Supplementary Material for Fully Sparse 3D Occupancy Prediction

Haisong Liu[1,2], Yang Chen[1], Haiguang Wang[1], Zetong Yang[2], Tianyu Li[2], Jia Zeng[2], Li Chen[2], Hongyang Li[2], and Limin Wang[1,2]

[1] State Key Laboratory for Novel Software Technology, Nanjing University
[2] Shanghai AI Lab

A Sparsification Methods

Table 1: Top-k vs.thresholding for sparsification.

Sparsification Method	Value	RayIoU	Average FPS
Top-k	32000	34.0	17.3
Thresholding	0.6	34.1	16.6
Thresholding	0.7	**34.2**	17.3
Thresholding	0.8	33.9	**18.3**

In the main paper, we use topk-k to prune the empty voxels. However, such k is specific to specific dataset and does not generalize to scenes with different complexities. In this section, we substitute top-k with a thresholding method, e.g. voxels scoring less than a certain threshold (e.g. 0.5) will be pruned. Thresholding achieves similar performance to top-k (see Tab. 1), and has the ability to generalize to different scenes.

B The Effect of Training with Visible Masks

Interestingly, we observed a peculiar phenomenon. Under the traditional voxel-level mIoU metric, methods can significantly benefit from disregarding the non-visible voxels during training. These non-visible voxels are indicated by a binary visible mask provided by the Occ3D-nuScenes dataset. However, we find that this strategy actually impairs performance under our new RayIoU metric. For instance, we train two variants of BEVFormer: one uses the visible mask during training, and the other does not. As shown in Tab. 4, the former scores 15 points higher than the latter on the voxel-based mIoU, but it scores 1 point lower on RayIoU. This phenomenon is also observed on FB-Occ.

To explore this phenomenon, we present the per-class RayIoU in Tab. 4. The table reveals that using the visible mask during training enhances performance

Table 4: To verify the effect of the visible mask, wo provide per-class RayIoU of BEVFormer and FB-Occ on the validation split of Occ3D-nuScenes. † uses the visible mask during training. We find that training with visible mask hurts the performance of ground classes such as drivable surface, terrian and sidewalk.

Method	RayIoU	others	barrier	bicycle	bus	car	cons. veh.	motor.	pedes.	tfc. cone	trailer	truck	drv. surf.	other flat	sidewalk	terrain	manmade	vegetation
BEVFormer	**33.7**	5.0	42.2	18.2	**55.2**	**57.1**	**22.7**	21.3	31.0	**27.1**	**30.7**	49.4	**58.4**	**30.4**	**29.4**	**31.7**	36.3	26.5
BEVFormer †	32.4	**6.4**	**44.8**	**24.0**	**55.2**	56.7	21.0	**29.8**	**33.5**	26.8	27.9	**49.5**	45.8	18.7	22.4	18.5	**39.1**	**29.8**
FB-Occ	**35.6**	**10.5**	44.8	25.6	55.6	51.7	22.6	27.2	**34.3**	**30.3**	23.7	44.1	**65.5**	**33.3**	**31.4**	**32.5**	**39.6**	**33.3**
FB-Occ †	33.5	5.0	**44.9**	**26.2**	**59.7**	**55.1**	**27.9**	**29.1**	**34.3**	29.6	**29.1**	**50.5**	44.4	22.4	21.5	19.5	39.3	31.1

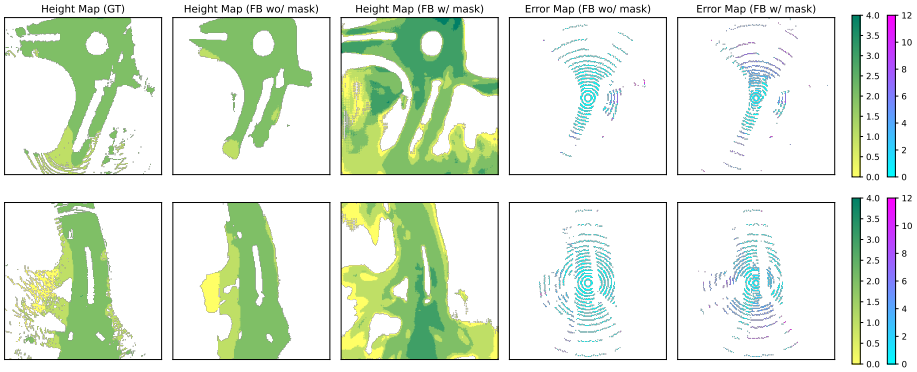

Fig. 10: Why does the performance of background classes, such as drivable surfaces, deteriorate when using the visible mask during training? We provide a visualization of the drivable surface as predicted by FB-Occ. Here, "FB w/ mask" and "FB wo/ mask" denote training with and without the visible mask, respectively. We observe that "FB w/ mask" tends to predict a higher and thicker road surface, resulting in significant depth errors along a ray. In contrast, "FB wo/ mask" predicts a road surface that is both accurate and consistent.

for most foreground classes such as bus, bicycle, and truck. However, it negatively impacts background classes like drivable surface, terrain, and sidewalk.

This observation prompts a further question: why does performance deteriorate for background classes? To address this, we offer a visual comparison of depth errors and height maps of the predicted drivable surface from FB-Occ, both with and without the use of visible mask during training, in Fig. 10. The figure illustrates that training with visible mask results in a thicker and higher ground representation, leading to substantial depth errors in distant areas. Conversely, models trained without the visible mask predict depth with greater accuracy.

From these observations, we derive some valuable insights: ignoring non-visible voxels during training benefits foreground classes by resolving the issue of ambiguous labeling of unscanned voxels. However, it also compromises the accuracy of depth estimation, as models tend to predict a thicker and closer surface. We hope that our findings will benefit future research.

C Panoptic Occupancy Prediction

Thanks to the mask transformer, our SparseOcc can produce panoptic occupancy prediction by simply replacing the semantic queries with instance queries.

Ground Truth Preparation. To evaluate our method, we utilize the ground-truth object bounding boxes from the 3D detection task to generate the panoptic occupancy ground truth. First, we define eight instance categories (including `car`, `truck`, `construction vehicle`, `bus`, `trailer`, `motorcycle`, `bicycle`, `pedestrian`) and ten staff categories (including `terrain`, `manmade`, `vegetation`, etc). Next, we identify each instance segment by grouping the voxels inside the box based on an existing semantic occupancy benchmark Occ3D-nuScenes.

However, we observe that using the original size of the box for grouping may cause some boundary voxels being missed due to the compactness of the box. Enlarging the box (such as 1.2x) leads to excessive overlap between boxes. To address these issues, we designed a two-stage grouping scheme. In the first stage, we use the original size of the box for grouping. Then, for voxels that have not been assigned to a specific instance, we select the closest box and assign it. This scheme effectively resolves the problems of boundary omission and box overlap.

Evaluation Metrics. We design RayPQ based on the well-known panoptic quality (PQ) metric, which is defined as the multiplication of *segmentation quality* (SQ) and *recognition quality* (RQ):

$$\mathrm{PQ} = \underbrace{\frac{\sum_{(p,g)\in TP} \mathrm{IoU}(p,g)}{|TP|}}_{\text{segmentation quality (SQ)}} \times \underbrace{\frac{|TP|}{|TP| + \frac{1}{2}|FP| + \frac{1}{2}|FN|}}_{\text{recognition quality (RQ)}}, \tag{1}$$

where the definition of true positive (TP) is the same as that in RayIoU. The threshold of IoU between prediction p and ground-truth g is set to 0.5.

Table 5: Panoptic occupancy prediction performance on Occ3D-nuScenes.

Method	Backbone	Input Size	Epochs	RayPQ	RayPQ_{1m}	RayPQ_{2m}	RayPQ_{4m}
SparseOcc	R50	704 × 256	24	**14.1**	10.2	14.5	17.6

Table 6: Experiments on enhancing sparsity by removing certain background categories (denoted by †). The RayIoU* metrics is only calculated on categories that are not ignored. By enhancing sparsity, the inference speed of SparseOcc can be further improved with negligible performance loss.

Method	Backbone	Input Size	Epochs	Top-k	RayIoU*	FPS
SparseOcc	R50	704 × 256	24	32000	**30.1**	24.0
SparseOcc	R50	704 × 256	24	24000	29.8	24.8
SparseOcc	R50	704 × 256	24	16000	28.8	**26.0**
SparseOcc †	R50	704 × 256	24	32000	**30.1**	24.0
SparseOcc †	R50	704 × 256	24	24000	30.0	24.8
SparseOcc †	R50	704 × 256	24	16000	29.4	**26.0**

Results. In Tab. 5, we report the performance of SparseOcc on panoptic occupancy benchmark. Similar to RayIoU, we calculate RayPQ under three distance thresholds: 1, 2 and 4 meters. SparseOcc achieves an averaged RayPQ of 14.1. The visualizations are presented in the main paper (Fig. 9).

D Enhancing Sparsity

As mentioned in the main paper, the majority of non-free occupancy data pertains to the background geometry, such as the road surface. In practice, the occupancy of road surface can be effectively substituted with High-Definition Map (HD Map) or online mapping techniques. Thus, the sparsity of the scene can be further enhanced by removing certain background categories, leading to faster inference speed with negligible performance loss. This is also an advantage of SparseOcc compared to the dense counterparts, because the dense methods will not speed up as the sparsity of the scene increases.

Settings. We train a variant of the model that the voxels belonging to the `drivable surface` and `terrian` in the ground truth are treated as free during training (denoted by † in Tab. 6). For fair evaluation, all models are evaluated on the categories that are not ignored.

Results. As shown in Tab. 6, the performance of baseline (modeling all categories) drops notably as the top-k decreases. This is reasonable as the number of voxels is not enough to express the entire scene. In contrast, if we ignore certain background categories, the performance loss is negligible (only 0.7 RayIoU) even

when top-k is reduced by half. This means the inference speed of SparseOcc can be further improved by enhancing sparsity, while for the dense counterparts it is not possible.

E Visualization of 3D Reconstruction

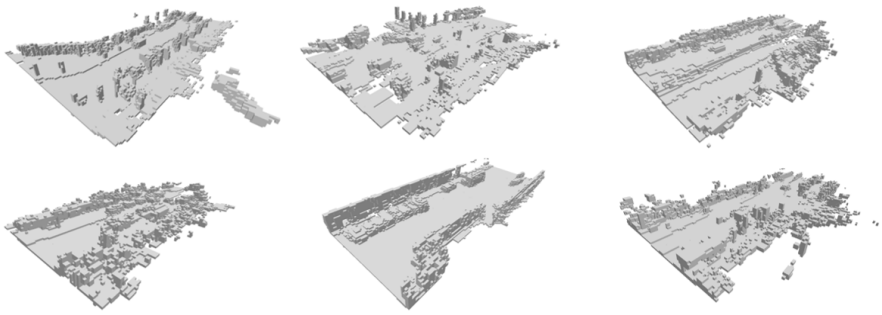

Fig. 11: Visualization of 3D reconstruction results from sparse voxel decoder.

In Fig. 11, we visualize the reconstructed 3D geometry from sparse voxel decoder. SparseOcc can reconstruct fine-grained details from camera-only inputs.