PMAFusion: Projection-Based Multi-Modal Alignment for 3D Semantic Occupancy Prediction

Shiyao Li[1] Wenming Yang[1,2,*] Qingmin Liao[1,2]
[1] Shenzhen International Graduate School, Tsinghua University, China
[2] Department of Electronic Engineering, Tsinghua University, China
lishiyao21@mails.tsinghua.edu.cn, {yang.wenming, liaoqm}@sz.tsinghua.edu.cn

Abstract

3D Semantic Occupancy Prediction offers a holistic scene understanding with both spatial structure and semantic analysis. Current research in this field primarily focuses on single-modal inputs, relying either on images or point cloud data. The potential of combining the complementary attributes of images and point clouds has not been fully explored. Previous method transforms image features into 3D space for direct concatenation with monocular depth estimation, which may introduce noises due to inaccurate depth prediction. It could also lead to substantial memory usage for explicitly constructing dense image feature volumes. To this end, we propose PMAFusion, an effective fusion module based on accurate multi-modal alignment. We first project the point cloud onto images using camera parameters, thereby aligning each voxel with its associated pixels. A cross-attention module is then used to adaptively fuse voxel-pixel features for improved representation. In order to handle empty voxels that are difficult to obtain aligned pixels naturally, we generate reference points through uniform sampling to supplement the missing spatial information. With PMAFusion, We yield the best results on the nuScenes-Occupancy dataset and conduct thorough experiments to evaluate the effectiveness and efficiency of our proposed method.

1. Introduction

With the rapid development of autonomous driving, the importance of 3D environmental perception has greatly increased. 3D Semantic Occupancy takes this a step further than traditional 3D object detection. Rather than simply identifying objects and their bounding boxes, it comprehensively understands the space they occupy and their semantic attributes. Vital environmental information can be gained through this detailed perception, thus ensuring driving safety and efficiency in diverse conditions.

3D semantic occupancy prediction needs to determine whether each voxel in a scene is occupied and categorize it with a semantic label. Identifying occupied voxels facilitates the detection of drivable areas, whereas the semantic labeling contributes to a more holistic scene understanding. However, publicly available occupancy datasets are mostly for indoor scenes [6, 17, 18]. In recent years, with the growing interest in outdoor tasks, occupancy datasets focusing on outdoor scenes have gradually developed. Occ3D-nuScenes[19] and nuScenes-Occupancy[21] provide annotations for the NuScenes dataset in the realm of semantic occupancy. Different from SemanticKITTI[1]'s approach of front-view occupancy, their focus is on surrounding occupancy, assessing the area around an autonomous vehicle to better meet the demands of outdoor applications.

Most existing semantic occupancy algorithms are of single-modal input, each struggling with inherent limitations. Images are generally considered to be rich in color and texture, which is highly beneficial for extracting semantic information. Yet, in the context of semantic occupancy prediction within 3D environments, they fall short in providing direct and accurate depth information. Algorithms based on images[3, 10, 13] often require monocular depth estimation for spatial information, a notoriously difficult and imprecise task. Methods that use point cloud as the single input[15, 16, 23] rely heavily on 3D structural feature. However, one of point clouds' primary drawbacks is their inherent sparsity, which often leads to representations that lack detail, especially when it concerns objects that are distant or of small size. Notably, CONet[21] was the first to propose a multi-modal fusion model to leverage the strengths and compensate for the weaknesses of both modalities. Their results underscore the effectiveness of such fusion, highlighting the advantages of multi-modal inputs in 3D semantic occupancy applications.

In this study, we build upon the nuScenes-Occupancy dataset and [21]'s multi-modal baseline to design an im-

*Corresponding author

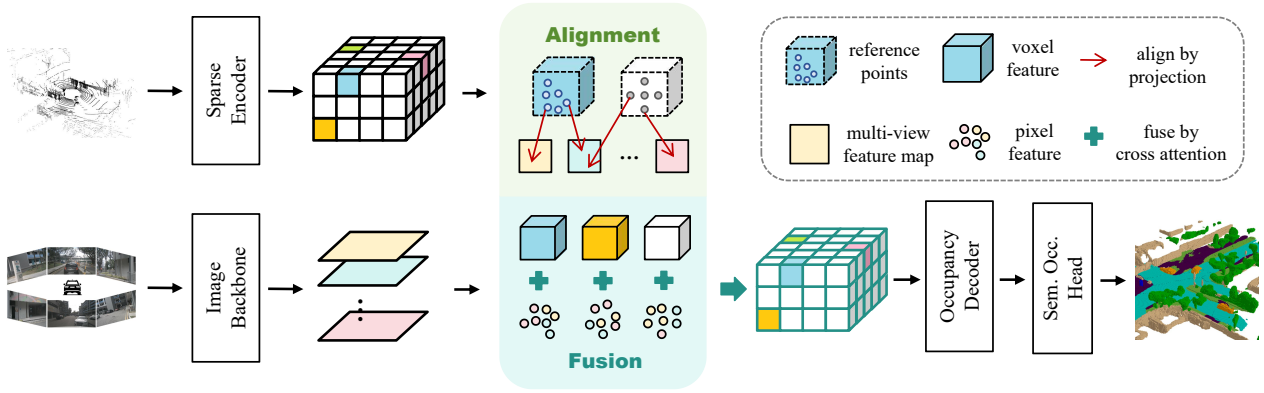

Figure 1. The architecture of our network is shown above. We begin by employing a 3D encoder and an image backbone to gather features from point cloud and multi-view images. The colored blocks in the figure represent voxels filled with point cloud data, whereas the white blocks indicate empty voxels. We then project reference points onto images to extract corresponding pixel features, thereby aligning voxel and image features. Reference points for non-empty voxels come directly from point cloud data. For empty voxels, they are obtained through spatial sampling. This is followed by the fusion of each voxel feature with its corresponding pixel features to create a unified feature volume. The final semantic occupancy output is derived from an occupancy decoder and a semantic occupancy head.

proved fusion module. [21] uses depth estimation to lift camera features to 3D space, then employs convolutional modules for fusion. Our approach uses camera projection for direct alignment of 3D voxel features with 2D pixel features. The advantages of our method are as follows: (1) We construct voxel-pixel alignment by camera projection instead of monocular depth estimation and back projection, bypassing the accuracy issues inherent in this ill-posed problem as well as boosting computational efficiency. (2) For empty voxels, we design an uniform reference points generation process to ensure reliable spatial representation. (3) We achieve the new SOTA for multi-modal models on the nuscenes-Occupancy dataset with our design and conduct thorough ablation experiments to prove its efficiency.

2. Related Works

2.1. LiDAR-based Methods

Point clouds are widely used in 3D tasks due to their ability to provide direct depth information and their robustness to lighting variations[15, 16, 22, 23, 27]. For instance, JS3CNet[23] designs a multi-scale semantic scene completion module, leveraging semantic occupancy prediction to enhance the accuracy of point cloud semantic segmentation. SPCNet[27] interprets complex scenes with local deep implicit functions, ensuring accurate predictions across varied environments. Despite its suitability for 3D tasks, point cloud generally requires more computational resources, especially when processing large-scale scenes. To counter this, LMSCNet[16] proposes a lightweight 3D model that handles point cloud data efficiently, thus providing insight into real-time semantic occupancy prediction.

Nonetheless, point clouds face certain drawbacks. Their typical sparsity results in less detailed representations of distant and small objects, thereby posing challenges for accurate prediction.

2.2. Camera-based Methods

Images offer rich semantic information but lack direct spatial geometry. Compared to point clouds, images can be captured and annotated at a lower cost and more readily integrated into practical systems. Consequently, many studies focus on utilizing multi-view images for predicting 3D occupancy[8, 20, 24, 26]. MonoScene[3] is the first to use monocular RGB images for deducing a scene's geometric and semantic properties. It combines 2D and 3D-UNet, bridging the divide between 2D images and 3D scene understanding. VoxFormer[13], in contrast to MonoScene's CNN-based approach, employs an attention-based architecture and depth estimation to construct feature volumes, boosting model performance. TPVFormer[10] further extends this concept by adopting a tri-perspective view for 3D feature, preserving comprehensive feature representation while reducing computational costs. OccFormer[26] builds around a novel architecture that processes 3D volume data through a dual-path transformer network and efficiently adapts the decoder of Mask2Former[5] for 3D semantic occupancy prediction.

In summary, while images are a valuable source of dense semantic information, their limitation in conveying spatial geometry still requires innovative algorithms to compensate.

2.3. Multi-modal Methods

Currently, RGB-D input forms the basis of most multi-modal occupancy prediction methods[7, 9, 11, 12, 14, 25].

CCPNet[25] fuses RGB and depth images before feature extraction, utilizing flipped TSDF(Truncated Signed Distance Function) for 3D spatial representation. In contrast, AIC-Net[12] processes RGB and depth images separately for feature extraction and combines them post back-projection into 3D space. SISNet[2] introduces an innovative framework that focuses on enhancing the semantic completion accuracy of 3D scenes by fusing instance-level and scene-level semantic information.These works are conducted on indoor datasets, because the limited range of RGB-D sensors makes them unsuitable for outdoor environments.

To address outdoor scenarios, CONet[21] first proposes a multi-modal solution, back-projecting surrounding image features into 3D space based on monocular depth estimations and integrates these with point cloud features through 3D convolution.

3. Method Description

3.1. Overview

Our model's architecture, as illustrated in Figure 1, builds upon the multi-modal baseline in [21]. We use ResNet50 to extract multi-view image features and a 3D sparse encoder to obtain voxel features for point cloud. This encoder consists of submanifold 3d convolutions and sparse 3d convolutions, downsampling the original voxelized feature by a factor of 8. The alignment between each voxel (both empty and non-empty) and image pixels is established based on reference points and camera projection. Next, we feed the features of both modalities into the fusion module. Here, voxel feature acts as the query while the aligned pixel features serve as the key and value. Through cross-attention mechanisms, we fuse these to obtain enhanced features. These features are then processed by the occupancy encoder for further refinement, with the occupancy head outputs the corresponding category for each voxel.

In Section 3.2, we offer a comprehensive description of our proposed fusion module. How we obtain reference points for empty voxels is then explained in detail in Section 3.3.

3.2. Accurate Alignment and Efficient Fusion

In the fusion module of CONet[21], monocular depth estimation is used to transform 2D image features into 3D counterparts. These 3D image features are then concatenated with voxel features, and a 3D convolution with kernel size of $1 \times 1 \times 1$ is used to complete the fusion. Although effective, this method has its drawbacks. Inaccuracies in monocular depth estimation can cause misalignments between image and voxel features in the spatial domain. Furthermore, each feature grid has a spatial size of 0.8m, covering a considerable range in both the point cloud and the original image. This method of direct concatenation fails to account for the distribution of points within these grids.

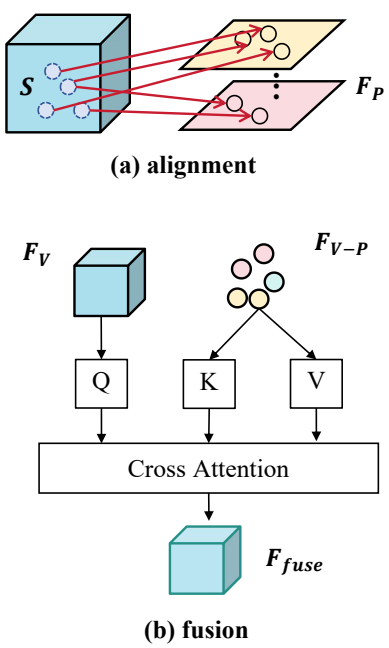

Figure 2. Our fusion module is depicted in the schematic above. First, in (a) alignment process, the set of voxel reference points S is projected onto the multi-view image features F_P. Bilinear interpolation is then applied to sample corresponding pixel features for each voxel, labeled as F_{V-P}. In the (b) fusion stage, we use the voxel feature F_V as the query and the associated pixel features F_{V-P} as key and value. These are then fed into a cross attention module, resulting in the fused feature F_{fuse}.

Cross attention is frequently employed to fuse diverse input features. However, when dealing with multi-view images from N_{view} viewpoints, and considering N_{voxel} voxels with image features of size $[H \times W \times C]$, applying cross attention on global image features results in a computational complexity of $O(N_{voxel} N_{view} HWC)$. This would lead to a significant computational load and potentially hinder the network's convergence. A more efficient method is to fuse each voxel with local image features that are highly correlated with it, which requires establishing a specific alignment between voxels and image regions.

We construct accurate alignment by projecting reference points onto multi-view images and fuse features using cross attention, the process of which can be seen in Figure 2. This method differs significantly from the ill-posed approach of using monocular depth estimation for image-to-voxel alignment. By employing camera parameters, we ensure a deterministic process, thereby reducing alignment errors.

Let $F_P = \{F_P^i\}_{i=1}^{N_{view}}$ represents the multi-view image features, where N_{view} denotes the number of cameras.

Each voxel V possesses a feature F_V and a set of reference points $S = \{s_j\}_{j=1}^{N_{ref}}$, where N_{ref} represents the number of reference points in a voxel. We introduce the operation $\phi_i(S)$, which projects each point in S onto the i^{th} image, sampling corresponding pixels feature based on bilinear interpolation. Points that fail to project are discarded. The process can be represented by the following equation:

$$\phi_i(S) = \{F^i_{V-P} = \mathcal{G}(\mathcal{P}^i(s), F^i_P), s \in S\}, \quad (1)$$

where $\mathcal{P}^i(s)$ represents the projection of a 3D spatial point s the image plane with the i^{th} camera's parameters, producing pixel coordinates $p = (u, v)$. The operation $\mathcal{G}(p, F^i_P)$ refers to the grid sampling technique, which leverages bilinear interpolation to extract the pixel feature at p on the i^{th} image feature map F^i_P. The term F^i_{V-P} denotes the set of pixel features that correspond to the voxel in the i^{th} image.

Hence, the set of pixel features F_{V-P} corresponding to F_V across all N_{view} image features can be written as:

$$F_{V-P} = \bigcup_{i=1}^{N_{view}} \phi_i(S). \quad (2)$$

We use F_V as the query, and F_{V-P} as the key and value. These elements are fed into cross attention, denoted as CA, producing the fused feature F_{fuse}:

$$F_{fuse} = CA(F_V, F_{V-P}, F_{V-P}). \quad (3)$$

3.3. Uniform Reference Points Generation

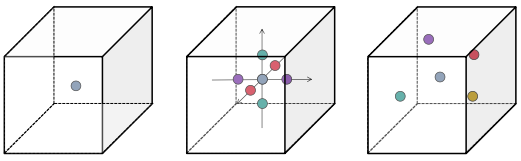

Figure 3. Three sampling techniques are shown as above. The white blocks represent empty voxels, and the colored dots denote the positions of the sampling points.

We assign each voxel with a set of reference points and use them for future aligned fusion. For non-empty voxels containing spatial points after voxelization, these spatial points themselves serve as references, embodying valuable voxel information. However, given the point cloud's sparsity, numerous voxels lack spatial points. For empty voxels, we uniformly sample 3D points within them, using these sampled points as references.

We experiment with these three sampling methods, each depicted in Figure 3: Center point, uniform sampling, and random sampling. The center point method involves selecting the voxel's center as the sampling point. Uniform sampling extends this by including six additional midpoints along the $\pm x, \pm y, \pm z$ directions, alongside the center point. For random sampling, we generate multiple random offsets, using these offsets to derive new sampling coordinates.

Our final choice is uniform sampling because it efficiently balances computational load while covering the entire voxel. Through cross attention, features from all sampling locations can be aggregated adaptively, resulting in a more comprehensive representation of the voxel.

4. Experiment

4.1. Implementation Details

Dataset: We conduct experiments on nuScenes-Occupancy[21], a newly proposed surround-view semantic occupancy dataset. It builds upon the nuScenes point cloud segmentation annotations and extends the classic nuScenes dataset by adding dense semantic occupancy annotations, aiming to create a large-scale surrounding semantic occupancy prediction dataset. nuScenes-Occupancy consists of 28,130 training frames and 6,019 validation frames, with 17 semantic labels including drivable areas, pedestrians and vehicles, which are consistent with those used in segmentation tasks.

Algorithm 1 Custom Segmented Attention

Input: Flattened voxel feature $f_{vox} \in \mathbb{R}^{N \times C}$; Flattened pixel feature $f_{pix} \in \mathbb{R}^{N \times C}$; Segment indices $I_{grid} \in \mathbb{R}^{N \times 1}$.

while training **do**
  $W_q, W_k, W_v \in \mathbb{R}^{C \times C}$
  $Q, K, V = W_q f_{vox}, W_k f_{pix}, W_v f_{pix}$
  $Attn = \text{einsum}('nc, nc \to n', Q, K)/\sqrt{C}$
  $Attn = \text{scatter\_softmax}(Attn, I_{grid})$
  $O = \text{einsum}('nc, n \to nc', V, Attn)$
  $f_{fuse} = \text{scatter\_add}(O, I_{grid})$
end while

Output: Fused feature $f_{fuse} \in \mathbb{R}^{M \times C}$.

Custom Segmented Attention: When implementing the PMA fusion module, we opt not to use the pre-defined attention operation in *torch*, but instead implement custom cross attention using *torch_scatter*. This decision was informed by the fact that *torch.nn.MultiheadAttention* demands the spatial alignment of features, which brings unnecessary computational and memory costs. Scatter operators, however, compute based on given indices, ensuring a sparse feature representation. This aligns more closely with the design philosophy of PMA.

Sparse representation of 3D features usually consists of an $N \times C$ flattened feature vector and an $N \times 4$ location vector, where the location is specified as $[b, h, w, d]$,

Table 1. Results on nuscenes-Occupancy val set

Methods	Input	Surround	mIoU	barrier	bicycle	bus	car	const. veh.	motorcycle	pedestrian	traffic cone	trailer	truck	drive. sur.	other flat	sidewalk	terrain	manmade	vegetation
MonoScene[3]	C	✗	6.9	7.1	3.9	9.3	7.2	5.6	3.0	5.9	4.4	4.9	4.2	14.9	6.3	7.9	7.4	10.0	7.6
TPVFormer[10]	C	✓	7.8	9.3	4.1	11.3	10.1	5.2	4.3	5.9	5.3	6.8	6.5	13.6	9.0	8.3	8.0	9.2	8.2
3DSketch[4]	C&D	✗	10.7	12.0	5.1	10.7	12.4	6.5	4.0	5.0	6.3	8.0	7.2	21.8	14.8	13.0	11.8	12.0	21.2
AICNet[12]	C&D	✗	10.6	11.5	4.0	11.8	12.3	5.1	3.8	6.2	6.0	8.2	7.5	24.1	13.0	12.8	11.5	11.6	20.2
LMSCNet[16]	L	✓	11.5	12.4	4.2	12.8	12.1	6.2	4.7	6.2	6.3	8.8	7.2	24.2	12.3	16.6	14.1	13.9	22.2
JS3C-Net[23]	L	✓	12.5	14.2	3.4	13.6	12.0	7.2	4.3	7.3	6.8	9.2	9.1	27.9	15.3	14.9	16.2	14.0	**24.9**
C-Baseline[21]	C	✓	10.3	9.9	6.8	11.2	11.5	6.3	8.4	8.6	4.3	4.2	9.9	22.0	15.8	14.1	13.5	7.3	10.2
L-Baseline[21]	L	✓	11.7	12.2	4.2	11.0	12.2	8.3	4.4	8.7	4.0	8.4	10.3	23.5	16.0	14.9	15.7	15.0	17.9
M-Baseine[21]	L&C	✓	15.1	14.3	12.0	15.2	14.9	13.7	15.0	13.1	9.0	10.0	14.5	23.2	17.5	16.1	17.2	15.3	19.5
Ours	L& C	✓	16.9	15.9	14.9	15.7	17.4	14.6	17.9	16.5	10.1	10.4	15.6	26.3	19.3	19.2	18.1	17.1	21.2
C-CONet[21]	C	✓	12.8	13.2	8.1	15.4	17.2	6.3	11.2	10.0	8.3	4.7	12.1	31.4	18.8	18.7	16.3	4.8	8.2
L-CONet[21]	L	✓	15.8	17.5	5.2	13.3	18.1	7.8	5.4	9.6	5.6	13.2	13.6	**34.9**	21.5	22.4	21.7	19.2	23.5
M-CONet[21]	L&C	✓	20.1	23.3	13.3	21.2	24.3	15.3	15.9	18.0	13.3	15.3	20.7	33.2	21.0	22.5	21.5	19.6	23.2
Ours	L&C	✓	**21.9**	**24.7**	**15.9**	**22.5**	**25.9**	**15.5**	**21.6**	**23.7**	**15.2**	**16.3**	**21.6**	**34.9**	**21.9**	**24.0**	**23.1**	**20.4**	23.6

representing the batch index and 3D grid indices respectively. In PMA, each voxel is associated with multiple reference points. We repeat and reshape the voxel features and their location vectors according to the number of reference points, ensuring they are correctly matched after flattening. Bilinear interpolation is then used to extract pixel features from image features. Using *torch.unique()* on the location vector with the *return_inverse* option enabled can conveniently retrieve segment indices. Features sharing the same location are assigned the same inverse index, simplifying the process of grouping pixel features by their voxel association. The implementation of segmented attention is shown in Algorithm(1). The original length of voxel feature is $M = H \times W \times D$ and the total number of reference points is N.

Experimental setting: In our experiments, we follow a setup similar to [21]. The input image size is 1600×900, and the point cloud's range is $[-51.2m, 51.2m]$ in the X, Y directions and $[-5m, 3m]$ in the Z direction. For training, we employ a loss function that includes cross-entropy loss L_{ce}, lovasz-softmax loss L_{ls}, affinity loss[3] L^{geo}_{scal} and L^{sem}_{scal}. The overall loss is calculated as the sum of these individual losses.

We replace the original fusion module with our design in M-Baseline and M-CONet from [21]. We use AdamW as the optimizer with a weight decay of 0.01. Training is conducted on 8 A100 GPUs, utilizing a batch size of 16 with a learning rate of $4e^{-4}$ for M-Baseline, and a batch size of 8 with a learning rate of $3e^{-4}$ for M-CONet.

4.2. Comparison with State-of-the-Art Methods

In this section, we compare our method with other approaches, as shown in Table 1. C, L, D, M denotes camera, LiDAR, depth and multi-modal respectively. If "Surround" is checked, the method predicts surrounding occupancy directly. Otherwise, it predicts occupancy for each view separately. As can be seen, our method outperforms others significantly. Compared with both M-Baseline and M-CONet, our method shows an impressive improvement of 1.8, which is considerable given the already high baseline.

Figure 4 gives a qualitative comparison. M-Baseline produces coarse prediction, while the output resolution of ours and M-CONet is the same as that of ground truth. Our method beats M-CONet in both completion and segmentation.

We boosts model performance in the most crucial categories in autonomous driving, such as car, motorcycle and pedestrian. We achieve a noticeable improvement of 5.7 in both the motorcycle and pedestrian categories, which are challenging but crucial to outdoor driving scenes. Our method is much better than CONet in vegetation, albeit falling short of achieving the best results. It is probably due to the sparse LiDAR points from irregular vegetation surfaces, leading to information loss in the first place. This problem needs to be solved in the future. In a nutshell, the overall significant improvement fully demonstrates the effectiveness of our method.

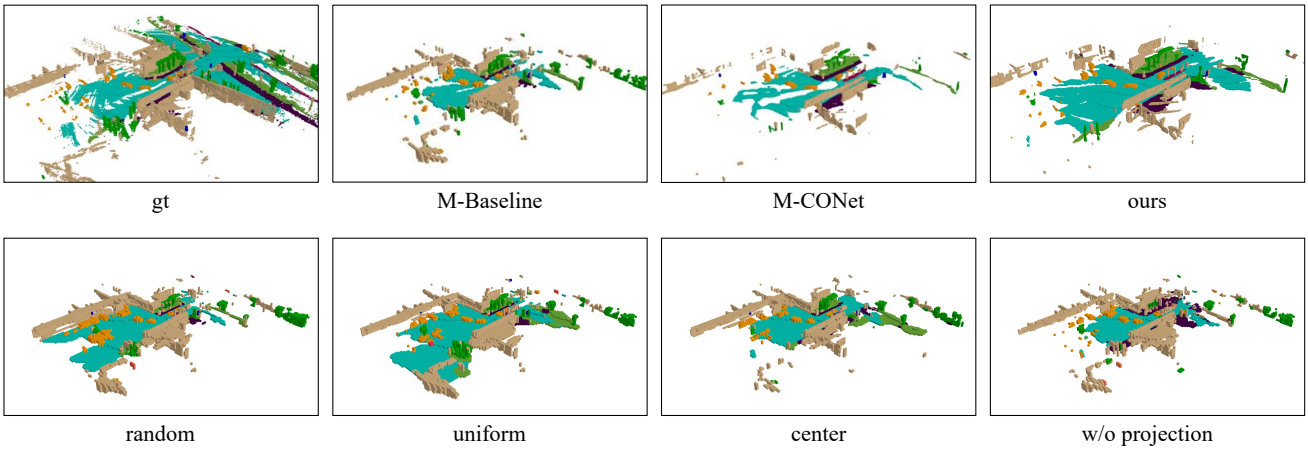

Figure 4. The first row shows a comparative analysis of our method against others. From left to right, the sequence includes visualization of ground truth, M-Baseline, M-CONet, and our method with PMAFusion. Our method gains superior performance in both completion and segmentation. For the second row, the first three images come from the sampling experiments detailed in Section 4.3.1 and the last one comes from the ablation study of the projection module in Section 4.3.2.

4.3. Ablation Study

We examine different sampling methods in Section 4.3.1 and compare between fusion with and without projection in Section 4.3.2. It is important to note that all experiments discussed in this section are conducted with M-Baseline.

4.3.1 Comparison of Different Sampling Technique

Table 2. Comparison of different sampling technique

Sample Method	mIoU (%)
Center Sample	16.6
Random Sample	16.3
Uniform Sample	16.9

Metric outcomes of various sampling methods are presented in Table 2. 'Center' refers to center point sampling, 'Random' indicates random sampling, and 'Uniform' represents uniform sampling. The result indicates that uniform sampling delivers the best performance, leading to an improvement of 0.6 points over random sampling, which is the least effective. The second row in Figure 4 shows a qualitative comparison. The outcomes of uniform sampling stand out, delivering the best performance in completing drivable surfaces and in the detailed representation of cars.

This superiority of uniform sampling can be attributed to its ability to more effectively represent the inherent spatial information of voxels, a feature crucial for accurate alignment.

Table 3. Effectiveness of aligned fusion

Method	GPU Mem.	mIoU (%)
w/o projection	35GB	11.8
w/ projection	17GB	16.9

Table 4. Efficiency Analysis on Aligned Fusion

Method	GPU Mem.	GFLOPs	mIoU (%)
C-Baseline[21]	17 GB	2241	10.3
C-CONet[21]	35 GB	6677	12.2
L-Baseline[21]	7.5 GB	749	11.7
L-CONet[21]	8.5 GB	810	15.8
M-Baseline[21]	19 GB	3050	15.1
Ours	17 GB	1757	16.9
M-CONet[21]	24 GB	3066	20.1
Ours	19 GB	1780	21.9

4.3.2 Efficiency and Effectiveness of Aligned Fusion

A key contribution of this paper is an accurate multi-modal alignment strategy. We conduct ablation experiment on M-Baseline to show the effectiveness of projection-based alignment. As shown in Table 3, the method labeled 'without projection' means employing an entirely learning-based alignment, i.e. let each voxel learn weights for all image pixels instead of those projected from reference points. It is clear from the result that a restricted local cross attention leads to a remarkable improvement in accuracy, boosting performance by 5.1 points. Furthermore, in comparison to global cross attention, the adoption of local cross atten-

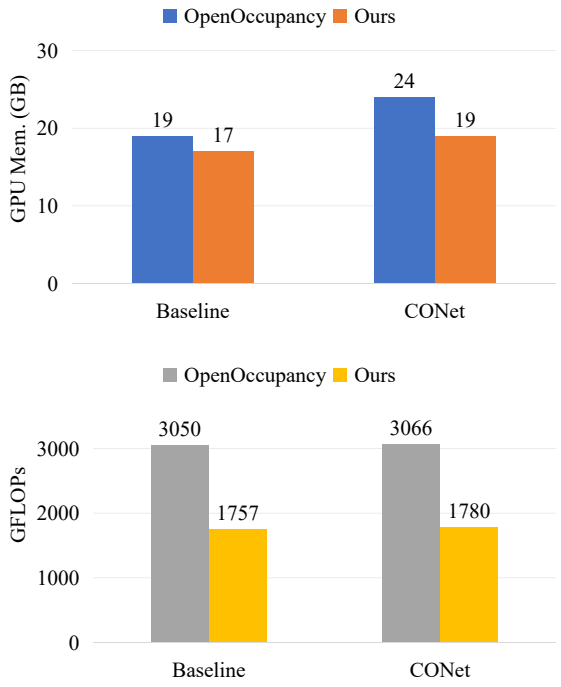

Figure 5. Two figures show the strengths of our proposed fusion module concerning GPU memory utilization and GFLOPs. PMA-Fusion, while elevating mIoU, also excels in computational efficiency.

tion results in a reduction of GPU memory consumption by ∼50%. This reduction not only boosts the model efficiency but also significantly aids in the convergence of the network. The natural match between voxel and its reference points stands as a strong prior for voxel-pixel alignment. This substantial increase demonstrates the effectiveness of our alignment-based fusion strategy. Figure 4 offers a more direct visual comparison.

In [21], the fusion module leverages monocular depth estimation results to construct a 3D image feature volume by back projecting from multi-view feature maps. This representation is then fused with a voxel feature volume of identical size generated from point clouds. Given the ill-posed nature of monocular depth estimation, this approach constitutes a relatively coarse alignment. Table 4 demonstrates the efficiency gains achieved by our proposed fusion method on both M-Baseline and M-CONet. By relying on projection-based alignment, it lowers GPU memory demands, allowing the high-resolution improved M-CONet to utilize memory comparable to that of the original M-Baseline. Additionally, it decreases GFLOPs by ∼40%, while gaining an enhancement in mIoU. A more intuitive comparison is presented in Figure 5. In comparison to the fusion techniques within OpenOccupancy, utilizing projection alignment not only increases the precision of alignment but also eliminates the requirement for direct construction of 3D image volumes, constituting a more resource-efficient and efficacious alternative.

5. Conclusion

In order to model the environment around the driving car, we design an occupancy prediction network with a fusion module that accurately fuses point cloud and image features. It does not need monocular depth estimation and relies solely on the projection of 3D points onto 2D images, thereby ensuring high accuracy and reliability. Unlike previous approach, our method avoids the explicit construction of 3D image feature volumes, relying instead on a singe cross-attention mechanism for the fusion process. This leads to reduced memory consumption and computational demands. Meanwhile, accurate alignment enables our method to achieve the best accuracy. In the field of multi-modal occupancy prediction, our projection-based voxel-pixel aligned fusion can be extensively used as a plugin module. How to select the reference points that best represent each voxel is of great significance for future research.

6. Acknowledgement

This work was partly supported by the Special Foundations for the Development of Strategic Emerging Industries of Shenzhen (Nos.JSGG20211108092812020 and CJGJZD20210408092804011).

References

[1] Jens Behley, Martin Garbade, Andres Milioto, Jan Quenzel, Sven Behnke, Cyrill Stachniss, and Jurgen Gall. Semantickitti: A dataset for semantic scene understanding of lidar sequences. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9297–9307, 2019. 1

[2] Yingjie Cai, Xuesong Chen, Chao Zhang, Kwan-Yee Lin, Xiaogang Wang, and Hongsheng Li. Semantic scene completion via integrating instances and scene in-the-loop. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 324–333, 2021. 3

[3] Anh-Quan Cao and Raoul de Charette. Monoscene: Monocular 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3991–4001, 2022. 1, 2, 5

[4] Xiaokang Chen, Kwan-Yee Lin, Chen Qian, Gang Zeng, and Hongsheng Li. 3d sketch-aware semantic scene completion via semi-supervised structure prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 4193–4202, 2020. 5

[5] Bowen Cheng, Ishan Misra, Alexander G Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. In *Proceed-*

ings of the IEEE/CVF conference on computer vision and pattern recognition, pages 1290–1299, 2022. 2

[6] Angela Dai, Angel X Chang, Manolis Savva, Macaiej Halber, Thomas Funkhouser, and Matthias Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5828–5839, 2017. 1

[7] Aloisio Dourado, Teofilo E De Campos, Hansung Kim, and Adrian Hilton. Edgenet: Semantic scene completion from a single rgb-d image. In *2020 25th international conference on pattern recognition (ICPR)*, pages 503–510. IEEE, 2021. 2

[8] Wanshui Gan, Ningkai Mo, Hongbin Xu, and Naoto Yokoya. A simple attempt for 3d occupancy estimation in autonomous driving. *arXiv preprint arXiv:2303.10076*, 2023. 2

[9] Martin Garbade, Yueh-Tung Chen, Johann Sawatzky, and Juergen Gall. Two stream 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019. 2

[10] Yuanhui Huang, Wenzhao Zheng, Yunpeng Zhang, Jie Zhou, and Jiwen Lu. Tri-perspective view for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9223–9232, 2023. 1, 2, 5

[11] Jie Li, Yu Liu, Dong Gong, Qinfeng Shi, Xia Yuan, Chunxia Zhao, and Ian Reid. Rgbd based dimensional decomposition residual network for 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7693–7702, 2019. 2

[12] Jie Li, Kai Han, Peng Wang, Yu Liu, and Xia Yuan. Anisotropic convolutional networks for 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3351–3359, 2020. 2, 3, 5

[13] Yiming Li, Zhiding Yu, Christopher Choy, Chaowei Xiao, Jose M Alvarez, Sanja Fidler, Chen Feng, and Anima Anandkumar. Voxformer: Sparse voxel transformer for camera-based 3d semantic scene completion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9087–9098, 2023. 1, 2

[14] Yu Liu, Jie Li, Qingsen Yan, Xia Yuan, Chunxia Zhao, Ian Reid, and Cesar Cadena. 3d gated recurrent fusion for semantic scene completion. *arXiv preprint arXiv:2002.07269*, 2020. 2

[15] Christoph B Rist, David Emmerichs, Markus Enzweiler, and Dariu M Gavrila. Semantic scene completion using local deep implicit functions on lidar data. *IEEE transactions on pattern analysis and machine intelligence*, 44(10):7205–7218, 2021. 1, 2

[16] Luis Roldao, Raoul de Charette, and Anne Verroust-Blondet. Lmscnet: Lightweight multiscale 3d semantic completion. In *2020 International Conference on 3D Vision (3DV)*, pages 111–119. IEEE, 2020. 1, 2, 5

[17] Nathan Silberman, Derek Hoiem, Pushmeet Kohli, and Rob Fergus. Indoor segmentation and support inference from rgbd images. In *Computer Vision–ECCV 2012: 12th European Conference on Computer Vision, Florence, Italy, October 7-13, 2012, Proceedings, Part V 12*, pages 746–760. Springer, 2012. 1

[18] Shuran Song, Fisher Yu, Andy Zeng, Angel X Chang, Manolis Savva, and Thomas Funkhouser. Semantic scene completion from a single depth image. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1746–1754, 2017. 1

[19] Xiaoyu Tian, Tao Jiang, Longfei Yun, Yue Wang, Yilun Wang, and Hang Zhao. Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving. *arXiv preprint arXiv:2304.14365*, 2023. 1

[20] Wenwen Tong, Chonghao Sima, Tai Wang, Li Chen, Silei Wu, Hanming Deng, Yi Gu, Lewei Lu, Ping Luo, Dahua Lin, et al. Scene as occupancy. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8406–8415, 2023. 2

[21] Xiaofeng Wang, Zheng Zhu, Wenbo Xu, Yunpeng Zhang, Yi Wei, Xu Chi, Yun Ye, Dalong Du, Jiwen Lu, and Xingang Wang. Openoccupancy: A large scale benchmark for surrounding semantic occupancy perception. *arXiv preprint arXiv:2303.03991*, 2023. 1, 2, 3, 4, 5, 6, 7

[22] Yida Wang, David Joseph Tan, Nassir Navab, and Federico Tombari. Forknet: Multi-branch volumetric semantic completion from a single depth image. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 8608–8617, 2019. 2

[23] Xu Yan, Jiantao Gao, Jie Li, Ruimao Zhang, Zhen Li, Rui Huang, and Shuguang Cui. Sparse single sweep lidar point cloud segmentation via learning contextual shape priors from scene completion. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 3101–3109, 2021. 1, 2, 5

[24] Chubin Zhang, Juncheng Yan, Yi Wei, Jiaxin Li, Li Liu, Yansong Tang, Yueqi Duan, and Jiwen Lu. Occnerf: Self-supervised multi-camera occupancy prediction with neural radiance fields. *arXiv preprint arXiv:2312.09243*, 2023. 2

[25] Pingping Zhang, Wei Liu, Yinjie Lei, Huchuan Lu, and Xiaoyun Yang. Cascaded context pyramid for full-resolution 3d semantic scene completion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 7801–7810, 2019. 2, 3

[26] Yunpeng Zhang, Zheng Zhu, and Dalong Du. Occformer: Dual-path transformer for vision-based 3d semantic occupancy prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9433–9443, 2023. 2

[27] Min Zhong and Gang Zeng. Semantic point completion network for 3d semantic scene completion. In *ECAI 2020*, pages 2824–2831. IOS Press, 2020. 2