Deformable Gaussian Occupancy: Decoupling Rigid and Nonrigid Motion with Factorized Distillation

Supplementary Material

1. Effect of Deformation on Gaussian Sparsity

Table 1 shows the robustness of our model under reduced Gaussian density by progressively decreasing the number of Gaussian primitives. Through it, we can find the performance degrades more when removing the deformation module, indicating a strong reliance on dense Gaussian representations. In contrast, adding the deformation module yields smaller performance drops across all settings, demonstrating its benefit in improving robustness to sparse Gaussians. This suggests that our method can maintain accurate scene modeling even with fewer primitives, enabling more efficient inference.

	10000	5000	1000	500
w/o Deformation Module	12.26	10.17 (-17%)	8.85 (-28%)	6.02 (-51%)
w/ Deformation Module	**18.05**	**17.10** (-5%)	**14.90** (-17%)	**13.09** (-27%)

Table 1. **Performance under Reduced Gaussian Density.**

2. Efficiency

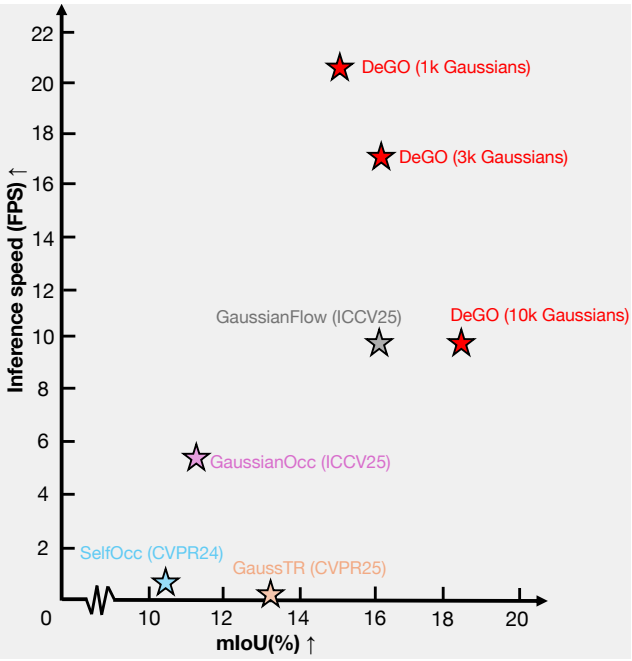

Figure 1. **Comparison of inference speed and accuracy with state-of-the-art methods on Occ3D-NuScenes validation set.**

To assess model efficiency, we follow the evaluation protocol of GaussianFlow [1] and measure inference speed on the full Occ3D-NuScenes validation set [2, 7] using a single A100 GPU. As shown in Figure 1, our model achieves over 20 FPS with only 1k Gaussians while maintaining competitive accuracy. With 3k Gaussians, it matches the mIoU of the best prior method while providing over 70% faster inference. These results demonstrate that our method enables efficient inference with significantly fewer Gaussians.

Method	RayIoU	RayIoU@1	RayIoU@2	RayIoU@4
GaussianOcc [3]	13.43	9.85	13.49	16.94
GaussianFlow [1]	18.00	12.24	18.13	23.69
DeGO (ours)	**18.89**	**13.37**	**18.93**	**24.37**

Table 2. **Performance under ray-based metrics.** Similar to the main text, we exclude the 'others' and 'other flat' classes and compute the RayIoU on the other 15 classes.

3. Performance on Ray-based Metric

In addition to conventional metrics such as IoU and mIoU for occupancy prediction, we also report results using the ray-based metric RayIoU introduced in [5]. Unlike standard voxel-level IoU, the ray-based metric computes the agreement between predicted and ground-truth voxels only along each camera ray. This formulation focuses on view-consistent occupancy and avoids penalizing voxels that are never observed by cameras.

As shown in Table 2, we follow the evaluation protocol of [1] and compare RayIoU at multiple depth intervals. Our method consistently surpasses previous state-of-the-art baselines across all depth units, demonstrating the strong view consistency of our predicted occupancy.

4. More Visualizations

To further examine performance on deformable, human-centric classes, we provide additional qualitative comparisons of the baseline, our method, and the ground-truth occupancy predictions. As shown in Figure 2, the baseline often misclassifies bicycles and motorcycles as pedestrians. Although these categories can appear visually similar, our method more reliably distinguishes them, producing noticeably more accurate predictions.

5. Implementation Details

We use a ResNet-50 [4] image encoder and a Gaussian Transformer consisting of three blocks with a hidden di-

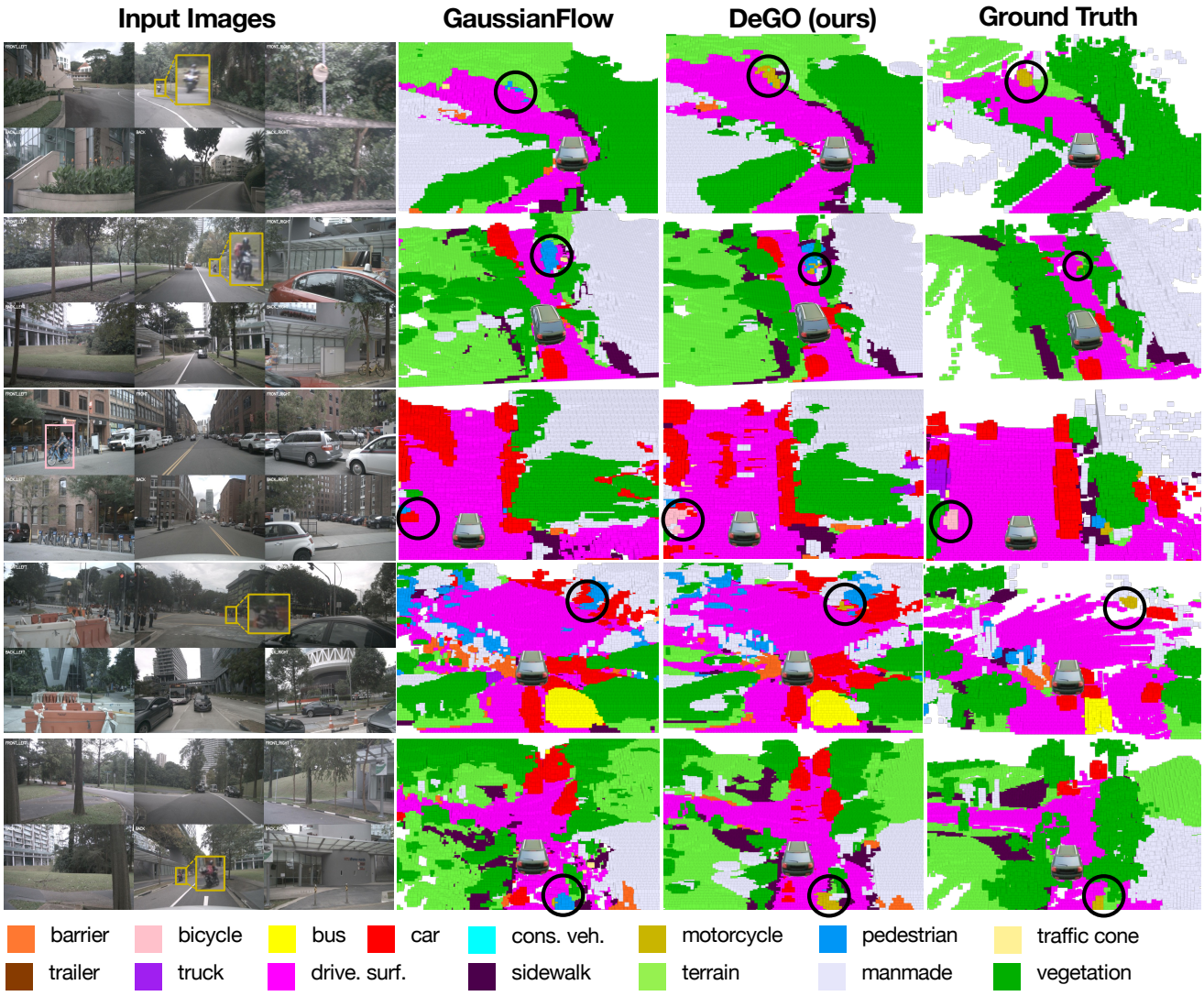

Figure 2. **Additional qualitative comparison with the state of the art.** We highlight two additional human-centric classes, bicycle and motorcycle. Compared with GaussianFlow, our method produces more accurate predictions for these visually similar categories.

mension of 256. The deformation module is configured with 32 temporal channels, a positional encoding level of 6, and a time-encoding level of 4. The feature network is a 6-layer MLP, and the 4D Gaussian prediction heads are implemented as 2-layer MLPs. For VGGT distillation, we use the feature layer at index 22 and project it to a 32-dimensional space. The model is trained with a batch size of four on four A100 GPUs using the Adam optimizer [6], starting with a learning rate of 1e-4 and a decay rate of 1e-2. To ensure stable optimization, we adopt a cosine learning-rate schedule with warm-up and standard gradient clipping. Additional architectural, data processing, and training hyperparameters are summarized in Table 3 for completeness.

References

[1] Simon Boeder, Fabian Gigengack, and Benjamin Risse. Gaussianflowocc: Sparse and weakly supervised occupancy estimation using gaussian splatting and temporal flow. *arXiv preprint arXiv:2502.17288*, 2025. 1

[2] Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 11621–11631, 2020. 1

[3] Wanshui Gan, Fang Liu, Hongbin Xu, Ningkai Mo, and Naoto Yokoya. Gaussianocc: Fully self-supervised and efficient 3d occupancy estimation with gaussian splatting. In *Proceedings of the IEEE/CVF International Conference on Computer Vi-*

Parameter	Value	Description
Data & input:		
Number of cameras	6	Surround-view camera inputs
Input image size	256×704	Network input resolution
Source image size	900×1600	Original image resolution
Voxel grid range (x, y, z)	$[-40, 40] \times [-40, 40] \times [-1, 5.4]$ m	3D occupancy volume
Voxel size (x, y, z)	$0.4, 0.4, 0.4$ m	Spatial resolution of BEV voxels
Model:		
Backbone	ResNet-50	Image feature extractor
Neck	FPN	Multi-scale feature aggregation
Hidden dimension	256	BEV and Gaussian feature channels
Gaussian init scale	4	Initial Gaussian scale factor
Temporal frame IDs	$[-8, \ldots, 8]$	Context frames for deformation
Training clip length	8	Number of adjacent frames per clip
Deformation depth	6	Layers in deformation MLP
Time embedding dim	32	Temporal embedding dimension
Positional enc. levels	6	Levels for 3D position encoding
Time enc. levels	4	Levels for time encoding
VGGT teacher layer	22	VGGT layer index for distillation
Gaussian feature dim	32	Per-Gaussian distilled feature size
Teacher feature dim	2048	VGGT feature dimension
Distillation loss	cosine	Normalized cosine feature loss
Training:		
Optimizer	AdamW	Optimization algorithm
Learning rate	1×10^{-4}	Initial learning rate
Weight decay	1×10^{-2}	Weight decay factor
LR schedule	cosine w/ warm-up	Custom cosine annealing schedule
Warm-up iterations	200	Linear warm-up steps
Warm-up ratio	0.001	Warm-up start LR ratio
Max epochs	30	Total training epochs
Batch size	4	Samples per GPU
Workers per GPU	12	Data loading workers
Gradient clipping	5.0	Max gradient norm

Table 3. **Detailed hyperparameters used in our implementation.**

sion, pages 28980–28990, 2025. 1

[4] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[5] Haisong Liu, Yang Chen, Haiguang Wang, Zetong Yang, Tianyu Li, Jia Zeng, Li Chen, Hongyang Li, and Limin Wang. Fully sparse 3d occupancy prediction. In *European Conference on Computer Vision*, pages 54–71. Springer, 2024. 1

[6] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 2

[7] Xiaoyu Tian, Tao Jiang, Longfei Yun, Yucheng Mao, Huitong Yang, Yue Wang, Yilun Wang, and Hang Zhao. Occ3d: A large-scale 3d occupancy prediction benchmark for autonomous driving. *Advances in Neural Information Processing Systems*, 36:64318–64330, 2023. 1