AD-GS: Object-Aware B-Spline Gaussian Splatting for Self-Supervised Autonomous Driving

Supplementary Material

A. Implementation Details

Details in Equation 5. According to the prior work [25], the explicit expression of M_k is

$$M_k = \frac{1}{k-1}\left(\begin{bmatrix} M_{k-1} \\ 0 \end{bmatrix}\begin{bmatrix} 1 & k-2 & & & 0 \\ & 2 & k-3 & & \\ & & \ddots & \ddots & \\ 0 & & & k-1 & 0 \end{bmatrix} + \begin{bmatrix} 0 \\ M_{k-1} \end{bmatrix}\begin{bmatrix} -1 & 1 & & & 0 \\ & -1 & 1 & & \\ & & \ddots & \ddots & \\ 0 & & & -1 & 1 \end{bmatrix}\right),$$

$$M_1 = [1]. \tag{17}$$

The above expression shows that M_k can be precomputed before training to improve efficiency.

Simplified Pseudo 2D Segmentation. We use Grounded-SAM-2 [12, 17, 26] with the Grounding DINO [17] base model to generate the segmentation results as the object mask. For the object segmentation in the KITTI [7] and Waymo [29] datasets, we use the prompt "car.bus.truck.van.human". Additionally, for the nuScenes [1] dataset, we include "bike" as an extra prompt. To generate pseudo labels for the sky mask, we use the prompt "sky".

Attribute Inheritance in Densification. In AD-GS, the object Gaussians can only generate object Gaussians through the splitting or cloning operations, which are the same as the background Gaussians. The newly created Gaussians will inherit all parameters from their parents, including the fixed parameter μ_t in Equation 10. With this design, the loss $\mathcal{L}_{obj}$ in Equation 9 optimizes the opacity of each Gaussian, allowing object or background Gaussians with low opacities in incorrect locations to be pruned, thereby refining their numbers and positions.

Others. Following StreetGS [35], we incorporate Structure-from-Motion (SfM) [27] points as the initial background Gaussians to account for regions beyond the LiDAR scan range. To mitigate the impact of imprecise camera poses in the KITTI and nuScenes datasets, we use a unified deformation for the positions of all Gaussians $G \in \Omega_{obj} \bigcup \Omega_{bkg}$. Additionally, for the learnable spherical environment map, we set the resolution to 8192×8192.

Table 6. Ablation of the B-spline control points on the KITTI [7] dataset by changing the ratio between the number of control points and total frames. The color of each cell shows the best and the second best.

Frames per Ctrl Pts	PSNR ↑	SSIM ↑	LPIPS ↓
1	27.26	0.902	0.046
2	28.69	0.910	0.038
3 (Ours)	29.16	0.920	0.033
4	29.11	0.920	0.033

Table 7. Ablation of the B-spline order on the KITTI [7] dataset. The color of each cell shows the best.

Order	PSNR ↑	SSIM ↑	LPIPS ↓
$k = 2$	29.10	0.919	0.033
$k = 6$ (Ours)	29.16	0.920	0.033
$k = 10$	29.12	0.920	0.033
$k = 6$ + quat sin&cos	28.99	0.919	0.034

B. Experimental Setup Details

Dataset. For the KITTI [7] dataset, we select 0001, 0002 and 0006 sequences for evaluation with the left and right cameras. For the Waymo [29] dataset, we select seg104481, seg123746, seg176124, seg190611, seg209468, seg424653, seg537228 and seg839851 with the FRONT camera, and use one out of every four frames for testing. For the nuScenes [1] dataset, we select 0230, 0242, 0255, 0295, 0518 and 0749 scenes from 10 to 69(inclusive) frames with the FRONT, FRONT_LEFT, and FRONT_RIGHT cameras.

Baselines. We mainly use the official implementation of EmerNeRF [36] for the experiments on the Waymo and nuScenes dataset. For the PVG [2], we apply the hyperparameters designed for the Waymo dataset to evaluate its performance on the nuScenes dataset. Notably, we use the same sky masks generated by SAM [12, 17, 26] for EmerNeRF, PVG and AD-GS (Ours). To adapt Grid4D [34] for auto-driving scenarios, we extend it by increasing the temporal grid resolution to $1024 \times 1024 \times 32$. We adapt 4DGF [5] for our experimental setting on the Waymo dataset by adding the SfM points and disable the camera optimization.

Others. We use the dynamic mask from StreetGS [35] to compute the PSNR* only for moving objects on the Waymo dataset in Table 2 and Table 5.

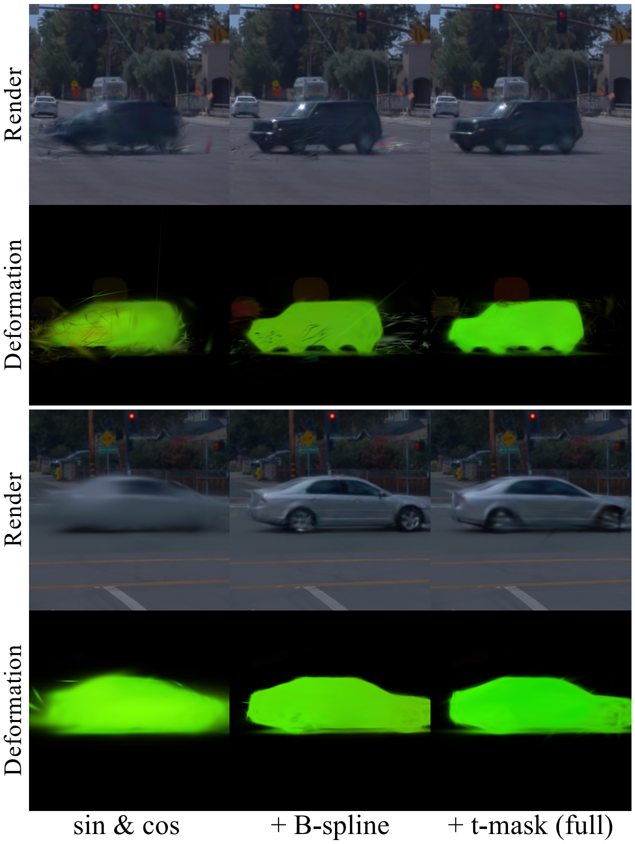

Figure 9. Deformation map of the object modeling module ablation study. In the deformation map, similar colors indicate similar deformations, and B-spline curves enhance the clarity and accuracy of the visualization. The bottom scenario depicts a car that is visible only briefly. In this case, optimization might be influenced by incorrect gradients from the training frames where the car is invisible. The results further demonstrate the effectiveness of B-spline curves in local fitting.

Table 8. Rendering speed comparison on the KITTI [7] dataset with self-supervised models. The color of each cell shows the best and the second best.

Model	Grid4D [34]	PVG [2]	AD-GS (Ours)
PSNR ↑	23.79	27.13	29.16
FPS ↑	40	58	47

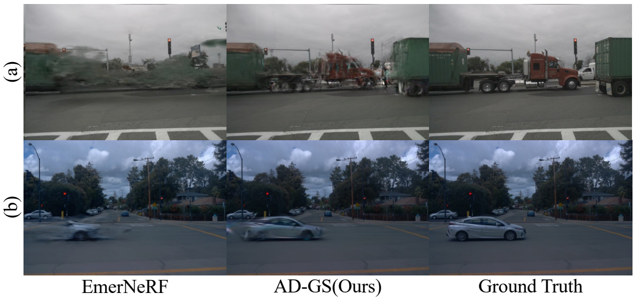

Figure 10. Failure cases when facing complex objects (a) and objects only visible in a quite short time (b).

Table 9. Quantitative comparison by only removing flow supervision L_f in our model. The color of each cell shows the best and the second best. * denotes the metric only for moving objects.

	Waymo				KITTI-75%		
Model	PSNR	SSIM	LPIPS	PSNR*	PSNR	SSIM	LPIPS
PVG	29.54	0.895	0.266	21.56	27.13	0.895	0.049
EmerNeRF	31.32	0.881	0.301	21.80	-	-	-
Ours w/oL_f	33.20	0.925	0.229	25.32	28.84	0.917	0.036
Ours	33.91	0.927	0.228	27.41	29.16	0.920	0.033

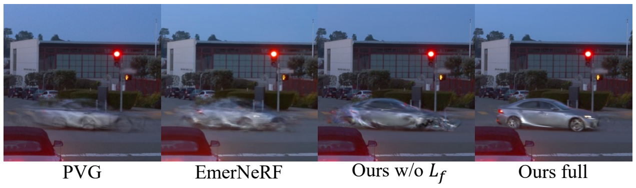

Figure 11. Qualitative comparison by removing flow supervision in our model.

C. Additional Results

Ablation of B-Splines. We conduct additional ablation studies on the parameters of B-spline curves using the “KITTI-75%” setting. The results are presented in Table 7 for the order k and Table 6 for the number of control points. When the order is too low or the control points are overly dense, the smoothness of the B-spline curves is constrained. In this case, each control point is optimized using fewer frames, leading to performance degradation under noisy self-supervision. Conversely, when the order is too high or the control points is insufficient, the local fitting capability decreases, resulting in artifacts. Based on these observations, we select an order of $k = 6$ and set the ratio between the number of control points and the total frames to $1/3$ in our experiments to balance the smoothness and the local fitting property. Additionally, we perform an ablation study on the deformation of the rotation parameter, with results shown in the last row of Table 7. The setting “sin&cos” refers to modeling Gaussian rotation deformations using trigonometric functions, and the results show that the trigonometric functions are not necessary for the rotation parameters.

Ablation of Optical Flow Supervision. A certain optical flow supervision is essential for this task, as it significantly aids in reconstructing the fast-moving objects commonly found in auto-driving scenarios. Pixels corresponding to such objects often exhibit significant displacements over time, making accurate matching challenging in the absence of flow supervision for trajectory reconstruction. We conducted an additional experiment in which only the flow supervision term L_f of our model was removed, and the results are shown in Table 9. Although our model still outperforms previous methods without optical flow, Figure 11 exhibits the obvious degradation caused by the absence of flow supervision, particularly in cases with fast-moving cars crossing the scene.

Analysis of Motion Fitting. To further demonstrate the effectiveness of trigonometric function and B-spline curve in global and local fitting, we visualize the deformation map mainly following the approach in Grid4D [34]. The results are shown in Figure 9, where similar colors indicate similar deformations. Although trigonometric functions can approximate the general deformation of an object under the noisy self-supervision, their representation tends to be inaccurate due to the omission of per-frame local details. In contrast, B-spline curves offer advantages in capturing local details, allowing for more precise fine-tuning of the representation. The bottom scenario in Figure 9 illustrates a special case where a car suddenly appears and then disappears, remaining visible for only about 17 frames (total about 160 frames). When the model has not been fully optimized, the car still appears in the invisible frames. However, the invisible frames cannot provide the correct information for the model to fit the trajectory at their timestamps. In such cases, trigonometric functions might be influenced by numerous invisible training frames, leading to incorrect gradients during optimization and resulting in severe blurring in motion representation and rendering. However, B-spline curves mitigate this issue by optimizing only the relevant control points, thereby reducing the impact of incorrect gradients from invisible training frames and significantly improving the accuracy of the representation. Therefore, by combining trigonometric functions and B-spline curves for motion fitting, we achieve more accurate motion representations.

Rendering Speed. We evaluate the rendering speed of AD-GS on the KITTI dataset with the “KITTI-75” setting. As shown in Table 8, AD-GS maintains fast rendering performance while improving quality, benefiting from the low computational overhead of trigonometric functions and B-spline curves.

Additional Visualization. Figure 13 shows the additional rendering results on the KITTI [7] dataset. Figure 12 is the additional rendering results on the Waymo [29] dataset. Figure 14 displays the additional rendering results in the nuScenes [1] dataset.

D. Limitations

Although AD-GS achieves state-of-the-art performances in self-supervised auto-driving scene rendering, it still has several limitations. In some cases, AD-GS fails to outperform certain state-of-the-art rendering models that leverage manual 3D annotations to avoid the challenges of motion and object reconstruction. Additionally, our model may produce artifacts if the quality of the pseudo-labels is quite low. As illustrated in Figure 10 (a), our model probably fails to reconstruct objects with highly complex motions and structures. Moreover, when an object is only visible for an extremely brief period, such as about 10 frames, AD-GS might obtain suboptimal rendering results, as shown in Figure 10 (b).

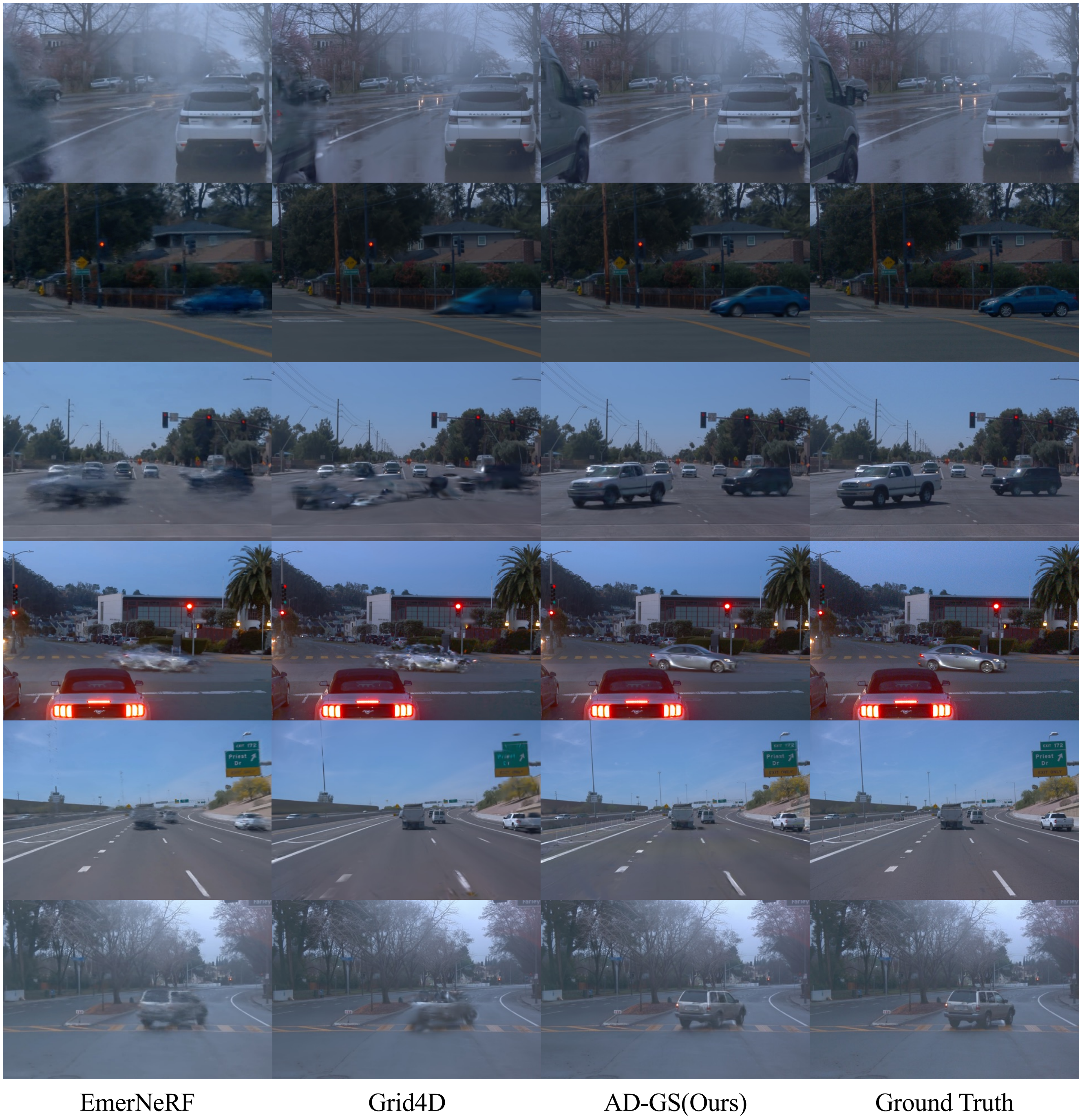

Figure 12. Additional qualitative comparisons on the Waymo [29] dataset.

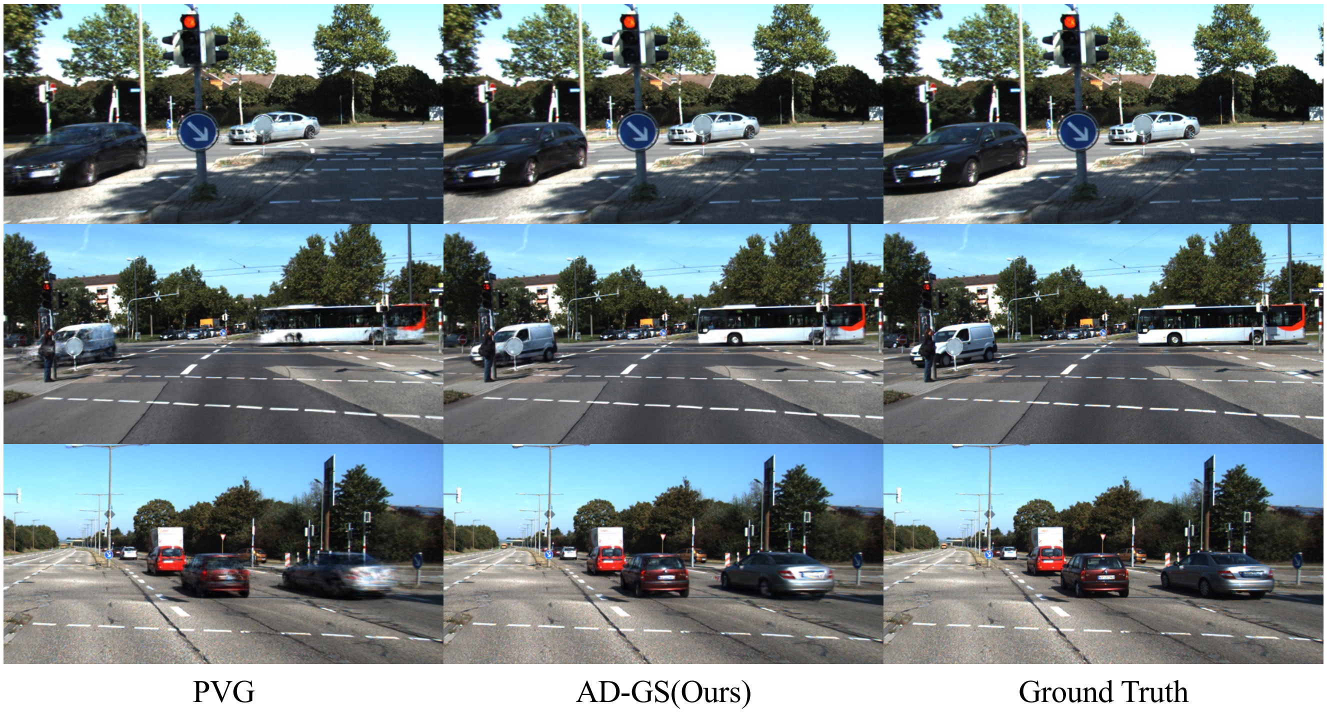

Figure 13. Additional qualitative comparisons on the KITTI [7] dataset.

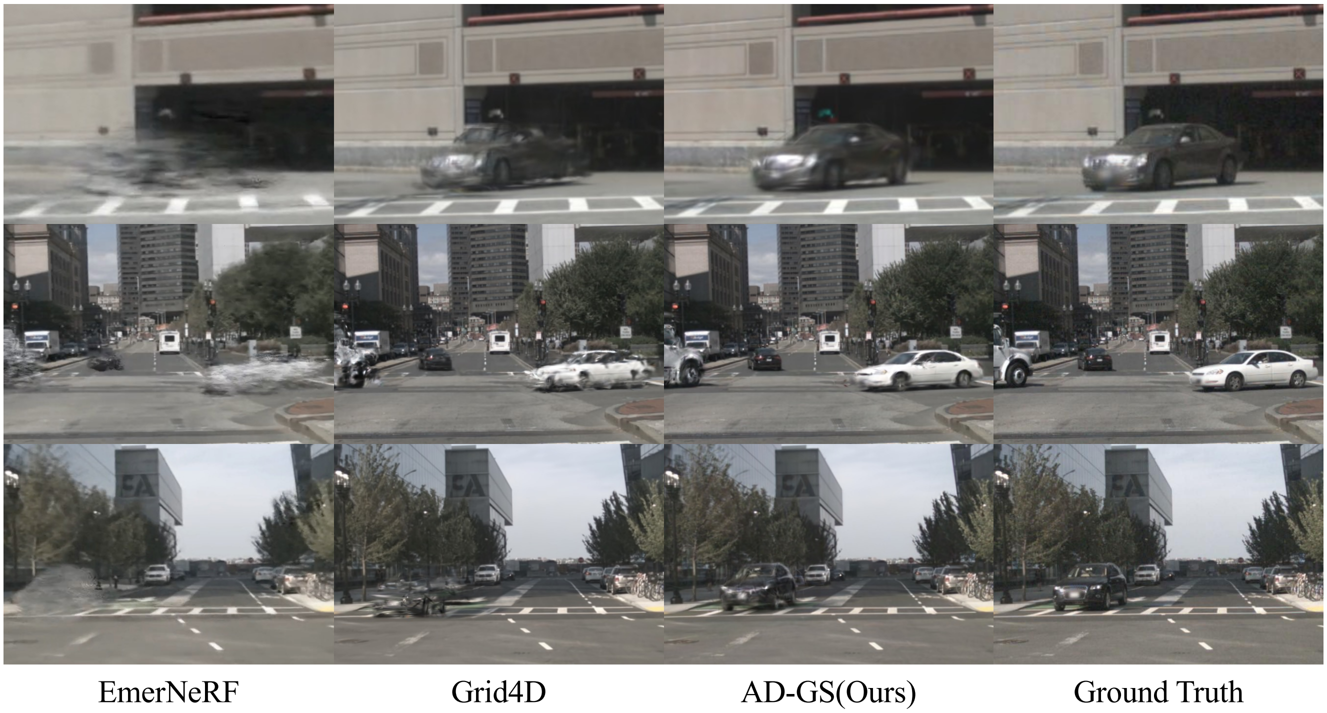

Figure 14. Additional qualitative comparisons on the nuScenes [1] dataset.