ProtoOcc: Accurate, Efficient 3D Occupancy Prediction Using Dual Branch Encoder-Prototype Query Decoder

Jungho Kim[1*], **Changwon Kang**[2*], **Dongyoung Lee**[2*], **Sehwan Choi**[2], **Jun Won Choi**[1†]
[1]Seoul National University,
[2]Hanyang University
jhkim@spa.snu.ac.kr, {changwonkang, dylee, sehwanchoi}@spa.hanyang.ac.kr, junwchoi@snu.ac.kr

Abstract

In this paper, we introduce ProtoOcc, a novel 3D occupancy prediction model designed to predict the occupancy states and semantic classes of 3D voxels via a deep semantic understanding of scenes. ProtoOcc consists of two main components: the *Dual Branch Encoder* (DBE) and the *Prototype Query Decoder* (PQD). The DBE produces a new 3D voxel representation by combining 3D voxel and BEV representations across multiple scales using a dual branch structure. This design combines the BEV representation, which offers a large receptive field, with the voxel representation, known for its higher spatial resolution, thereby improving both performance and computational efficiency. The PQD employs two types of prototype-based queries to expedite the Transformer decoding process. Scene-Adaptive Prototypes are generated from the 3D voxel features of the input sample, while Scene-Agnostic Prototypes are updated during training using an Exponential Moving Average of the Scene-Adaptive Prototypes. Using these prototype-based queries for decoding, we can directly predict 3D occupancy in a single step, eliminating the need for iterative Transformer decoding. Additionally, we propose *Robust Prototype Learning*, which introduces noise into the prototype generation process and trains the model to denoise during the training phase. This approach enhances the robustness of ProtoOcc against degraded prototype feature quality. ProtoOcc achieves state-of-the-art performance with 45.02% *mIoU* on the Occ3D-nuScenes benchmark. For the single-frame method, it reaches 39.56% *mIoU* with 12.83 FPS on an NVIDIA RTX 3090.

Code — https://github.com/SPA-junghokim/ProtoOcc

1 Introduction

Vision-based 3D occupancy prediction is a critical task for comprehensive scene understanding around the ego vehicle in autonomous driving. This task aims to simultaneously estimate occupancy states and semantic classes using multi-view images in 3D space, providing detailed 3D scene information. The typical prediction pipeline of previous methods comprises three main components: 1) a view transformation module, 2) an encoder, and 3) a decoder. Initially, backbone feature maps extracted from multi-view images are transformed into 3D spatial representations through a 2D-to-3D view transformation. An encoder network then processes these 3D representations to produce high-level semantic spatial features, capturing the overall scene context. Finally, a decoder network utilizes these encoded 3D spatial features to predict both semantic occupancy and class for all voxels composing the scene.

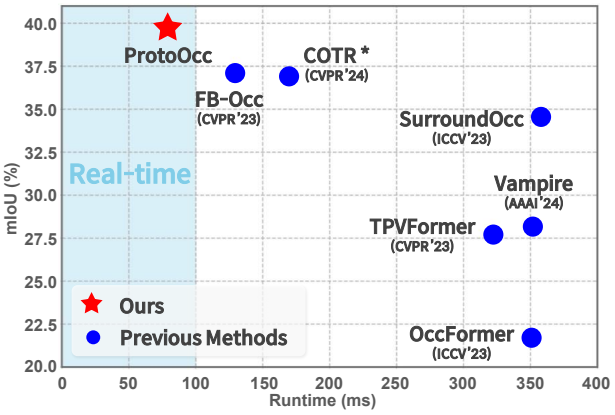

Figure 1: Comparisons of the *mIoU* and runtimes of different methods on the Occ3D-nuScenes validation set. ⋆ indicates results reproduced using publicly available codes. Inference time is measured on a single NVIDIA RTX 3090 GPU.

Existing works have explored enhancing encoder-decoder networks to improve both the accuracy and computational efficiency of 3D occupancy prediction. Various attempts have been made to optimize encoders using 3D spatial representations. Figure 2 (a) illustrates two commonly used 3D representations, including voxel representation (Li et al. 2023a; Wang et al. 2023; Wei et al. 2023) and Bird's-Eye View (BEV) representation (Hou et al. 2024; Yu et al. 2023). Voxel-based encoding methods (Zhang, Zhu, and Du 2023; Cao and De Charette 2022) used 3D Convolutional Neural Networks (CNNs) to encode voxel structures. However, the large number of voxels needed to represent 3D surroundings results in high memory and computational demands. While reducing the capacity of 3D CNNs can alleviate this com-

*These authors contributed equally.
†Corresponding author.

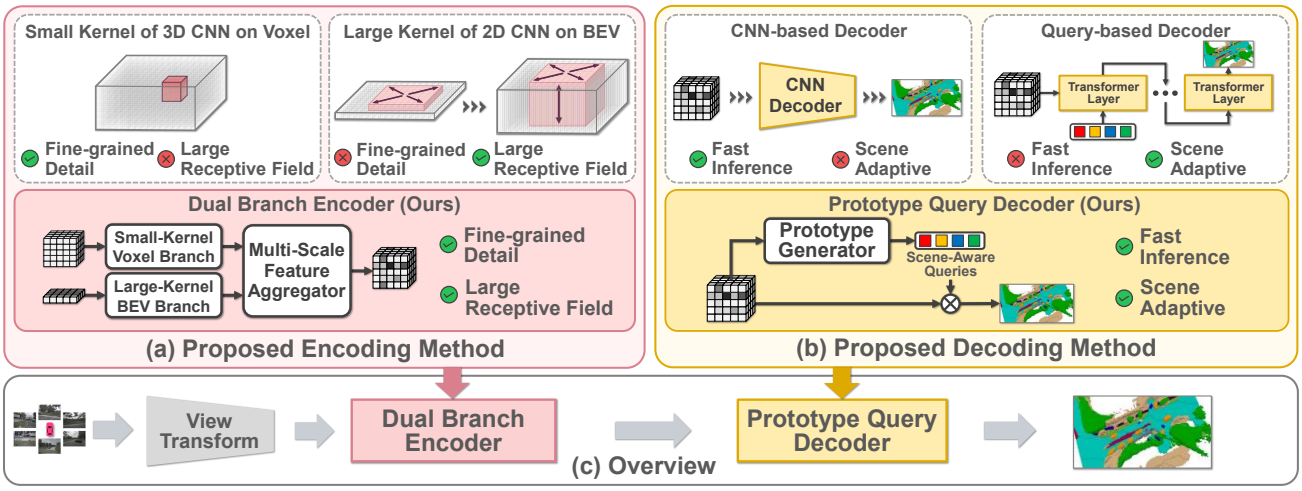

Figure 2: Overall structure of ProtoOcc. (a) Dual Branch Encoder combines voxel and BEV representations to efficiently model the large receptive fields while minimizing computational demands. (b) The Prototype Query Decoder utilizes prototypes to generate Scene-Aware Queries, enabling rapid inference without the need for iterative query decoding. (c) Our ProtoOcc framework combines the Dual Branch Encoder with the Prototype Query Decoder to enhance 3D occupancy prediction.

plexity, it also reduces the receptive field, which may compromise overall performance.

Unlike voxel representations, BEV representations project 3D information onto a 2D BEV plane, significantly reducing memory and computational requirements. After encoding the BEV representation using 2D CNNs, it is converted back into a 3D voxel structure for 3D occupancy prediction. However, this approach inherently loses detailed 3D geometric information due to the compression of the height dimension. Although incorporating additional 3D information (Hou et al. 2024; Yu et al. 2023) can enhance BEV representation, its performance remains limited by the inherent constraints of representing 3D scenes in a 2D format.

Another line of research focuses on enhancing the decoders. As illustrated in Figure 2 (b), two main decoding strategies exist: 1) CNN-based decoders (Cao and De Charette 2022; Zhou et al. 2024; Xu et al. 2024; Zhang et al. 2024) and 2) query-based decoders (Zhang, Zhu, and Du 2023; Tang et al. 2024; Liu et al. 2023). CNN-based decoders employed lightweight 3D CNNs to extract semantic voxel features, while query-based decoders iteratively decoded a query using the 3D representation obtained from the encoder. Although query-based decoders achieved better prediction accuracy, they required processing through multiple decoding layers, leading to increased inference time. Therefore, it is crucial to reduce this complexity while retaining the performance benefits of query-based decoders.

To address the aforementioned challenges, we introduce ProtoOcc, an efficient encoder-decoder framework for a 3D occupancy prediction network. As shown in Figure 1, ProtoOcc achieves state-of-the-art performance while achieving relatively fast inference (i.e., 77.9 ms) on a single NVIDIA RTX 3090 GPU.

As shown in Figure 2 (a), ProtoOcc utilizes a *Dual Branch Encoder* (DBE) with a dual-branch architecture. The voxel branch uses 3D CNNs with small kernel sizes to reduce computational complexity, while the BEV branch applies 2D CNNs with large kernel sizes to capture scene semantics with a larger receptive field. To combine the strengths of both representations, BEV and voxel features are fused across multiple scales to generate *Comprehensive Voxel Feature*. This dual encoding approach effectively captures fine-grained 3D structures and long-range spatial relationships across various scales.

Query-based decoding typically demands high computational complexity due to processing across multiple decoding layers. To overcome this, we propose the *Prototype Query Decoder* (PQD), which accelerates the decoding process by utilizing prototype-based queries and eliminating the need for iterative decoding. PQD generates Scene-Adaptive Prototypes by utilizing class-specific masks to aggregate features for each class from the Comprehensive Voxel Feature. While these prototypes can represent the semantic classes present in the input, challenges arise when certain semantic classes are absent in the input sample. To address this, we introduce Scene-Agnostic Prototypes, which are generated by accumulating Scene-Adaptive Prototypes across samples using an Exponential Moving Average (EMA) during training. By combining Scene-Adaptive and Scene-Agnostic Prototypes together, PQD forms Scene-Aware Queries, enabling efficient 3D occupancy prediction in a single iteration.

We also develop a novel training method for enhancing the performance of the proposed decoder. Since the prototypes are directly utilized for 3D occupancy prediction without an iterative query decoding, the quality of the prototypes significantly impacts the overall performance. To ensure robust predictions, we devise the *Robust Prototype Learning* framework that injects noise into the prototype generation

process and trains the model to counteract this noise during the training phase.

We evaluated ProtoOcc on the challenging Occ-3D nuScenes benchmark (Tian et al. 2024). ProtoOcc achieves an *mIoU* of $\mathbf{39.56\%}$, surpassing the performance of all existing single-frame methods, while operating at a processing speed of $\mathbf{12.83}$ FPS on an NVIDIA RTX 3090. Combined with multi-frame temporal fusion, ProtoOcc also achieves state-of-the-art performance among the latest multi-frame methods, with an *mIoU* of $\mathbf{45.02\%}$.

The contributions of this study are summarized below:

- We introduce ProtoOcc, a novel 3D occupancy prediction model that integrates a dual-branch encoding and query-based decoding to enhance both computational efficiency and accuracy for complex 3D environments.
- We propose an enhanced 3D representation for the encoder that jointly aggregates voxel and BEV representations through dual branch pipelines. This DBE method efficiently allocates resources, forming the largest receptive field with minimal computational cost.
- We propose a computationally efficient decoder performing 3D occupancy prediction in a single pass. This PQD generates queries representing each class from the encoded 3D spatial features and directly predicts semantic occupancy without a decoding process, thereby significantly reducing the computational complexity.
- ProtoOcc achieves state-of-the-art performance, with a 45.02% *mIoU* on the Occ-3D benchmark. It also achieves a 39.56% *mIoU* at a processing speed of 12.83 FPS.

2 Related Works

3D Encoding Methods for Occupancy Prediction

3D occupancy prediction (Tong et al. 2023) has attracted considerable interest in recent years due to its ability to reconstruct 3D volumetric scene structures from multi-view images. These approaches primarily utilize two widely adopted 3D representations, voxel and BEV, to encode 3D spatial information. MonoScene (Cao and De Charette 2022) bridged the gap between 2D and 3D representations by projecting 2D features along their line of sight and encoding voxelized semantic scenes with a 3D UNet. OccFormer (Zhang, Zhu, and Du 2023) introduced a dual-path transformer that independently processes voxel and BEV representations, dividing voxel data into BEV slices to decompose heavy 3D processing. FastOcc (Hou et al. 2024) reduced computational cost by replacing high-cost 3D CNNs in voxel space with efficient 2D CNNs in BEV space.

3D Decoding Methods for Occupancy Prediction

Recent studies (Zhao et al. 2024; Cao, Dai, and de Charette 2024; Liu et al. 2023; Tang et al. 2024) have introduced query-based decoders that capture scene-adaptive features by interacting with voxel features. OccFormer (Zhang, Zhu, and Du 2023) adopted masked attention in 3D space to iteratively decode query embeddings, thereby extracting semantic information from voxel features. COTR (Ma et al. 2024) introduced a coarse-to-fine semantic grouping strategy, dividing categories into semantic groups based on granularity and assigning distinct supervision for each group to address class imbalance.

2D Encoding Methods with Large Receptive Fields

Transformer-based models, such as ViT (Dosovitskiy et al. 2020) and Swin Transformer (Liu et al. 2021), have gained significant popularity in the field of computer vision. Recent studies (Luo et al. 2016; Yan et al. 2021) have shown that large receptive fields are a crucial factor in the success of these models. Recent research on CNN-based models has demonstrated that models with large receptive fields can achieve competitive performance with Transformer-based architectures. ConvNeXt (Liu et al. 2022) achieved competitive performance by modifying ConvNets with design principles from vision Transformers, including 7×7 depthwise convolutions. RepLKNet (Ding et al. 2022) scaled up convolutional kernels to as large as 31×31 utilizing reparameterization. LargeKernel3D (Chen et al. 2023) proposed spatial-wise partition convolutions, achieving a large receptive field in 3D while reducing computational costs.

3 ProtoOcc Method

Overview

The overall architecture of ProtoOcc is illustrated in Figure 2 (c). Initially, a 2D-to-3D view transformation generates both 3D voxel and BEV features from multi-view camera images. DBE then combines these features across multiple scales to produce Comprehensive Voxel Feature. Next, PQD produces class-specific Scene-Aware Queries from the Comprehensive Voxel Feature and utilizes them to predict 3D occupancy in a single pass.

2D-to-3D View Transformation. The 2D-to-3D View transformation process converts multi-view camera inputs into 3D features in both voxel and BEV formats through Lift-Splat-Shoot (LSS) method (Philion and Fidler 2020). 2D feature maps are extracted from multi-view images using a backbone network such as ResNet (He et al. 2016). These features are then fed into a depth network to predict depth distributions. Frustum features are generated by computing the outer product between 2D feature maps and depth distributions. The voxel-pooling method transforms these frustum features into a unified 3D voxel feature F_{vox}. Finally, the BEV feature F_{BEV} is reshaped from F_{vox} along the Z axis, changing from (D, X, Y, Z) to $(D{\times}Z, X, Y)$, where D denotes the channel dimension and (X, Y, Z) represents the volume scale.

Dual Branch Encoder

The structure of DBE is depicted in Figure 3 (a). DBE consists of two main components: the *Dual Feature Extractor* (DFE) and the *Hierarchical Fusion Module* (HFM). The DFE module captures fine-grained 3D structures in the voxel domain and long-range spatial relationships in the BEV domain, extracting features across multiple scales. The HFM

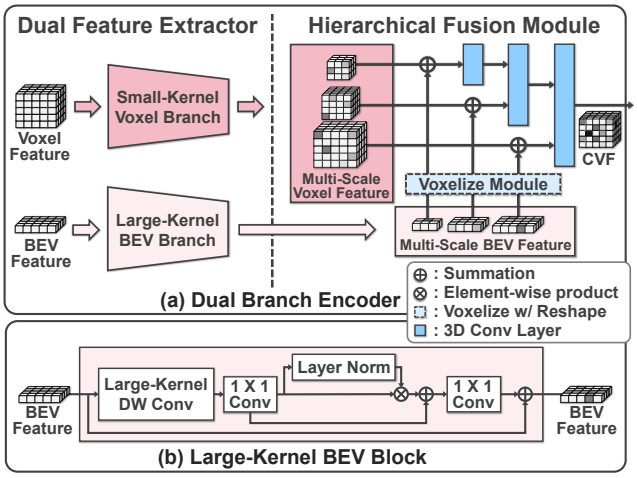

Figure 3: Details of Dual Branch Encoder. (a) DBE consists of DFE and HFM. DFE extracts multi-scale features using the dual encoders in the voxel and BEV domain. HFM aggregates these features from low to high scales to generate Comprehensive Voxel Feature V_{CVF}. (b) The Large-Kernel BEV Block comprises a large kernel depth-wise convolution, 1x1 convolutions, and layer normalization.

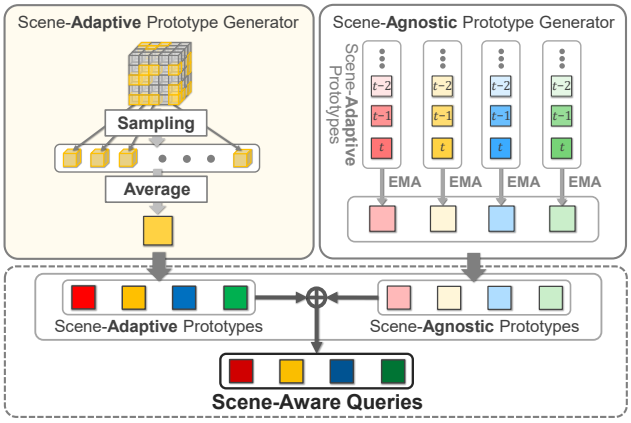

Figure 4: Details of prototype generation. AdaPG generates Scene-Adaptive Prototypes by sampling and averaging Comprehensive Voxel Feature for each class based on class-specific masks. AgnoPG generates Scene-Agnostic Prototypes by computing Scene-Adaptive Prototypes through the EMA method. Finally, Scene-Adaptive Prototypes and Scene-Agnostic Prototypes are combined into Scene-Adaptive Queries.

module hierarchically aggregates features from each domain, generating comprehensive context representations at various levels of detail.

Dual Feature Extractor. DFE consists of a voxel branch with 3D CNNs and a BEV branch with 2D CNNs designed for distinct spatial representations. The voxel branch aims to efficiently extract fine-grained features by utilizing small kernels to minimize computational complexity. F_{vox} is processed through 3D CNN residual blocks and downsampling layers, generating multi-scale voxel features $\mathbf{V}^{\text{vox}} = \{V_i^{\text{vox}} \in \mathbb{R}^{D_i \times X_i \times Y_i \times Z_i}\}_{i=1}^{S}$, where i denotes the scale index and S represents the total number of scales.

The BEV branch is designed to capture long-range spatial relationships by utilizing 2D CNNs with larger kernel sizes, which effectively expand the receptive field. This approach avoids the high computational burden required by 3D CNNs. Multi-scale BEV features $\mathbf{B}^{\text{BEV}} = \{B_i^{\text{BEV}} \in \mathbb{R}^{D_i' \times X_i \times Y_i}\}_{i=1}^{S}$ are extracted from F_{BEV} through a series of 2D CNN residual blocks followed by a downsampling layer.

Hierarchical Fusion Module. HFM integrates multi-scale voxel and BEV representations to generate the Comprehensive Voxel Feature. This process involves hierarchical aggregation of features from both domains through a sequence of upsampling layers and a 3D CNN. In each layer, the BEV feature B_i^{BEV} at the i-th scale is voxelized into $V_i^{\text{BEV}} \in \mathbb{R}^{D_i \times X_i \times Y_i \times Z_i}$ through a reshape operation, aligning it with the voxel feature space. Subsequently, the fused voxel feature V_i^{fused} is derived by combining the voxel feature V_i^{vox}, the voxelized BEV feature V_i^{BEV}, and the upsampled fused voxel feature $Up(V_{i-1}^{\text{fused}})$ from the previous layer as follows

$$V_i^{\text{fused}} = \begin{cases} Conv(Up(V_{i-1}^{\text{fused}}) + V_i^{BEV} + V_i^{\text{vox}}) & \text{for } i > 1 \\ Conv(V_i^{BEV} + V_i^{\text{vox}}) & \text{for } i = 1 \end{cases}, \tag{1}$$

where Up denotes upsampling layer by trilinear interpolation and $Conv$ denotes a 3D convolution layer with a small kernel size. After processing through S upsampling layers, DBE ends up with Comprehensive Voxel Feature $V_{\text{CVF}} = V_S^{\text{fused}}$.

Prototype Query Decoder

As illustrated in Figure 4, PQD comprises two components: the *Scene-Adaptive Prototype Generator* (AdaPG) and the *Scene-Agnostic Prototype Generator* (AgnoPG). The AdaPG generates Scene-Adaptive Prototypes to capture the unique features of each class in the current scene. AgnoPG produces Scene-Agnostic Prototypes across diverse scenes using the EMA method (Polyak and Juditsky 1992), mitigating challenges arising from missing certain classes and capturing comprehensive features for each class. Finally, PQD predicts semantic occupancy for all voxels through a single step operation that leverages the Comprehensive Voxel Feature and the prototype-based queries.

Scene-Adaptive Prototype Generator. AdaPG aims to generate Scene-Adaptive Prototypes that encapsulate class-specific features extracted from Comprehensive Voxel Feature of the current scene. First, the AdaPG uses a shallow 3D CNN classifier to produce voxel-wise class probabilities $O_s \in \mathbb{R}^{C \times X \times Y \times Z}$, where C denotes the number of semantic categories, including the empty class. These probabilities are utilized to construct class-specific binary masks M_c^{cls} for each

class $c \in \{1, \ldots, C\}$, as follows

$$M_c^{cls}(x,y,z) = \begin{cases} 1 & \text{if } \underset{\tilde{c}\in\{1,\ldots,C\}}{\text{argmax}} \; O_s(x,y,z) = c \\ 0 & \text{otherwise} \end{cases}. \quad (2)$$

The resulting M_c^{cls} is used to sample the voxel features for the c-th class. Subsequently, the Scene-Adaptive Prototypes $\mathbf{P}^d = \{P_c^d \in \mathbb{R}^D\}_{c=1}^C$ are derived by aggregating the sampled voxel features for each class through average pooling in both x, y, and z domains

$$P_c^d = \frac{1}{N_c^{nz}} \sum_{(x,y,z)} (M_c^{cls}(x,y,z) \otimes V_{\text{CVF}}(x,y,z)), \quad (3)$$

where N_c^{nz} denotes the number of non-zero voxels in M_c^{cls} and $\otimes$ is the element-wise product. When N_c^{nz} is zero, P_c^d is set to a zero vector. The resulting $\mathbf{P}^d$ are delivered to the AgnoPG for query generation process.

Scene-Agnostic Prototype Generator. While AdaPG effectively captures class-specific features within the current scene, the absence of sampled features for certain classes results in incomplete prototypes. To address this, the AgnoPG generates Scene-Agnostic Prototypes $\mathbf{P}^g$ by applying the EMA (Polyak and Juditsky 1992) method to $\mathbf{P}^d$, continuously integrating features across diverse scenes. That is, for each iteration, $\mathbf{P}^g$ is updated as

$$\mathbf{P}^g(t) = \alpha \cdot \mathbf{P}^d(t) + (1-\alpha) \cdot \mathbf{P}^g(t-1), \quad (4)$$

where t denotes the iteration index and α is the EMA coefficient. This process ensures the generation of comprehensive prototype features encompassing all classes.

Prototype-Driven Occupancy Prediction. Scene-Aware Queries $Q^{SA} \in \mathbb{R}^{C\times D}$ are generated by combining $\mathbf{P}^d$ from AdaPG and $\mathbf{P}^g$ from AgnoPG through summation. Notably, the occupancy prediction results are obtained directly from the Scene-Aware Queries, eliminating the need for iterative Transformer decoding. The Scene-Aware Queires are processed through MLP layers to predict semantic logits p_c and mask embedding $\varepsilon_c^{\text{mask}}$ for each class c. Subsequently, the occupancy masks M_c^{occ} are generated by performing a dot product between the Comprehensive Voxel Feature and the mask $\varepsilon_c^{\text{mask}}$ along the channel dimension, followed by the application of a sigmoid function to normalize the resulting masks. Finally, the 3D semantic occupancy prediction $\mathbf{O}_s$ is obtained

$$\mathbf{O}_s = \sum_{c=1}^{C} p_c \cdot M_c^{\text{occ}}. \quad (5)$$

Our approach simplifies the decoding process by processing prototype-based queries in a single step.

Training

Robust Prototype Learning. Scene-Adaptive Prototypes $\mathbf{P}^d$ are determined by the class-specific masks $\mathbf{M}^{cls}$ obtained from AdaPG. However, when these masks are inaccurately estimated, features from voxels of incorrect classes may be erroneously included in the prototypes $\mathbf{P}^d$, resulting in a decline in overall Occupancy prediction performance.

To address this, RPL injects noise into class-specific masks $\mathbf{M}^{cls}$ to generate Noisy Scene-Adaptive Prototypes $\hat{\mathbf{P}}^d$. These prototypes are then combined with the Scene-Agnostic Prototypes $\mathbf{P}^g$ to form Noisy Scene-Aware Queries $\hat{Q}^{SA}$. Subsequently, $\hat{Q}^{SA}$ is concatenated with the original Scene-Aware Queries Q^{SA}, and these queries are used separately to predict the occupancy and class labels.

RPL introduces two types of noise to enhance the inference robustness of ProtoOcc: scaling noise and random flipping noise. Scaling noise enlarges or shrinks $\mathbf{M}^{cls}$ by a random ratio based on the ego vehicle's position, while random flipping noise randomly reallocates voxel grid classes. By injecting these perturbations, the model is trained through RPL to effectively denoise and predict occupancy. This ensures robust predictions even when the class-specific masks $\mathbf{M}^{cls}$ are inaccurately estimated during inference. This approach improves prediction robustness during inference while maintaining computational efficiency, as RPL is applied only during training.

Training Loss. The total loss $\mathcal{L}_{total}$ is given by

$$\mathcal{L}_{total} = \mathcal{L}_{depth} + \mathcal{L}_{AdaPG} + \mathcal{L}_{occ} + \mathcal{L}_{RPL}, \quad (6)$$

where $\mathcal{L}_{depth}$ is for depth estimation, $\mathcal{L}_{AdaPG}$ is for class-specific mask prediction in AdaPG, $\mathcal{L}_{occ}$ is for query-based occupancy prediction, and $\mathcal{L}_{RPL}$ is for the Robust Prototype Learning. Specifically, $\mathcal{L}_{depth}$ employs cross-entropy (CE) loss using LiDAR point clouds projected onto the image. $\mathcal{L}_{AdaPG}$ includes Lovasz (Berman, Triki, and Blaschko 2018) and Dice (Sudre et al. 2017) losses for class-specific mask prediction used in Scene-Adaptive Prototypes generation. Note that $\mathcal{L}occ$ is computed without employing a bipartite matching process, as the prototypes are directly assigned to each class. This loss combines cross-entropy (CE) loss for classification with focal loss (Lin et al. 2017) and dice loss for mask prediction. Similarly, $\mathcal{L}_{RPL}$ applies the same functions as $\mathcal{L}_{occ}$ to the $\hat{Q}^{SA}$ introduced in RPL.

4 Experiments

Experimental Settings

Dataset and Metrics. We conducted the experiments on the Occ3D dataset (Tian et al. 2024), which evaluates the mean Intersection over Union (*mIoU*) across 17 classes. Additionally, we measured the latency of our model.

Implementation Details. We utilized ResNet-50 (He et al. 2016) for the image backbone network. In DBE, the voxel branch uses a kernel size of 3 for 3D convolution, while the BEV branch employs a kernel size of 7 for 2D convolution. Our model was trained for 24 epochs with a total batch size of 16 on 4 NVIDIA RTX 3090 GPUs. The AdamW optimizer was used with a learning rate of 4×10^{-4} for single-frame and 2×10^{-4} for multi-frame.

Performance Comparison

Table 1 presents a detailed comparison of single-frame methods on the Occ3D-nuScenes validation set, demonstrating our method's superior performance. ProtoOcc, utilizing the

Method	Venue	Image Backbone	Image Size	mIoU (%)	Latency (ms)
MonoScene(Cao and De Charette 2022)	CVPR'22	ResNet-101	928 × 1600	6.06	830.1
TPVFormer(Huang et al. 2023)	CVPR'23	ResNet-101	928 × 1600	27.83	320.8
Vampire(Xu et al. 2024)	AAAI'24	ResNet-101	256 × 704	28.30	349.2
CTF-Occ(Tian et al. 2024)	NIPS'23	ResNet-101	928 × 1600	28.53	-
SurroundOcc(Wei et al. 2023)	ICCV'23	ResNet-101	800 × 1333	34.40	355.6
BEVDet(Huang et al. 2021)	arXiv'21	ResNet-50	256 × 704	19.38	-
OccFormer(Zhang, Zhu, and Du 2023)	ICCV'23	ResNet-50	928 × 1600	21.93	349.2
COTR* (Ma et al. 2024)	CVPR'24	ResNet-50	256 × 704	37.02	168.9
FB-Occ(Li et al. 2023b)	ICCV'23	ResNet-50	256 × 704	37.39	129.7
Ours	-	ResNet-50	256 × 704	**39.56**	**77.9**

Table 1: Comparison of different single-frame 3D occupancy prediction methods when evaluated on the Occ3D-nuScenes validation set. Latency is measured on a single NVIDIA RTX 3090 GPU. - denotes that the results are not in public. † indicates that the latency was measured on an NVIDIA V100 GPU. ∗ indicates results reproduced using publicly available codes.

Method	Venue	Image Backbone	Image Size	mIoU
BEVFormer	ECCV'22	ResNet-101	928×1600	26.88
FastOcc	ICRA'24	ResNet-101	640×1600	39.21
PanoOcc	CVPR'24	ResNet-101	864×1600	42.13
BEVDet4D	arXiv'21	ResNet-50	384×704	39.25
FB-Occ	ICCV'23	ResNet-50	256×704	40.69
COTR	CVPR'24	ResNet-50	256×704	44.45
Ours	-	ResNet-50	256×704	**45.02**

Table 2: Comparison of different multi-frame 3D occupancy prediction methods when evaluated on the Occ3D-nuScenes validation set.

DBE	PQD	RPL	mIoU	Latency (ms)
			34.18	**60.0**
✓			35.87 (+1.69)	75.7
	✓		35.63 (+1.45)	61.1
✓	✓		37.05 (+2.87)	77.9
✓	✓	✓	**37.45** (+3.27)	77.9

Table 3: Ablation study for evaluating the main components of ProtoOcc.

ResNet-50 backbone, achieves a performance of 39.56% *mIoU*, outperforming all other methods (Zhang, Zhu, and Du 2023; Huang et al. 2023; Wei et al. 2023; Xu et al. 2024), including those employing the larger ResNet-101 backbone. Notably, ProtoOcc achieves an inference time of 77.9 ms, demonstrating a 1.7× faster speed compared to the previous state-of-the-art method, while also achieving a remarkable performance improvement of 2.17% in *mIoU*. These results demonstrate that ProtoOcc achieves both high efficiency and superior accuracy, making it well-suited for real-time applications.

We also adopt multi-frame methods for ProtoOcc, fusing eight consecutive voxel features over time. Following BEVDet4D (Huang et al. 2021), these voxel features are concatenated along the channel dimension and processed through a residual block followed by a $1 \times 1 \times 1$ convolution layer to reduce the channel dimensionality. Table 2 provides a performance comparison with other multi-frame methods evaluated on the Occ3D-nuScenes validation set (Tian et al. 2024). ProtoOcc establishes a new state-of-the-art performance, exhibiting substantial improvements over existing methods (Li et al. 2022; Hou et al. 2024; Wang et al. 2024; Huang et al. 2021; Li et al. 2023b) and surpassing the previous best model, COTR (Ma et al. 2024), by 0.57% in *mIoU*.

Ablation Study

We performed an ablation study to evaluate the contributions of the components proposed in ProtoOcc. We trained on a quarter of the dataset for 24 epochs and evaluated the entire validation set using a ResNet-50 backbone (He et al. 2016) with a 256×704 resolution and a single frame.

Contributions of Main Components. Table 3 shows the impact of our main modules. The first row denotes a baseline employing 3D CNNs with small kernel sizes for both the encoder and the decoder. When adding DBE into the baseline, we demonstrate a notable 1.69% increase in *mIoU*. This improvement shows that DBE effectively integrates long-range spatial relationships by expanding the receptive field in the BEV domain while capturing fine-grained 3D structures in the voxel domain. We integrated PQD into the baseline, achieving a 1.45% *mIoU* improvement while maintaining latency. This demonstrates that PQD effectively captures class distributions through prototypes, enhancing performance without iterative query decoding. Incorporating both DBE and PQD surpasses the baseline by 2.87% in *mIoU*. RPL improves the *mIoU* by an additional 0.4%, reducing the impact of inaccurate class-specific masks.

Contributions of Dual Branch Encoder. Table 4 presents the results of the ablation study conducted on the Dual Branch Encoder. We focus on the impact of varying kernel sizes within the voxel and BEV branches. We tried kernel sizes of 3 and 7 in the voxel branch, as shown in Table 4. Using a kernel size of 7 in scenario (b) resulted in significant latency increases due to the high dimensionality of 3D space. In contrast, increasing the kernel size within the BEV domain, as demonstrated in scenario (d), led to a comparatively minor latency increase when contrasted with scenario (c). Scenario (b), with its larger voxel branch kernel size, delivered marginally better perfor-

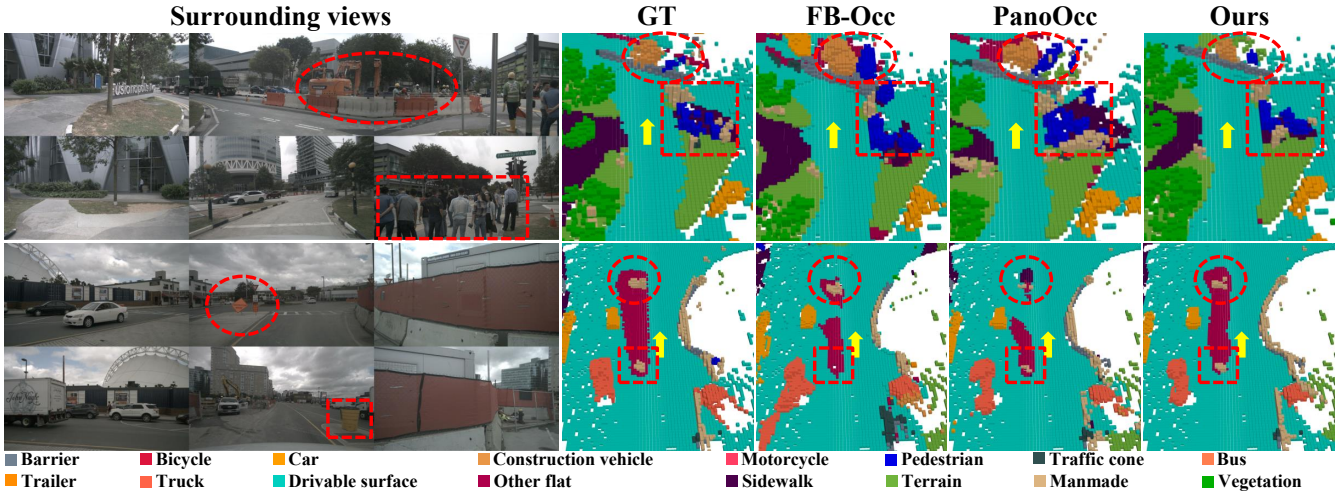

Figure 5: Qualitative results on the Occ3D-nuScenes validation set. The regions marked by red ellipses and rectangles emphasize the superior results generated by our proposed model. The yellow arrow indicates the position and direction of the ego vehicle.

Branch	Model	Voxel Kernel	BEV Kernel	MS Fusion	mIoU	Latency (ms)
Voxel Only	(a)	3			35.71	60.7
	(b)	7			36.14	87.2
BEV Only	(c)		3		35.64	**52.0**
	(d)		7		36.07	55.1
Dual Branch	(e)	3	3		36.70	71.1
	(f)	3	3	✓	36.93	74.5
	Ours, (g)	3	7	✓	**37.45**	77.9

Table 4: Ablation study for Dual Branch Encoder. *MS Fusion* indicates the use of multi-scale fusion in HFM.

mance than scenario (d) in the BEV branch.

Further enhancements were observed when the voxel and BEV branches were combined, as seen in scenarios (e) through (g). Specifically, setting the kernel sizes to 3 for the voxel branch and 7 for the BEV branch, and incorporating multi-scale fusion, not only outperformed scenario (b) in terms of performance but also maintained lower latency. The multi-scale fusion alone contributed an increase of 0.23% in *mIoU* compared to scenario (e), while our specific configuration provided an additional improvement of 0.52% in *mIoU*.

Impact of the Prototype Query Decoder. Table 5 presents a comparison of different decoder types, assessing their performance in terms of *mIoU* and latency. While a query-based decoder (Zhang, Zhu, and Du 2023) yields higher performance compared to a CNN-based decoder, it incurs much higher latency due to their iterative decoding process. Conversely, when utilizing AdaPG and AgnoPG without iterative decoding, not only do they surpass the query-based decoder by an additional 0.79% in *mIoU*, but they also achieve a substantial reduction in latency, amounting to 73.7ms.

Decoder Type	AdaPG	AgnoPG	Iterative Decoding	mIoU	Latency (ms)
CNN-based				35.87	**76.1**
Query-based			✓	36.66	151.6
PQD	✓			36.87	77.4
	✓	✓		**37.45**	77.9

Table 5: Comparison of different decoder types.

Qualitative Analysis

Figure 5 presents qualitative results on the Occ3D-nuScenes validation set, comparing the proposed model with FB-Occ (Li et al. 2023b) and PanoOcc (Wang et al. 2024). ProtoOcc provides accurate predictions in complex scenes, particularly for regions with ambiguous boundaries and diverse object types.

5 Conclusions

In this paper, we introduced ProtoOcc, an efficient encoder-decoder framework designed for 3D occupancy prediction. The DBE leverages both voxel and BEV representations, capturing fine-grained interactions and efficiently modeling long-range spatial relationships to enhance encoder performance. Furthermore, the PQD employs Scene-Adaptive and Scene-Agnostic Prototypes as queries, which eliminate the need for an iterative decoding process, thereby significantly reducing computational complexity. We also introduced the RPL to increase the model's robustness against inaccuracies in prototypes. Our method achieved state-of-the-art performance with faster inference speeds on the Occ3D-nuScenes benchmark.

Acknowledgements

This work was partly supported by Institute of Information & communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) [NO.RS-2021-II211343, Artificial Intelligence Graduate School Program (Seoul National University)], the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (No.2020R1A2C2012146), and the National Research Foundation (NRF) funded by the Korean government (MSIT) (No. RS-2024-00421129).

References

Berman, M.; Triki, A. R.; and Blaschko, M. B. 2018. The lovász-softmax loss: A tractable surrogate for the optimization of the intersection-over-union measure in neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 4413–4421.

Cao, A.-Q.; Dai, A.; and de Charette, R. 2024. Pasco: Urban 3d panoptic scene completion with uncertainty awareness. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14554–14564.

Cao, A.-Q.; and De Charette, R. 2022. Monoscene: Monocular 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 3991–4001.

Chen, Y.; Liu, J.; Zhang, X.; Qi, X.; and Jia, J. 2023. Largekernel3d: Scaling up kernels in 3d sparse cnns. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 13488–13498.

Ding, X.; Zhang, X.; Han, J.; and Ding, G. 2022. Scaling up your kernels to 31x31: Revisiting large kernel design in cnns. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11963–11975.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. 2020. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 770–778.

Hou, J.; Li, X.; Guan, W.; Zhang, G.; Feng, D.; Du, Y.; Xue, X.; and Pu, J. 2024. FastOcc: Accelerating 3D Occupancy Prediction by Fusing the 2D Bird's-Eye View and Perspective View. *arXiv preprint arXiv:2403.02710*.

Huang, J.; Huang, G.; Zhu, Z.; Ye, Y.; and Du, D. 2021. Bevdet: High-performance multi-camera 3d object detection in bird-eye-view. *arXiv preprint arXiv:2112.11790*.

Huang, Y.; Zheng, W.; Zhang, Y.; Zhou, J.; and Lu, J. 2023. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9223–9232.

Li, Y.; Yu, Z.; Choy, C.; Xiao, C.; Alvarez, J. M.; Fidler, S.; Feng, C.; and Anandkumar, A. 2023a. Voxformer: Sparse voxel transformer for camera-based 3d semantic scene completion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9087–9098.

Li, Z.; Wang, W.; Li, H.; Xie, E.; Sima, C.; Lu, T.; Qiao, Y.; and Dai, J. 2022. Bevformer: Learning bird's-eye-view representation from multi-camera images via spatiotemporal transformers. In *European conference on computer vision*, 1–18. Springer.

Li, Z.; Yu, Z.; Austin, D.; Fang, M.; Lan, S.; Kautz, J.; and Alvarez, J. M. 2023b. Fb-occ: 3d occupancy prediction based on forward-backward view transformation. *arXiv preprint arXiv:2307.01492*.

Lin, T.-Y.; Goyal, P.; Girshick, R.; He, K.; and Dollár, P. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, 2980–2988.

Liu, H.; Wang, H.; Chen, Y.; Yang, Z.; Zeng, J.; Chen, L.; and Wang, L. 2023. Fully sparse 3d panoptic occupancy prediction. *arXiv preprint arXiv:2312.17118*.

Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; and Guo, B. 2021. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF international conference on computer vision*, 10012–10022.

Liu, Z.; Mao, H.; Wu, C.-Y.; Feichtenhofer, C.; Darrell, T.; and Xie, S. 2022. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 11976–11986.

Luo, W.; Li, Y.; Urtasun, R.; and Zemel, R. 2016. Understanding the effective receptive field in deep convolutional neural networks. *Advances in neural information processing systems*, 29.

Ma, Q.; Tan, X.; Qu, Y.; Ma, L.; Zhang, Z.; and Xie, Y. 2024. Cotr: Compact occupancy transformer for vision-based 3d occupancy prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 19936–19945.

Philion, J.; and Fidler, S. 2020. Lift, splat, shoot: Encoding images from arbitrary camera rigs by implicitly unprojecting to 3d. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, 194–210. Springer.

Polyak, B. T.; and Juditsky, A. B. 1992. Acceleration of stochastic approximation by averaging. volume 30, 838–855. SIAM.

Sudre, C. H.; Li, W.; Vercauteren, T.; Ourselin, S.; and Jorge Cardoso, M. 2017. Generalised dice overlap as a deep learning loss function for highly unbalanced segmentations. In *Deep Learning in Medical Image Analysis and Multimodal Learning for Clinical Decision Support: Third International Workshop, DLMIA 2017, and 7th International Workshop, ML-CDS 2017, Held in Conjunction with MICCAI 2017, Québec City, QC, Canada, September 14, Proceedings 3*, 240–248. Springer.

Tang, P.; Wang, Z.; Wang, G.; Zheng, J.; Ren, X.; Feng, B.; and Ma, C. 2024. Sparseocc: Rethinking sparse latent representation for vision-based semantic occupancy prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 15035–15044.

Tian, X.; Jiang, T.; Yun, L.; Mao, Y.; Yang, H.; Wang, Y.; Wang, Y.; and Zhao, H. 2024. Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving. *Advances in Neural Information Processing Systems*, 36.

Tong, W.; Sima, C.; Wang, T.; Chen, L.; Wu, S.; Deng, H.; Gu, Y.; Lu, L.; Luo, P.; Lin, D.; et al. 2023. Scene as occupancy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 8406–8415.

Wang, X.; Zhu, Z.; Xu, W.; Zhang, Y.; Wei, Y.; Chi, X.; Ye, Y.; Du, D.; Lu, J.; and Wang, X. 2023. Openoccupancy: A large scale benchmark for surrounding semantic occupancy perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 17850–17859.

Wang, Y.; Chen, Y.; Liao, X.; Fan, L.; and Zhang, Z. 2024. Panoocc: Unified occupancy representation for camera-based 3d panoptic segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 17158–17168.

Wei, Y.; Zhao, L.; Zheng, W.; Zhu, Z.; Zhou, J.; and Lu, J. 2023. Surroundocc: Multi-camera 3d occupancy prediction for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 21729–21740.

Xu, J.; Peng, L.; Cheng, H.; Xia, L.; Zhou, Q.; Deng, D.; Qian, W.; Wang, W.; and Cai, D. 2024. Vampire: Regulating Intermediate 3D Features for Vision-Centric Autonomous Driving. *Proceedings of the AAAI Conference on Artificial Intelligence*.

Yan, H.; Li, Z.; Li, W.; Wang, C.; Wu, M.; and Zhang, C. 2021. ConTNet: Why not use convolution and transformer at the same time? *arXiv preprint arXiv:2104.13497*.

Yu, Z.; Shu, C.; Deng, J.; Lu, K.; Liu, Z.; Yu, J.; Yang, D.; Li, H.; and Chen, Y. 2023. Flashocc: Fast and memory-efficient occupancy prediction via channel-to-height plugin. *arXiv preprint arXiv:2311.12058*.

Zhang, H.; Yan, X.; Bai, D.; Gao, J.; Wang, P.; Liu, B.; Cui, S.; and Li, Z. 2024. Radocc: Learning cross-modality occupancy knowledge through rendering assisted distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 7060–7068.

Zhang, Y.; Zhu, Z.; and Du, D. 2023. Occformer: Dual-path transformer for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9433–9443.

Zhao, L.; Xu, X.; Wang, Z.; Zhang, Y.; Zhang, B.; Zheng, W.; Du, D.; Zhou, J.; and Lu, J. 2024. LowRankOcc: Tensor Decomposition and Low-Rank Recovery for Vision-based 3D Semantic Occupancy Prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 9806–9815.

Zhou, Q.; Cao, J.; Leng, H.; Yin, Y.; Kun, Y.; and Zimmermann, R. 2024. SOGDet: Semantic-occupancy guided multi-view 3D object detection. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 7668–7676.