and show that a more lightweight image backbone reduces the model parameters by 12.43M while the performance of the mIoU score drops by 0.47% points.

5) Ablation on View Transformation: The view transformation generates the initial 3D features for the subsequent hybrid guidance and informative voxel features. We further investigate the effect of different view transformation modules.

TABLE X

NUMBER OF MODEL DIMENSIONS AND DEPTH OF MSSP. MEMORY DENOTES TRAINING MEMORY

Dimensions	Depth	IoU (%)	mIoU (%)	Params (M)	Memory (G)
64	1	43.16	**14.24**	**24.88**	**7.23**
64	2	**43.73**	14.17	24.93	7.27
64	3	43.32	14.20	24.97	8.04
128	3	**43.60**	**14.55**	28.16	14.21

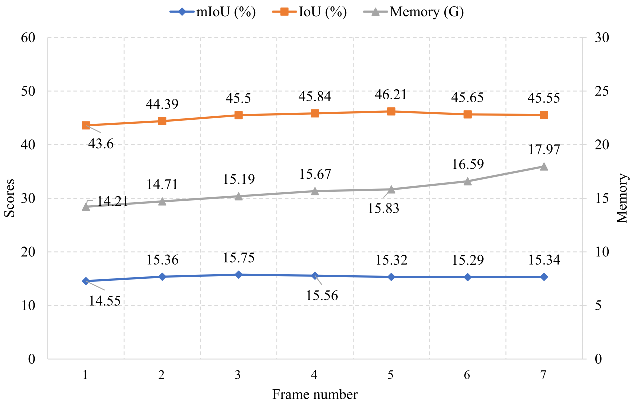

Fig. 5. Effect of temporal frames. The frames are sampled every three frames. Memory denotes training memory.

We implement three commonly used modules, i.e., FLoSP in Monoscene [7], LSS [11], and cross-attention adopted from VoxFormer [12]. The results are presented in Table VIII, showing that using FLoSP contains fewer parameters while achieving comparable performance to other variants on both mIoU and IoU scores. Therefore, our SGN lifts the 2D features to 3D volume with the view transformation designed in the spirit of FLoSP.

6) Exploration on the Threshold for Seed Voxels: We change the value θ to investigate the impact of different thresholds for selecting seed voxels. We calculate the average occupancy rate of seed voxels, mIoU score, and IoU score for variants with θ from 0.1 to 0.9. The results are shown in Figure 7, showing that the performance of the model first increases and then decreases as θ increases. And when $\theta = 0.4$, the model achieves the best performance on both mIoU and IoU scores. Interestingly, we found our model still achieves notable mIoU and IoU scores when the ratio of the seed voxels is very low ($< 5\%$ points). It shows that seed voxels with high confidence play an essential role in our semantic propagation.

7) Number of Model Dimensions: The impact of the number of model dimensions of our SGN-S is evaluated and presented in Table X. The results reveal that using a large number of feature channels for 3D features boosts the models' performance while increasing the model complexity. For example, the model with 64 dimensions contains fewer parameters and requires less training memory, although its performance drops by 0.28% points in IoU and 0.35% points in mIoU (Line3 vs Line4). We also provide the results of different model depths for the multi-scale semantic propagation module. We see that using different depths has comparable mIoU and IoU, demonstrating that our model with hybrid guidance and

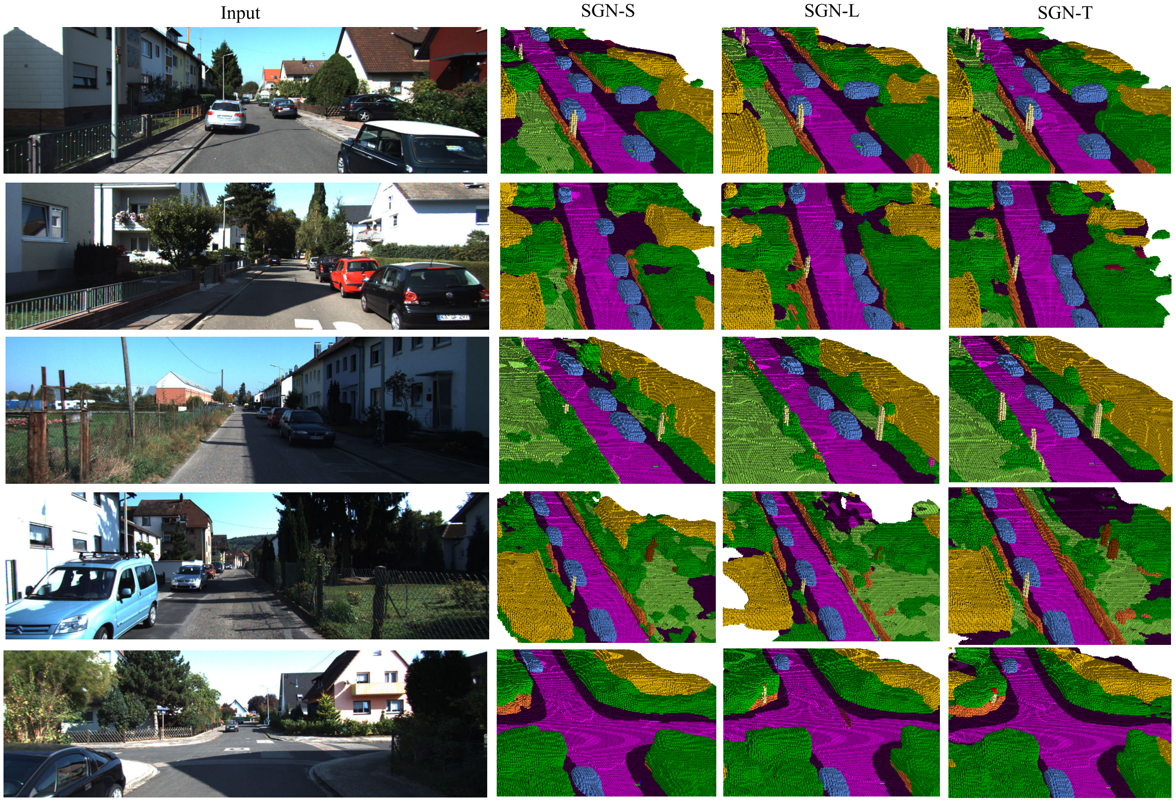

Fig. 6. Qualitative results of our SGN-S, SGN-L, and SGN-T on SemanticKITTI hidden test set. Our method can provide accurate semantic occupancy prediction and handle typical driving scenes such as crowded cars, shadows, tiny poles, and crossroads.

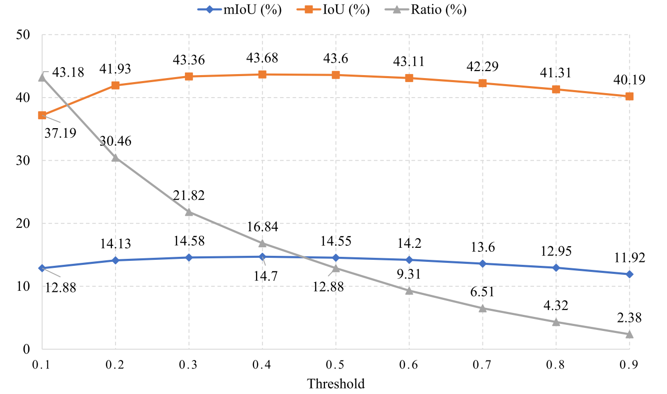

Fig. 7. Impact of threshold values for seed voxels. The performance of the model first increases and then decreases as θ increases. And when $\theta = 0.4$, the model achieves the best performance on both mIoU and IoU scores.

semantic propagation avoids the dependency on the heavy 3D model for processing 3D features.

8) Effect of Temporal Input: Finally, we explore the impact of the number of temporal frames in Figure 5. We take historical frames to form the temporal input. As Figure 5 shows, the model's performance on IoU scores first increases with the number of frames and then decreases. We explain that the camera extrinsic matrix from the historical frame to the current system is not always accurate, especially when the temporal interval is long. Therefore, the 3D points may project on the wrong image patches of the historical frames, which disturbs the learning of voxel features. To better balance the performance and memory consumption, our temporal version takes five frames (four past frames and the current frame).

F. Efficiency Analysis

We perform runtime experiments on a single V100 GPU. The mean value over the SemanticKITTI test set is reported. Our SGN-S, SGN-L, and SGN-T run in 327.71 ms, 315.35 ms, and 436.24 ms, respectively. We also tested the recent VoxFormer-T (261.46 ms, ~60M parameters) and OccFormer (322.87 ms, ~200M parameters) on the same platform with the officially provided weights for a fair comparison. Compared with these methods, our lightweight version SGN-L achieves better mIoU and IoU scores and comparable latency, but with better applicability due to its lightweight (12.5M parameters) and less training memory consumption (7.16 G).

G. Generalization on Indoor Scenes

Although our proposed SGN mainly focuses on the outdoor driving scene as mentioned in Section I, we further provide detailed experiments on the NYUv2 [58] dataset to demonstrate the generalization ability on indoor scenes. NYUv2 comprises 1449 indoor scenes, represented as $240 \times 144 \times 240$ voxel grids labeled with 13 classes (11 semantics, 1 free, 1 unknown). The input resolution is 640×480. Following [7] and [32], we utilize a train/test splits of 795/654 scenes to perform evaluations on the test set at the scale of 1:4. Consistent with MonoScene [7], to verify the effectiveness on the indoor scenes, we utilize our single-image version SGN-S with the pre-trained EfficientNetB7 [62] as the image encoder and change the size $X \times Y \times Z$ to $60 \times 36 \times 60$. The 2D feature maps with 1/8 of the input resolution are taken for the subsequent processing. We apply [63] to generate monocular

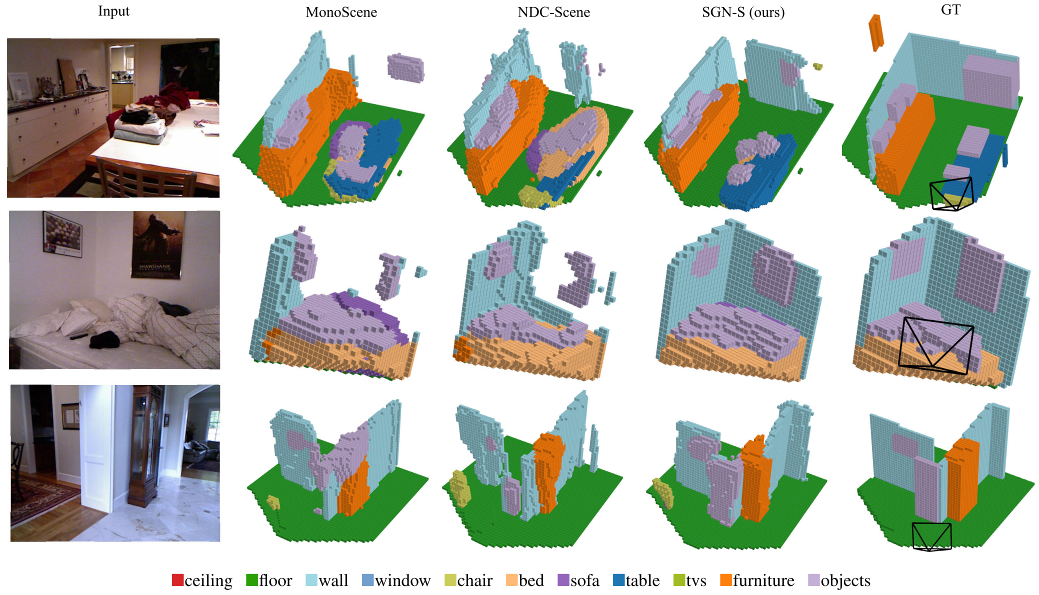

Fig. 8. Qualitative results of our SGN-S on NYUv2 test set. Our method can provide accurate semantic occupancy prediction, demonstrating the generalization ability on the indoor scenes.

TABLE XI

Semantic Scene Completion on NYUv2 Test Set. These Compared Methods Are Copy From MonoScene [7] and NDC-Scene [32]. Bold and Underline Denote the Best Performance and the Second-Best Performance, Respectively

Method	IoU	ceiling (1.37%) 🟥	floor (17.58%) 🟩	wall (15.26%) 🟦	window (1.99%) 🟦	chair (3.01%) 🟨	bed (7.08%) 🟧	sofa (4.70%) 🟪	table (4.31%) 🟦	tvs (0.47%) 🟩	furniture (30.04%) 🟧	objects (14.19%) 🟪	mIoU
LMSCNet^{rgb} [1]	33.93	4.49	88.41	4.63	0.25	3.94	32.03	15.44	6.57	0.02	14.51	4.39	15.88
AICNet^{rgb} [13]	30.03	7.58	82.97	9.15	0.05	6.93	35.87	22.92	11.11	0.71	15.90	6.45	18.15
3DSketch^{rgb} [61]	38.64	8.53	90.45	9.94	5.67	10.64	42.29	29.21	13.88	9.38	23.83	8.19	22.91
MonoScene [7]	42.51	8.89	93.50	12.06	<u>12.57</u>	13.72	48.19	36.11	<u>15.13</u>	15.22	27.96	<u>12.94</u>	26.94
NDC-Scene [32]	44.17	12.02	93.51	13.11	**13.77**	**15.83**	**49.57**	**39.87**	**17.17**	**24.57**	**31.00**	**14.96**	**29.03**
SGN-S (ours)	**44.85**	**14.67**	**93.56**	**13.36**	10.35	<u>14.64</u>	<u>48.59</u>	<u>37.47</u>	14.08	<u>15.75</u>	<u>30.31</u>	12.07	<u>27.71</u>

depth prediction and train SGN-S for 30 epochs using the AdamW [59] optimizer with the initial learning rate of 2e-4 and a weight decay of 1e-3.

As shown in Table XI, without any bells and whistles, our SGN-S achieves the best performance in terms of IoU score and delivers comparable results in mIoU score, which demonstrates the generalization ability of our proposed method on indoor scenes. For instance, SGN-S surpasses MonoScene by 0.77% and 2.34% points in mIoU and IoU scores, respectively. However, we observe that NDC-Scene outperforms SGN-S in mIoU score. We attribute this to the complexity and sensitivity of indoor scenes to the accuracy of the depth estimator. Additionally, SGN-S uses an image scale of only 1/8, which may impact the segmentation details, resulting in worse performance on some categories such as “TVs” and “objects,” as shown in Table XI. We believe that incorporating a more accurate depth estimator, more effective multi-scale feature fusion, and advanced loss designs, such as the frustum proportion loss used in [7], could further enhance performance. This will be a focus of our future work. We also present the qualitative results of our method and recent methods on the NYUv2 test set. As illustrated in Figure 8, even without a special design for the indoor scenarios, our SGN-S still generates precise semantic scene completion prediction, which further demonstrates the generalization ability of our method.

V. Conclusion

This work focuses on camera-based semantic scene completion (SSC). Existing methods usually rely on sophisticated 3D models to directly process the coarse lifted 3D features that are not discriminative enough for clear segmentation boundaries. Therefore, we propose the one-stage SGN to propagate semantics from the semantic-aware seed voxels to the whole scene based on spatial geometry information. We first redesign the sparse voxel proposal network with the coarse-to-fine paradigm for dynamically and accurately selecting seed voxels. Then, we design hybrid guidance and effective voxel aggregation to enhance the intra-category feature separations and expedite the convergence of semantic propagation. Finally, the multi-scale semantic propagation is proposed for the final semantic scene completion. Extensive experiments on the SemanticKITTI and SSCBench-KITTI-360 benchmarks demonstrate the effectiveness of Our SGN, which achieves state-of-the-art performance while being more lightweight.

We hope our work can promote the exploration of model optimization and lightweighting in 3D scene understanding and provide innovative solutions for applications in scenarios with limited resources.

References

[1] L. Roldão, R. de Charette, and A. Verroust-Blondet, "LMSCNet: Lightweight multiscale 3D semantic completion," in *Proc. Int. Conf. 3D Vis. (3DV)*, Nov. 2020, pp. 111–119.

[2] R. Cheng, C. Agia, Y. Ren, X. Li, and B. Liu, "S3CNet: A sparse semantic scene completion network for LiDAR point clouds," in *Proc. Conf. Robot Learn.*, Nov. 2021, pp. 2148–2161.

[3] C. B. Rist, D. Emmerichs, M. Enzweiler, and D. M. Gavrila, "Semantic scene completion using local deep implicit functions on LiDAR data," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, pp. 7205–7218, Oct. 2022.

[4] Y. Xu et al., "Sparse single sweep LiDAR point cloud segmentation via learning contextual shape priors from scene completion," in *Proc. AAAI Conf. Artif. Intell.*, 2021, vol. 35, no. 4, pp. 3101–3109.

[5] Z. Xia et al., "SCPNet: Semantic scene completion on point cloud," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 17642–17651.

[6] P. Li et al., "LODE: Locally conditioned Eikonal implicit scene completion from sparse LiDAR," 2023, *arXiv:2302.14052*.

[7] A.-Q. Cao and R. de Charette, "MonoScene: Monocular 3D semantic scene completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 3991–4001.

[8] R. Miao et al., "OccDepth: A depth-aware method for 3D semantic scene completion," 2023, *arXiv:2302.13540*.

[9] Y. Wei, L. Zhao, W. Zheng, Z. Zhu, J. Zhou, and J. Lu, "SurroundOcc: Multi-camera 3D occupancy prediction for autonomous driving," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2023, pp. 21729–21740.

[10] Y. Zhang, Z. Zhu, and D. Du, "OccFormer: Dual-path transformer for vision-based 3D semantic occupancy prediction," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 9433–9443.

[11] J. Philion and S. Fidler, "Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3D," in *Proc. 16th Eur. Conf.*, Glasgow, U.K. Springer, Aug. 2020, pp. 194–210.

[12] Y. Li et al., "VoxFormer: Sparse voxel transformer for camera-based 3D semantic scene completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 9087–9098.

[13] J. Li, K. Han, P. Wang, Y. Liu, and X. Yuan, "Anisotropic convolutional networks for 3D semantic scene completion," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3348–3356.

[14] J. Behley et al., "SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 9297–9307.

[15] Y. Li et al., "SSCBench: Monocular 3D semantic scene completion benchmark in street views," 2023, *arXiv:2306.09001*.

[16] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser, "Semantic scene completion from a single depth image," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 190–198.

[17] H. Zou et al., "Up-to-down network: Fusing multi-scale context for 3D semantic scene completion," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2021, pp. 16–23.

[18] J. Zhang, H. Zhao, A. Yao, Y. Chen, L. Zhang, and H. Liao, "Efficient semantic scene completion network with spatial group convolution," in *Proc. IEEE Eur. Conf. Comput. Vis. (ECCV)*, Sep. 2018, pp. 733–749.

[19] X. Yang et al., "Semantic segmentation-assisted scene completion for LiDAR point clouds," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2021, pp. 3555–3562.

[20] J. Mei, Y. Yang, M. Wang, T. Huang, X. Yang, and Y. Liu, "SSC-RS: Elevate LiDAR semantic scene completion with representation separation and BEV fusion," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2023, pp. 1–8.

[21] J. Huang, G. Huang, Z. Zhu, Y. Ye, and D. Du, "BEVDet: High-performance multi-camera 3D object detection in bird-eye-view," 2021, *arXiv:2112.11790*.

[22] Y. Wang, V. C. Guizilini, T. Zhang, Y. Wang, H. Zhao, and J. Solomon, "DETR3D: 3D object detection from multi-view images via 3D-to-2D queries," in *Proc. Conf. Robot Learn.*, 2022, pp. 180–191.

[23] Y. Liu, T. Wang, X. Zhang, and J. Sun, "PETR: Position embedding transformation for multi-view 3D object detection," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2022, pp. 531–548.

[24] Z. Li et al., "Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2022, pp. 1–18.

[25] T. Wang, X. Zhu, J. Pang, and D. Lin, "FCOS3D: Fully convolutional one-stage monocular 3D object detection," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. Workshops (ICCVW)*, Oct. 2021, pp. 913–922.

[26] S. Chen, X. Wang, T. Cheng, Q. Zhang, C. Huang, and W. Liu, "Polar parametrization for vision-based surround-view 3D detection," 2022, *arXiv:2206.10965*.

[27] B. Zhou and P. Krähenbühl, "Cross-view transformers for real-time map-view semantic segmentation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 13760–13769.

[28] S. Chen, T. Cheng, X. Wang, W. Meng, Q. Zhang, and W. Liu, "Efficient and robust 2D-to-BEV representation learning via geometry-guided kernel transformer," 2022, *arXiv:2206.04584*.

[29] Y. Huang, W. Zheng, Y. Zhang, J. Zhou, and J. Lu, "Tri-perspective view for vision-based 3D semantic occupancy prediction," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2023, pp. 9223–9232.

[30] H. Jiang et al., "Symphonize 3D semantic scene completion with contextual instance queries," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2024, pp. 20258–20267.

[31] B. Li et al., "Bridging stereo geometry and BEV representation with reliable mutual interaction for semantic scene completion," 2023, *arXiv:2303.13959*.

[32] J. Yao et al., "NDC-scene: Boost monocular 3D semantic scene completion in normalized device coordinates space," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2023, pp. 9421–9431.

[33] W. Gan, N. Mo, H. Xu, and N. Yokoya, "A simple framework for 3D occupancy estimation in autonomous driving," 2023, *arXiv:2303.10076*.

[34] Z. Li et al., "FB-OCC: 3D occupancy prediction based on forward-backward view transformation," 2023, *arXiv:2307.01492*.

[35] X. Tian et al., "Occ3D: A large-scale 3D occupancy prediction benchmark for autonomous driving," 2023, *arXiv:2304.14365*.

[36] X. Wang et al., "OpenOccupancy: A large scale benchmark for surrounding semantic occupancy perception," 2023, *arXiv:2303.03991*.

[37] W. Bao, B. Xu, and Z. Chen, "MonoFENet: Monocular 3D object detection with feature enhancement networks," *IEEE Trans. Image Process.*, vol. 29, pp. 2753–2765, 2020.

[38] C. Huang, T. He, H. Ren, W. Wang, B. Lin, and D. Cai, "OBMO: One bounding box multiple objects for monocular 3D object detection," *IEEE Trans. Image Process.*, vol. 32, pp. 6570–6581, 2023.

[39] F. Cai, H. Chen, and L. Deng, "CI3D: Context interaction for dynamic objects and static map elements in 3D driving scenes," *IEEE Trans. Image Process.*, vol. 33, pp. 2867–2879, 2024.

[40] W. Zhang, Y. Zhang, R. Song, Y. Liu, and W. Zhang, "3D layout estimation via weakly supervised learning of plane parameters from 2D segmentation," *IEEE Trans. Image Process.*, vol. 31, pp. 868–879, 2022.

[41] Y. Li et al., "BEVDepth: Acquisition of reliable depth for multi-view 3D object detection," in *Proc. AAAI Conf. Artif. Intell.*, Jun. 2023, vol. 37, no. 2, pp. 1477–1485.

[42] L. Peng, Z. Chen, Z. Fu, P. Liang, and E. Cheng, "BEVSegFormer: Bird's eye view semantic segmentation from arbitrary camera rigs," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2023, pp. 5935–5943.

[43] Y. Zhang et al., "BEVerse: Unified perception and prediction in birds-eye-view for vision-centric autonomous driving," 2022, *arXiv:2205.09743*.

[44] A. Hu et al., "FIERY: Future instance prediction in bird's-eye view from surround monocular cameras," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 15273–15282.

[45] T. Roddick, A. Kendall, and R. Cipolla, "Orthographic feature transform for monocular 3D object detection," 2018, *arXiv:1811.08188*.

[46] Z. Liu et al., "BEVFusion: Multi-task multi-sensor fusion with unified bird's-eye view representation," 2022, *arXiv:2205.13542*.

[47] J. Mei et al., "CenterLPS: Segment instances by centers for LiDAR panoptic segmentation," in *Proc. 31st ACM Int. Conf. Multimedia*, Oct. 2023, pp. 1884–1894.

[48] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 770–778.

[49] T. Y. Lin, P. Dollàr, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jul. 2017, pp. 2117–2125.

[50] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2012, pp. 3354–3361.

[51] H. Zhou et al., "Cylinder3D: An effective 3D framework for driving-scene LiDAR semantic segmentation," 2020, *arXiv:2008.01550*.

[52] M. Ye, R. Wan, S. Xu, T. Cao, and Q. Chen, "Efficient point cloud segmentation with geometry-aware sparse networks," in *Proc. 17th Eur. Conf.*, Tel Aviv, Israel. Cham, Switzerland: Springer, 2022, pp. 196–212.

[53] F. Shamsafar, S. Woerz, R. Rahim, and A. Zell, "MobileStereoNet: Towards lightweight deep networks for stereo matching," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2022, pp. 677–686.

[54] M. Ye, S. Xu, T. Cao, and Q. Chen, "DRINet: A dual-representation iterative learning network for point cloud segmentation," in *Proc. IEEE/CVF Int. Conf. Comput. Vis. (ICCV)*, Oct. 2021, pp. 7427–7436.

[55] M. Berman, A. R. Triki, and M. B. Blaschko, "The Lovasz-Softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 4413–4421.

[56] J. Yao and J. Zhang, "DepthSSC: Depth-spatial alignment and dynamic voxel resolution for monocular 3D semantic scene completion," 2023, *arXiv:2311.17084*.

[57] L. C. Chen, G. Papandreou, and I. Kokkinos, "DeepLab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 40, no. 4, pp. 834–848, Jun. 2017.

[58] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus, "Indoor segmentation and support inference from RGBD images," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 2012, pp. 746–760.

[59] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," 2017, *arXiv:1711.05101*.

[60] S. F. Bhat, I. Alhashim, and P. Wonka, "AdaBins: Depth estimation using adaptive bins," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jan. 2021, pp. 4009–4018.

[61] X. Chen, K.-Y. Lin, C. Qian, G. Zeng, and H. Li, "3D sketch-aware semantic scene completion via semi-supervised structure prior," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 4192–4201.

[62] M. Tan and Q. V. E. Le, "Rethinking model scaling for convolutional neural networks," 2019, *arXiv:1905.11946*.

[63] L. Yang, B. Kang, Z. Huang, X. Xu, J. Feng, and H. Zhao, "Depth anything: Unleashing the power of large-scale unlabeled data," 2024, *arXiv:2401.10891*.

Jianbiao Mei received the B.S. degree in control science and engineering from Zhejiang University, Zhejiang, China, in 2021, where he is currently pursuing the Ph.D. degree with the Laboratory of Advanced Perception on Robotics and Intelligent Learning, College of Control Science and Engineering. His research interests include video segmentation, 3D perception, and autonomous driving.

Yu Yang received the B.S. degree in control science and engineering from China University of Geosciences, Hubei, China, in 2021. He is currently pursuing the Ph.D. degree with the Laboratory of Advanced Perception on Robotics and Intelligent Learning, College of Control Science and Engineering, Zhejiang University, Zhejiang, China. His research interests include 3D perception and autonomous driving.

Mengmeng Wang received the B.S., M.S., and Ph.D. degrees in control science and engineering from Zhejiang University, Zhejiang, China, in 2015, 2018, and 2024, respectively. Her research interests include visual tracking, action recognition, computer vision, and deep learning.

Junyu Zhu (Member, IEEE) received the B.S. degree in automation from Wuhan University, Hubei, China, in 2021, and the M.S. degree in control science and engineering from Zhejiang University, Zhejiang, China, in 2024. His research interests include depth estimation and autonomous driving.

Jongwon Ra received the B.S. and M.S. degrees in control science and engineering from Zhejiang University, Zhejiang, China, in 2021 and 2024, respectively. His research interests include visual tracking, computer vision, and deep learning.

Yukai Ma received the B.Eng. degree in electrical engineering and its automation from Zhejiang University of Technology in 2021. He is currently pursuing the Ph.D. degree with the Laboratory of Advanced Perception on Robotics and Intelligent Learning, College of Control Science and Engineering, Zhejiang University. His research interests include deep learning in sensor fusion and SLAM.

Laijian Li received the B.S. degree from the College of Information Engineering, Zhejiang University of Technology, Zhejiang, China, in 2021, and the M.S. degree in control science and engineering from Zhejiang University, Zhejiang, in 2024. His research interests include deep learning, robot localization, and autonomous driving.

Yong Liu received the B.S. degree in computer science and engineering and the Ph.D. degree in computer science from Zhejiang University, Zhejiang, China, in 2001 and 2007, respectively. He is currently a Professor with the Institute of Cyber-Systems and Control, Zhejiang University. His research interests include robot perception and vision, deep learning, big data analysis, multi-sensor fusion, machine learning, computer vision, information fusion, and robotics.