IEEE International Conference on Systems, Man, and Cybernetics (SMC), Vienna, Austria, 2025.

SWA-SOP: Spatially-aware Window Attention for Semantic Occupancy Prediction in Autonomous Driving

Helin Cao*[1,2,3], Rafael Materla*[1], and Sven Behnke[1,2,3]

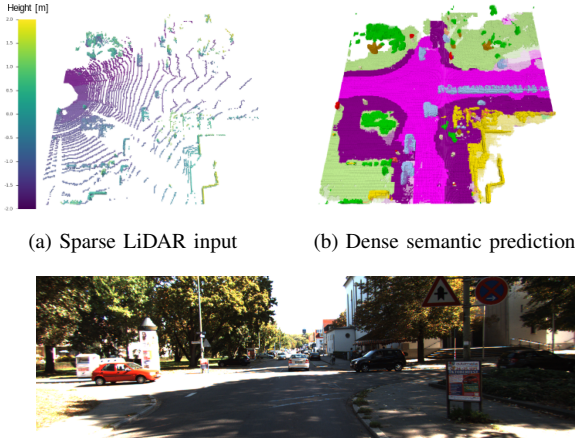
***Abstract*— Perception systems in autonomous driving rely on sensors such as LiDAR and cameras to perceive the 3D environment. However, due to occlusions and data sparsity, these sensors often fail to capture complete information. Semantic Occupancy Prediction (SOP) addresses this challenge by inferring both occupancy and semantics of unobserved regions. Existing transformer-based SOP methods lack explicit modeling of spatial structure in attention computation, resulting in limited geometric awareness and poor performance in sparse or occluded areas. To this end, we propose Spatially-aware Window Attention (SWA), a novel mechanism that incorporates local spatial context into attention. SWA significantly improves scene completion and achieves state-of-the-art results on LiDAR-based SOP benchmarks. We further validate its generality by integrating SWA into a camera-based SOP pipeline, where it also yields consistent gains across modalities.**

I. Introduction

Autonomous vehicles rely on sensors such as LiDAR and cameras to perceive their surroundings. However, these sensors are inherently constrained: Their measurements into 3D space are sparse and occlusions from nearby objects often block critical parts of the scene. These limitations result in incomplete geometric and appearance observations, which can degrade perception reliability and hinder safe operation in complex traffic environments. To support robust decision-making and ensure driving safety, it is crucial to infer missing information and reconstruct a more complete understanding of the scene.

Semantic Occupancy Prediction (SOP) addresses this challenge by estimating a dense 3D voxel grid with semantic labels from a single-frame input, such as an image or a LiDAR scan. The objective is to predict both occupancy and semantic categories for all voxels, including those in occluded or unobserved regions. By transforming sparse, partial sensor observations into a structured and semantically rich 3D representation, SOP enables a more comprehensive understanding of the driving scene. This dense representation supports robust perception, planning, and interaction in complex traffic environments. As illustrated in Fig. 1, SOP helps to bridge perceptual gaps and contributes to a more consistent and complete scene understanding.

Recent SOP methods have adopted Transformer-based architectures [1]–[3] to improve scene completion by leveraging attention mechanisms. However, despite their ability to capture long-range dependencies, existing Transformer-based SOP models face two critical challenges. First, they lack explicit modeling of local spatial context, which is essential for maintaining geometric consistency and accurate reasoning in 3D environments. Second, the reliance on sparse depth-guided query sampling, commonly used to reduce the high computational cost of full 3D self-attention, limits their effectiveness in distant or occluded regions where LiDAR data is extremely sparse and unevenly distributed.



(a) Sparse LiDAR input (b) Dense semantic prediction

(c) RGB image of a traffic scene

Fig. 1: SWA-SOP estimates a dense, semantically labeled scene (b) from a single-frame LiDAR input (a). The proposed SWA module can also be integrated into image-based pipelines. RGB images (c) are used for visualization and extended experiments.

To address these issues, we propose Spatially-aware Window Attention Semantic Occupancy Prediction (SWA-SOP), an attention mechanism that explicitly captures local spatial context while maintaining computational efficiency. Inspired by sparse convolution, SWA-SOP applies a sliding-window attention strategy with adaptive skipping, allowing the model to selectively process regions based on query validity. Furthermore, SWA-SOP incorporates spatial information into keys, queries, and values, enabling the attention mechanism to better model geometric relationships and preserve structural continuity in scene prediction. The proposed SWA module is simple to implement and can be readily integrated into existing SOP pipelines, consistently improving prediction accuracy across different input modalities.

In summary, our main contributions are as follows:

1) We develop a window-based attention mechanism. Compared to kernel-based convolutions, it supports flex-

*Equal contribution.

[1]Autonomous Intelligent Systems group, Computer Science Institute VI – Intelligent Systems and Robotics, University of Bonn, Germany cao@ais.uni-bonn.de

[2]Center for Robotics, University of Bonn, Germany

[3]Lamarr Institute for Machine Learning and Artificial Intelligence, Germany

ible feature aggregation during downsampling and more effective geometric expansion during upsampling.
2) We introduce the Spatially-aware Window Attention (SWA) module, which incorporates local spatial context into attention computation, enhancing spatial reasoning and structural consistency in scene prediction.
3) We demonstrate that SWA can be used as a plug-and-play component in existing camera-based SOP pipelines, leading to consistent improvements across input modalities.

II. Related Work

A. LiDAR-based and Camera-based 3D Perception

Processing LiDAR point clouds and images into meaningful 3D representations is a core challenge in perception. For LiDAR-based semantic segmentation and object detection, common approaches include voxel-based and point-based methods. Voxel-based models [4], [5] convert point clouds into regular grids for efficient processing with 3D CNNs or sparse convolutions. Point-based methods [6]–[9] operate directly on irregular point sets to preserve fine-grained geometry. Hybrid methods such as CenterPoint [10] combine both paradigms to balance accuracy and efficiency. In parallel, monocular and stereo methods [11]–[13] estimate depth or generate pseudo-LiDAR from RGB images. Sensor fusion techniques further enhance perception by combining LiDAR accuracy with visual semantics. SLCF-Net [14], for example, introduces a sequential fusion framework for semantic scene completion. Recent advances in domain-adaptive modeling [15] and prompt-driven reasoning [16] further suggest promising directions for improving robustness and transferability in multimodal perception.

B. Semantic Occupancy Prediction (SOP)

Semantic Occupancy Prediction (SOP) was first formulated by Song et al. [17] as the task of inferring a complete, semantically labeled 3D occupancy grid from a single RGB-D image. With the introduction of SemanticKITTI [18], SOP has expanded to large-scale outdoor LiDAR data, motivating the development of more scalable architectures. Early methods such as LMSCNet [19] adopted hierarchical 3D convolutions for efficient voxel processing. Recently, SOP from monocular images has attracted increasing attention. onoScene [20] initiated this direction via depth-to-voxel projection, while OC-SOP [21] builds on it by introducing object-centric awareness for enhanced semantic and geometric reasoning. In parallel, transformer-based methods [22], [23] refine voxel features using deformable attention [24], but often suffer from sparsity and errors in depth estimation, especially in distant or occluded regions. Alternatively, Diff-SSC [25] explores a generative approach based on diffusion processes, offering a different modeling paradigm at the cost of higher computational complexity.

C. Window-based Computation for Sparse 3D Data

In 3D scene understanding, data is inherently sparse, as most meaningful structures lie on 2D surfaces within 3D space. Applying dense convolutions leads to redundant computation over empty or occluded regions. Sparse convolutional neural networks (SCNNs) [26]–[28] improve efficiency by restricting operations to active voxels through a sliding-window mechanism. Inspired by this, many attention-based models adopt similar windowed strategies to reduce the computational burden of global attention. Instead of fixed kernels, they perform local attention within windows, allowing more flexible, data-adaptive feature aggregation. For example, Stratified Transformer [29] uses non-overlapping windows with sparse global links, SphereFormer [30] introduces radial windows, and VoTr [31] applies attention to valid voxels within sliding windows. However, these models are primarily designed for detection or segmentation, focusing only on observed regions without generating dense scene completions. In contrast, semantic occupancy prediction (SOP) requires not only flexible data association, but also the ability to reason about observed geometry and complete unobserved regions. Existing attention mechanisms often lack modeling of local spatial context, limiting their ability to capture geometric structure in sparse 3D scenes. To overcome this, we introduce Spatially-aware Window Attention (SWA), which incorporates local spatial context into attention computation, enabling both improved local reasoning and more effective geometric expansion.

III. Method

We formulate semantic occupancy prediction as the task of estimating a dense semantic volume from a sparse point cloud. Given a LiDAR point cloud $\mathcal{X} = \{p_1, ..., p_N\} \subset \mathbb{R}^3$, where each p_i denotes a 3D coordinate, the goal is to predict a semantic occupancy volume $\hat{\mathcal{Y}}$ defined over a discretized 3D space $\mathcal{V} \subset \mathbb{R}^3$. The space $\mathcal{V}$ is divided into a regular voxel grid of size $H \times W \times D$, and each voxel in $\hat{\mathcal{Y}}$ is assigned a label from the set $\mathcal{L} = \{0, 1, ..., C\}$, where 0 indicates empty space and 1 to C correspond to semantic classes. The model aims to infer $\hat{\mathcal{Y}} \in \mathcal{L}^{H \times W \times D}$ based on the sparse input $\mathcal{X}$, including predictions for unobserved regions, to reconstruct a complete semantic representation of the scene.

To solve this task, we design a pipeline comprised of a series of geometric and semantic reasoning stages, as illustrated in Fig. 2. We first apply SphereFormer [30] to perform point-wise semantic segmentation on $\mathcal{X}$. To enrich the feature representation, the final semantic logits are concatenated with intermediate point-wise features extracted from SphereFormer. These enhanced point features are then voxelized to form a sparse semantic voxel grid, which serves as input to a U-Net architecture with four hierarchical levels. The U-Net processes the sparse input in a coarse-to-fine manner to generate a dense volumetric prediction.

The U-Net backbone is augmented with our proposed Spatially-aware Window Attention (SWA) module, which replaces conventional convolutions with attention-based operations. SWA is adaptable to different hierarchical levels of feature representation, making it naturally integrable into both encoder and decoder stages, unlike standard U-Nets that utilize pooling layers during up- and down-sampling. SWA

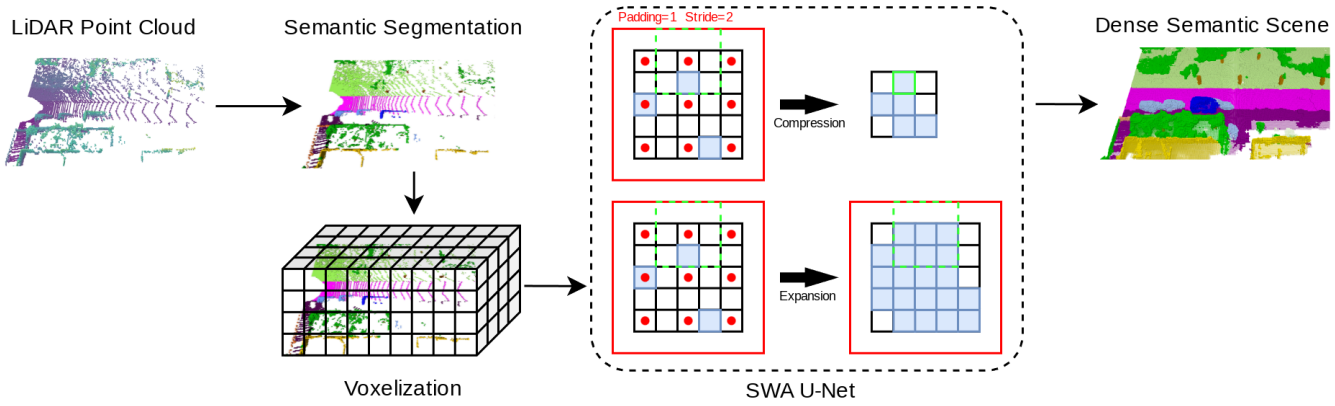


Fig. 2: Overall pipeline of SWA-SOP. The raw LiDAR point cloud is first semantically segmented by SphereFormer [30] and then voxelized to form an initial semantic scene grid. This grid is processed by a SWA U-Net, which applies sliding-window attention inspired by sparse convolutions. During downsampling, SWA performs feature compression across valid regions; during upsampling, it supports geometric expansion by propagating features into neighboring empty voxels, enabling effective scene completion.

performs attention-driven computations conditioned on voxel validity. Each block adopts an architecture with multi-head attention and a feedforward network, using layer normalization and residual connections in a pre-norm configuration for improved training stability. By combining sliding windows with intra-window spatial attention, SWA offers a flexible and structurally informed alternative to convolutional layers.

A. Sliding Window Computation

We define a local window $\mathcal{W}$ as a spatial subregion of size $h \times w \times d$ within the volume $\mathcal{V}$. Each window can be flattened into an ordered sequence $\mathcal{S} = [s_1, s_2, ..., s_L]$, where $L = hwd$ is the number of voxel slots. The sequence is constructed using a fixed scanning pattern (e.g., z-major order), and each element s_i denotes a voxel located at slot index i. The index i reflects the voxel's relative spatial position within the window, enabling the model to capture local geometric structure through consistent positional alignment.

Following the design of sparse convolutional networks [28], we apply a sliding-window traversal over the 3D volume to determine where attention computation should occur. As illustrated in SWA U-Net of Fig. 2, the window moves across the padded volume $\mathcal{V}$ with stride S. At each location, attention is computed only if the window contains at least one active (non-empty) voxel. This conditional activation reduces unnecessary computation and mirrors the sparsity-aware behavior of sparse convolutions.

Compared to standard U-Nets that rely on pooling and interpolation, our SWA U-Net performs attention-based computations within these local windows. In the encoder, attention compresses input features by aggregating them over valid regions. In the decoder, attention propagates information into adjacent empty voxels. These positions are initialized with shared learnable embeddings, allowing the model to fill geometric gaps and densify the volume. This hierarchical transformation—from sparse to compact, and from sparse to dense—enables the network to complete large-scale semantic scenes.

B. Intra-Window Spatial Embedding

To incorporate local spatial context into attention computation, we propose a spatial embedding mechanism, which is applied in addition to the standard multi-head attention, as shown in Fig. 3. Let $F \in \mathbb{R}^{L \times d}$ denote the input features of L voxels in a window, each represented by a d-dimensional vector. For each attention head $m \in \{1, ..., M\}$, we compute head-specific key and value features using a dedicated feedforward layer followed by layer normalization:

$$K^{[m]} = \mathrm{LN}(\mathrm{FF}_k^{[m]}(F)), \quad V^{[m]} = \mathrm{LN}(\mathrm{FF}_v^{[m]}(F)) \tag{1}$$

Here, $K^{[m]} = [k_1^{[m]}, ..., k_L^{[m]}]$ and $V^{[m]} = [v_1^{[m]}, ..., v_L^{[m]}]$ denote the key and value vectors across all positions for head m. Each projection is shared across positions but varies with the attention head, enabling diverse representations.

To introduce spatial awareness, we apply a position-aware modulation to the key and value features. For each slot $i \in \{1, ..., L\}$, a dedicated feedforward layer FF_i is applied to all heads at that position:

$$k_i'^{[m]} = \mathrm{FF}_{k,i}(k_i^{[m]}), \quad v_i'^{[m]} = \mathrm{FF}_{v,i}(v_i^{[m]}) \tag{2}$$

This position-aware modulation allows the attention mechanism to encode relative spatial position directly into the key and value features.

C. Center Query

To construct the query vector for attention, we adopt a center query strategy. Each attention window selects a center position c, which determines the query location for that window. If the center voxel is active, the query vector q_c is computed from its input feature. Otherwise, a synthetic query is generated from the voxel's global grid index using a separate MLP. This design ensures that a valid query vector is always available, even in sparsely populated regions.

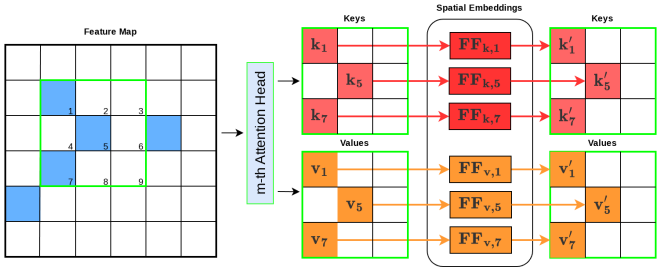


Fig. 3: Illustration of the spatial embedding module. All valid voxels (highlighted in blue) are provided to each attention head with identical input features. Taking the m-th head as an example, spatial embedding is applied to each slot index through a dedicated feedforward layer that is shared across all heads, enabling the attention mechanism to integrate local spatial context in a position-dependent, but head-independent manner.

Formally, the query vector is defined as:

$$q_c = \begin{cases} \mathrm{FF}_q(f_c), & \text{if the } f_c \text{ is non-empty} \\ \mathrm{FF}_q(p_c), & \text{otherwise} \end{cases} \tag{3}$$

where $f_c \in \mathbb{R}^d$ is the feature at voxel c, and $p_c \in \mathbb{Z}^3$ denotes its global grid index.

Unlike spatial embedding, which encodes all valid voxels individually, the center query q_c captures the contextual intent of the entire window and serves as a consistent anchor point across windows. When the center voxel is inactive, the center query still incorporates global context into the attention computation, allowing the model to integrate both global positioning and local spatial structure. The contrast between spatial embedding and center query, as well as the overall intra- and inter-window attention computation flow, is illustrated in Fig. 4.

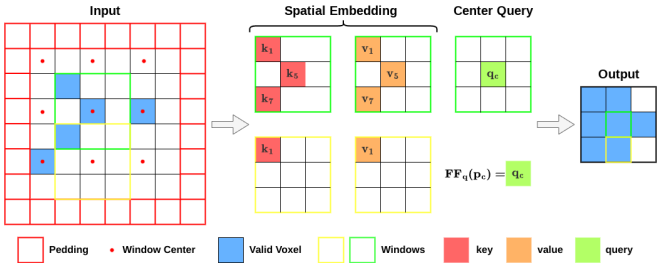


Fig. 4: Overview of the intra- and inter-window attention pipeline. Windows traverse the voxel volume, and attention is computed only when active voxels are present inside the window. Keys and values are extracted from active elements with spatial embedding. The center query is generated either from the center voxel feature if present, or from its global grid index embedding if absent, and attends to all valid positions in the window.

D. Attention Computation

Our attention computation is built upon the combination of spatial embeddings and the center query strategy described above. Given a set of key vectors $\{k_i\}_{i=1}^{L}$ extracted from neighboring voxels within the window, we compute the attention energy at each position i as the element-wise dot product between the center query q_c and the corresponding key k_i:

$$e_i = q_c^\top k_i = \sum_{k=1}^{d} q_{c,k} \cdot k_{i,k} \tag{4}$$

The energy values $\{e_i\}$ are then passed through a softmax operation to obtain normalized attention weights $\{\alpha_i\}$. To further enhance spatial adaptability, we introduce per-head modulation weights $\gamma^{[m]} \in \mathbb{R}^L$, where each element $\gamma_i^{[m]}$ reweights the attention score at position i for attention head m:

$$\tilde{\alpha}_i^{[m]} = \gamma_i^{[m]} \cdot \alpha_i^{[m]} \tag{5}$$

These spatial weights act analogously to learnable convolutional kernels, enabling each attention head to emphasize or suppress specific positions within the window.

IV. Evaluation

A. Experiment Setup

1) Dataset: We conduct our experiments on the SemanticKITTI dataset [18], a widely used real-world autonomous driving dataset derived from the KITTI Odometry Benchmark [36]. It provides dense point-wise semantic annotations for large-scale outdoor driving environments. For semantic occupancy prediction, consecutive LiDAR scans are aggregated and voxelized to form dense semantic volumes, where each voxel is assigned a semantic label.

The input space is defined as $\mathcal{V}_{\text{kitti}} = \{(x, y, z) \mid x \in [0, 51.2]\,\text{m},\ y \in [-25.6, +25.6]\,\text{m},\ z \in [-2.0, 4.4]\,\text{m}\}$ and discretized into a regular voxel grid of size $256 \times 256 \times 32$, where each voxel represents a volume of $(0.2\,\text{m})^3$. Each voxel is labeled with one of 21 categories, including 19 semantic classes, empty space, and unknown regions. SemanticKITTI also provides an unknown mask for regions never directly observed from any scan position. These voxels are considered invalid and are excluded from both training and evaluation. The dataset comprises 22 sequences in total. Following the standard split, sequences 00–10 (excluding 08) are used for training, sequence 08 for validation, and sequences 11–21 for testing. Semantic occupancy prediction is evaluated using two standard metrics. The voxel-wise IoU focuses solely on binary occupancy status, measuring whether each voxel is occupied or free. In contrast, the mean IoU (mIoU) reflects the overall semantic segmentation accuracy across all valid categories within occupied regions.

2) Implementation Details: All models are trained for 50 epochs using the cross-entropy loss. We adopt a polynomial learning rate decay schedule with an initial learning rate of 0.006 and a decay power of 0.9. Each attention layer employs 8 heads with a hidden feature dimension of 128. Attention is computed within local windows of size $3 \times 3 \times 3$, using a stride of 2 and padding of 1. Sparse convolution operations are implemented using the SpConv library [37]. Experiments on the point cloud pipeline are conducted using 6 NVIDIA TITAN RTX GPUs, while the RGB-based pipeline is trained with 4 GPUs.

TABLE I: Quantitative results on SemanticKITTI hidden test set using LiDAR inputs.

Method	SC	SSC																		
	IoU	**car** (3.92%)	**bicycle** (0.03%)	**motorcycle** (0.03%)	**truck** (0.16%)	**other-vehicle** (0.20%)	**person** (0.07%)	**bicyclist** (0.07%)	**motorcyclist** (0.05%)	**road** (15.30%)	**parking** (1.12%)	**sidewalk** (11.13%)	**other-ground** (0.56%)	**building** (14.10%)	**fence** (3.90%)	**vegetation** (39.30%)	**trunk** (0.51%)	**terrain** (9.17%)	**pole** (0.29%)	**traffic-sign** (0.08%)	mIoU
LMSCNet [19]	56.7	30.9	0.0	0.0	1.5	0.8	0.0	0.0	0.0	64.8	29.0	34.7	4.6	38.1	21.3	41.3	19.9	32.1	15.0	0.8	17.6
Local-DIFs [32]	57.7	34.8	3.6	2.4	4.4	4.8	2.5	1.1	0.0	67.9	**40.1**	42.9	11.4	40.4	29.0	42.2	26.5	39.1	21.3	17.5	22.7
SSA-SC [33]	**58.8**	**36.5**	13.9	4.6	5.7	7.4	4.4	2.6	0.7	**72.2**	37.4	**43.7**	10.9	43.6	30.7	43.5	25.6	41.8	14.5	6.9	23.5
JS3C-Net [34]	56.6	33.3	14.4	8.8	7.2	12.7	8.0	5.1	0.4	64.7	34.9	39.9	14.1	39.4	30.4	43.1	19.6	40.5	18.9	15.9	23.8
S3CNet [35]	45.6	31.2	**41.5**	**45.0**	6.7	16.1	**45.9**	**35.8**	**16.0**	42.0	17.0	22.5	7.9	**52.2**	31.3	39.5	**34.0**	21.2	**31.0**	24.3	**29.5**
SWA-SOP(Ours)	57.9	35.4	17.2	15.5	**8.9**	**18.2**	5.8	3.7	1.3	68.4	39.0	41.6	**20.0**	41.3	**34.5**	**44.5**	31.2	**42.2**	26.4	**24.7**	27.4

IoU focuses solely on the occupancy status, while mIoU evaluates individual semantic categories. **Best** and second best results are highlighted.

B. Main Results for LiDAR Input

In Tab. I, we present the results of SWASOP on the hidden test set of the SemanticKITTI benchmark, compared to other state-of-the-art approaches. SWASOP integrates our proposed spatial embeddings and center queries. Our method achieves the second best IoU, slightly behind SSA-SC. In terms of mIoU, SWASOP also ranks second, marginally lower than S3CNet, and consistently ranks among the top across most semantic categories, often achieving either the best or second-best class-wise performance. It should be noted that S3CNet benefits from additional 2D BEV features, which significantly enhance the recognition of foreground objects. However, these 2D projections do not bring notable improvements to 3D scene completion and result in a significantly lower IoU compared to other baselines, indicating weaker capability in holistic occupancy prediction and leading to relatively inferior overall performance. In contrast, SWASOP maintains a more balanced and spatially consistent reconstruction, achieving strong results when considering both mIoU and IoU jointly.

Fig. 5 provides a qualitative comparison between SWASOP and JS3C-Net on selected scenes. In the first example, JS3C-Net incorrectly classifies a person as a pole, which also leads to a noticeable distortion in the predicted road geometry. In the second scene, a bicyclist is misclassified as a vehicle, resulting in shape and semantic inconsistency. It is worth noting that in the SemanticKITTI dataset, all dynamic objects are accumulated over time, which often leads to artifacts in the voxelized ground truth. Beyond improvements related to object-level recognition, SWASOP demonstrates overall superior performance in both the geometric prediction of foreground objects and the accurate reconstruction of large-scale background regions, highlighting its strength in producing consistent and detailed scene prediction.

C. Extended Experiments for RGB Inputs

To further demonstrate the generality of the proposed SWA module, we evaluate it as a plug-in component within an existing SOP pipeline based on RGB inputs. Specifically, we replace the original deformable self-attention module in VoxFormer with an adapted SWA U-Net, using the proposal queries generated in the first stage of VoxFormer as input voxels. This substitution is motivated by the observation that deformable attention relies on accurate depth-guided sampling and may suffer in far-range regions where the input becomes extremely sparse and the depth estimation is unreliable. In contrast, the SWA module leverages structured local attention to model geometric context within each window, offering enhanced robustness under such conditions. The results, reported in Tab. II, show that SWA consistently improves performance, indicating stronger scene prediction capabilities in sparse and uncertain areas. These findings highlight the module's effectiveness beyond LiDAR-based settings and its potential for integration into diverse perception architectures.

TABLE II: Quantitative results of our SWA module on the SemanticKITTI hidden test set using RGB inputs.

Method	MonoScene [20]	VoxFormer-S [22]	OccFormer [23]	VoxFormer-S + SWA
mIoU	11.08	12.20	12.32	**13.19**
IoU	34.16	42.95	34.53	**44.20**
car	18.8	20.8	21.6	**23.3**
bicycle	0.5	1.0	**1.5**	0.9
motorcycle	0.7	0.7	**1.7**	0.4
truck	3.3	3.5	1.2	**4.8**
other-vehicle	**4.4**	3.7	3.2	3.0
person	1.0	1.4	**2.2**	1.1
bicyclist	1.4	**2.6**	1.1	1.2
motorcyclist	**0.4**	0.2	0.2	0.2
road	54.7	53.9	55.9	**56.0**
parking	24.8	21.1	**31.5**	23.6
sidewalk	27.1	25.3	**30.3**	26.5
other-ground	5.7	5.6	6.5	**7.1**
building	14.4	19.8	15.7	**21.5**
fence	11.1	11.1	11.9	**13.3**
vegetation	14.9	22.4	16.8	**25.0**
trunk	2.4	7.5	3.9	**8.3**
terrain	19.5	21.3	21.3	**22.2**
pole	3.3	5.1	3.8	**5.9**
traffic-sign	2.1	4.9	3.7	**6.4**

IoU focuses solely on the occupancy status, while mIoU evaluates individual semantic categories. **Best** results are highlighted.

D. Ablation Studies

We conduct an ablation study to investigate the individual contributions of the spatial embedding and center query

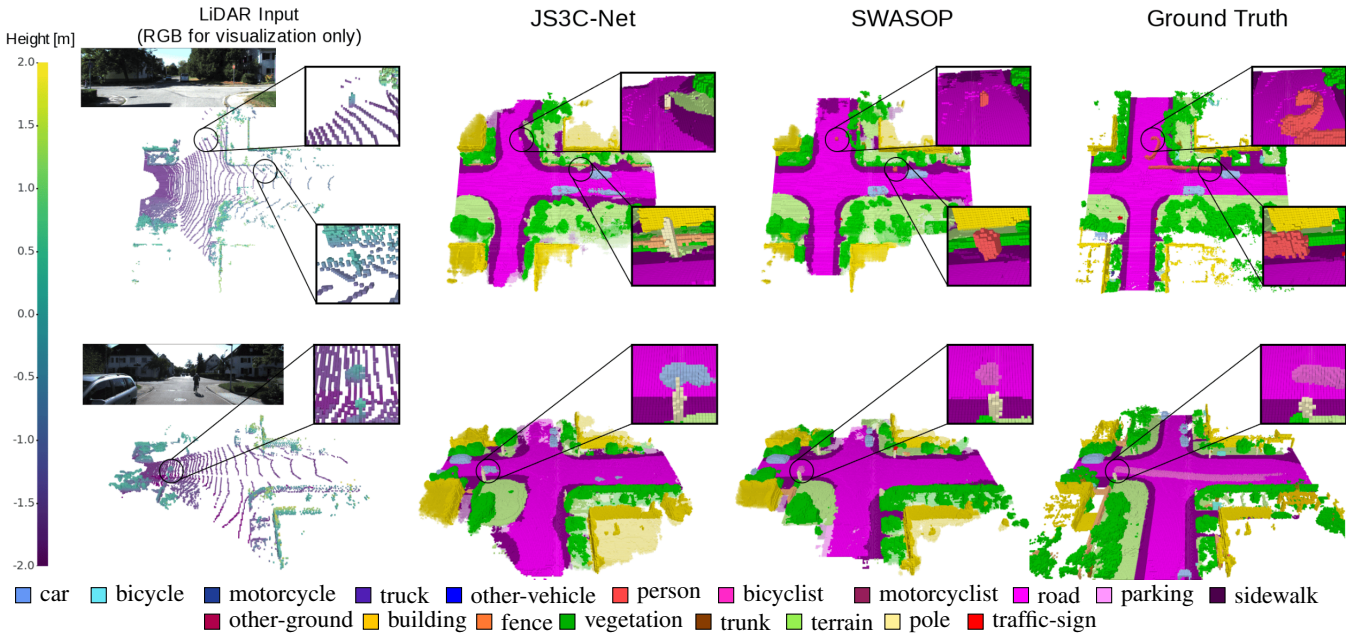


Fig. 5: Qualitative results on the SemanticKITTI validation set. We show the voxelized LiDAR input (RGB image included for visualization only), the ground truth, and predictions from JS3C-Net for comparison. All 19 semantic classes are rendered without empty (void) regions. Predictions located in unknown areas are visualized with 20% opacity.

components, with results summarized in Tab. III. We use mIoU as the primary evaluation metric, as it most directly reflects semantic prediction quality in the SOP task. Three configurations (A, B, and C) are compared. Method A represents the full SWASOP architecture and achieves the best performance. Method B adopts the same spatial embedding for keys, values, and queries, without using the center query. The performance gap between A and B demonstrates the effectiveness of the center query, which provides a consistent and adaptive anchor for attention. When the center voxel is inactive, the query is instead generated from its global position embedding, enabling stable attention even in sparse regions. Method C disables the SWA module and directly uses a sparse convolutional U-Net. The lower performance confirms the superiority of our attention-based spatial modeling over traditional convolutional operations.

TABLE III: Ablation study of each sub-module on the SemanticKITTI validation set.

Method	A	B	C
Spatial Embedding	✓	✓	-
Center Query	✓	-	-
mIoU	27.91	27.07	24.46

V. Conclusions

In this work, we propose SWA-SOP, a novel semantic occupancy prediction framework. By adopting a sliding-window mechanism that selectively computes attention over valid regions, SWA-SOP significantly improves the efficiency of global attention-based models. Within each window, we leverage spatial embeddings and queries to perform localized attention, offering greater flexibility than traditional convolutional operations. Experiments on the SemanticKITTI benchmark demonstrate that our LiDAR-based method achieves state-of-the-art performance compared to existing approaches. Furthermore, we show that the proposed SWA module can be effectively integrated into image-based pipelines, leading to consistent performance gains. These results highlight the versatility of SWA and its potential for deployment across diverse sensor configurations in autonomous driving perception systems. In future work, we plan to further integrate SWA into recent advances in cross-modal fusion [38], [39] and prompt-based representation learning [40], enabling more scalable and flexible deployment in large-scale multimodal perception systems. Additionally, to support deployment on resource-constrained platforms, we aim to incorporate lightweight modeling techniques [41], [42] to reduce computational cost without sacrificing performance.

References

[1] F. Zhang, G. Chen, H. Wang, and C. Zhang, "Cf-dan: Facial-expression recognition based on cross-fusion dual-attention network," *Computational Visual Media*, 2024.

[2] F. Zhang, G. Chen, H. Wang, J. Li, and C. Zhang, "Multi-scale video super-resolution transformer with polynomial approximation," *IEEE Transactions on Circuits and Systems for Video Technology*, 2023.

[3] Y. Wang, H. Wang, and F. Zhang, "A medical image segmentation model with auto-dynamic convolution and location attention mechanism," *Computer Methods and Programs in Biomedicine*, 2025.

[4] Y. Zhou and O. Tuzel, "VoxelNet: End-to-end learning for point cloud based 3D object detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[5] X. Zhu, H. Zhou, T. Wang, F. Hong, Y. Ma, W. Li, H. Li, and D. Lin, "Cylindrical and asymmetrical 3D convolution networks for LiDAR segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021, pp. 9939–9948.

[6] C. R. Qi, H. Su, K. Mo, and L. J. Guibas, "PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[7] C. R. Qi, L. Yi, H. Su, and L. J. Guibas, "PointNet++: Deep hierarchical feature learning on point sets in a metric space," *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 30, 2017.

[8] Y. Li, J. Dong, Z. Dong, C. Yang, Z. An, and Y. Xu, "SRKD: Towards efficient 3D point cloud segmentation via Structure- and Relation-aware knowledge distillation," *arXiv preprint arXiv:2506.17290*, 2025.

[9] M. Wang, D. Li, J. R. Casas, and J. Ruiz-Hidalgo, "Adaptive fusion of lidar features for 3d object detection in autonomous driving," *Sensors*, 2025.

[10] T. Yin, X. Zhou, and P. Krahenbuhl, "Center-based 3D object detection and tracking," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

[11] F. Shamsafar, S. Woerz, R. Rahim, and A. Zell, "MobileStereoNet: Towards lightweight deep networks for stereo matching," in *IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2022.

[12] Y. Wang, W.-L. Chao, D. Garg, B. Hariharan, M. Campbell, and K. Weinberger, "Pseudo-LiDAR from visual depth estimation: Bridging the gap in 3D object detection for autonomous driving," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019.

[13] Y. You, Y. Wang, W.-L. Chao, D. Garg, G. Pleiss, B. Hariharan, M. Campbell, and K. Q. Weinberger, "Pseudo-LiDAR++: Accurate depth for 3D object detection in autonomous driving," *International Conference on Learning Representations, (ICLR)*, 2020.

[14] H. Cao and S. Behnke, "SLCF-Net: Sequential LiDAR-camera fusion for semantic scene completion using a 3D recurrent U-Net," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2024, pp. 2767–2773.

[15] L. Zhou, N. Li, M. Ye, X. Zhu, and S. Tang, "Source-free domain adaptation with class prototype discovery," *Pattern Recognition*, 2024.

[16] N. Li, M. Ye, L. Zhou, S. Tang, Y. Gan, Z. Liang, and X. Zhu, "Self-prompting analogical reasoning for uav object detection," in *National Conference on Artificial Intelligence (AAAI)*, 2025.

[17] S. Song, F. Yu, A. Zeng, A. X. Chang, M. Savva, and T. Funkhouser, "Semantic scene completion from a single depth image," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017, pp. 1746–1754.

[18] J. Behley, M. Garbade, A. Milioto, J. Quenzel, S. Behnke, C. Stachniss, and J. Gall, "SemanticKITTI: A dataset for semantic scene understanding of LiDAR sequences."

[19] L. Roldão, R. de Charette, and A. Verroust-Blondet, "LMSCNet: Lightweight multiscale 3D semantic completion," in *International Conference on 3D Vision (3DV)*, 2020.

[20] A.-Q. Cao and R. De Charette, "MonoScene: Monocular 3D semantic scene completion," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[21] H. Cao and S. Behnke, "OC-SOP: Enhancing Vision-Based 3d semantic occupancy prediction by Object-Centric awareness," *arXiv preprint arXiv:2506.18798*, 2025.

[22] Y. Li, Z. Yu, C. Choy, C. Xiao, J. M. Alvarez, S. Fidler, C. Feng, and A. Anandkumar, "VoxFormer: Sparse voxel transformer for camera-based 3d semantic scene completion," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[23] Y. Zhang, Z. Zhu, and D. Du, "OccFormer: Dual-path transformer for vision-based 3d semantic occupancy prediction," in *IEEE International Conference on Computer Vision (ICCV)*, 2023.

[24] X. Zhu, W. Su, L. Lu, B. Li, X. Wang, and J. Dai, "Deformable DETR: Deformable transformers for end-to-end object detection," *International Conference on Learning Representations, (ICLR)*, 2021.

[25] H. Cao and S. Behnke, "DiffSSC: Semantic LiDAR scan completion using denoising diffusion probabilistic models," *arXiv preprint arXiv:2409.18092*, 2024.

[26] B. Graham, "Sparse 3D convolutional neural networks," in *British Machine Vision Conference (BMVC)*, 2015.

[27] B. Graham, M. Engelcke, and L. van der Maaten, "3D semantic segmentation with submanifold sparse convolutional networks," *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[28] B. Liu, M. Wang, H. Foroosh, M. Tappen, and M. Pensky, "Sparse convolutional neural networks," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015.

[29] X. Lai, J. Liu, L. Jiang, L. Wang, H. Zhao, S. Liu, X. Qi, and J. Jia, "Stratified transformer for 3D point cloud segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[30] X. Lai, Y. Chen, F. Lu, J. Liu, and J. Jia, "Spherical transformer for LiDAR-based 3D recognition," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[31] J. Mao, Y. Xue, M. Niu *et al.*, "Voxel transformer for 3D object detection," in *IEEE International Conference on Computer Vision (ICCV)*, 2021.

[32] C. B. Rist, D. Emmerichs, M. Enzweiler, and D. M. Gavrila, "Semantic scene completion using local deep implicit functions on LiDAR data," *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)*, 2021.

[33] X. Yang, H. Zou, X. Kong, T. Huang, Y. Liu, W. Li, F. Wen, and H. Zhang, "Semantic segmentation-assisted scene completion for LiDAR point clouds," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021.

[34] X. Yan, J. Gao, J. Li, R. Zhang, Z. Li, R. Huang, and S. Cui, "Sparse single sweep LiDAR point cloud segmentation via learning contextual shape priors from scene completion," in *National Conference on Artificial Intelligence (AAAI)*, 2021.

[35] R. Cheng, C. Agia, Y. Ren, X. Li, and L. Bingbing, "S3CNnet: A sparse semantic scene completion network for LiDAR point clouds," in *Proceedings of Machine Learning Research (PMLR)*, 2021.

[36] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the KITTI vision benchmark suite," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[37] SpConv Contributors, "SpConv: Spatially sparse convolution library," https://github.com/traveller59/spconv, 2022.

[38] W. Wu, Z. Chen, X. Qiu, S. Song, X. Huang, F. Ma, and J. Xiao, "LLM-Enhanced multimodal fusion for Cross-Domain sequential recommendation," *arXiv preprint arXiv:2506.17966*, 2025.

[39] W. Wu, S. Song, X. Qiu, X. Huang, F. Ma, and J. Xiao, "Image fusion for cross-domain sequential recommendation," in *ACM Web Conference*, 2025.

[40] W. Wu, X. Qiu, S. Song, Z. Chen, X. Huang, F. Ma, and J. Xiao, "Prompt categories cluster for weakly supervised semantic segmentation," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025, pp. 3198–3207.

[41] Y. Li, Y. Lu, Z. Dong, C. Yang, Y. Chen, and J. Gou, "SGLP: A similarity guided fast layer partition pruning for compressing large deep models," *arXiv preprint arXiv:2410.14720*, 2024.

[42] Y. Li, C. Yang, H. Zeng, Z. Dong, Z. An, Y. Xu, Y. Tian, and H. Wu, "Frequency-Aligned knowledge distillation for lightweight spatiotemporal forecasting," *arXiv:2507.02939*, 2025.