

Unifying Appearance Codes and Bilateral Grids for Driving Scene Gaussian Splatting

– Supplementary Material –

This supplementary document provides additional insights and technical details regarding our proposed multi-scale bilateral grid framework. We begin with an expanded discussion of implementation specifics, covering our guidance map generation, slice operation, multi-scale fusion strategy, comprehensive training details (including loss functions and optimization), and dynamic rendering techniques for real-world ISP adaptation in Section A1. To assess performance, we describe our evaluation protocol, detailing the diverse autonomous driving datasets utilized (Waymo, NuScenes, PandaSet, and Argoverse2) and the key metrics for geometry and appearance evaluation in Section A2. Next, we provide a comprehensive presentation of the detailed quantitative experimental results on these datasets, including per-scene and average metrics for both geometry and appearance, in Section A3. Finally, we showcase additional qualitative visualization results, including error maps and comparisons with baseline methods, to further illustrate the performance of our method across various challenging scenarios in Section A4.

Scene	Geometry Evaluation		
	CD↓	RMSE↓	Depth↓
152	1.227	3.463	0.042
164	0.889	2.469	0.103
171	1.042	2.813	0.075
200	1.300	3.617	0.020
209	1.294	3.524	0.065
359	1.238	3.542	0.036
529	0.678	2.675	0.021
916	1.623	4.617	0.109
Average	1.161	3.340	0.059

Table A1: Detailed Scene-by-Scene Geometry Evaluation on the NuScenes [1]. This table presents key geometry metrics—Chamfer Distance (CD), Root Mean Square Error (RMSE), and Depth error—for individual scenes and averaged across the evaluated sequences from the NuScenes dataset.

Scene	Geometry Evaluation		
	CD↓	RMSE↓	Depth↓
0	1.638	3.217	1.733
3	1.427	2.905	0.350
31	0.278	1.636	0.026
233	1.149	3.465	0.594
551	1.339	3.362	0.095
621	0.103	1.879	0.007
Average	0.989	2.744	0.467

Table A2: Detailed Scene-by-Scene Geometry Evaluation on the Waymo Open Dataset [3]. This table showcases the Chamfer Distance (CD), Root Mean Square Error (RMSE), and Depth error for specific scenes and their average, evaluating the geometric reconstruction accuracy on the Waymo Open Dataset.

A1 Implementation details

A1.1 Expanded Details on Guidance Map, Slice Operation, and Multi-Scale Fusion

To provide a more comprehensive understanding of our multi-scale bilateral grid framework, we elaborate on the key components of guidance map generation, the slice operation, and the hierarchical fusion strategy.

In the main paper, we introduced the luminance-based guidance map $I^g(u, v)$ to spatially guide our photometric corrections. Here, we elaborate on its definition and the subsequent "slice" operation.

As defined in the original submission:

$$I^g(u, v) = g(I^r(u, v)) = \text{GrayScale}(I^r(u, v)) \quad , \quad (\text{A1})$$

Scene	Geometry Evaluation		
	CD↓	RMSE↓	Depth↓
63	0.760	3.812	0.015
66	0.517	2.657	0.008
70	0.310	2.244	0.027
73	0.280	2.012	0.007
74	0.680	3.240	0.009
77	0.190	2.926	0.004
78	0.403	2.819	0.008
79	0.198	2.081	0.004
88	0.968	4.674	0.039
149	0.228	2.052	0.007
Average	0.453	2.852	0.013

Table A3: Detailed Scene-by-Scene Geometry Evaluation on the PandaSet [7]. This table provides a per-scene breakdown and average of Chamfer Distance (CD), Root Mean Square Error (RMSE), and Depth error, assessing geometric reconstruction performance on challenging nighttime scenarios from PandaSet.

Scene	Geometry Evaluation		
	CD↓	RMSE↓	Depth↓
0	0.522	3.001	0.023
1	2.071	5.845	0.085
2	0.461	3.226	0.011
3	0.348	3.677	0.014
4	1.198	4.485	0.059
5	0.694	5.069	0.040
6	0.967	6.193	0.075
8	0.743	4.436	0.045
9	0.257	1.952	0.005
Average	0.807	4.209	0.040

Table A4: Detailed Scene-by-Scene Geometry Evaluation on the Argoverse2 Dataset [6]. This table displays Chamfer Distance (CD), Root Mean Square Error (RMSE), and Depth error for individual sequences and their average, evaluating geometric accuracy on the Argoverse2 dataset.

Scene	Scene Reconstruction								Novel View Synthesis							
	Full Image			human		vehicle			Full Image			human		vehicle		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	PSNR↑	SSIM↑		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	PSNR↑	SSIM↑	
152	27.39	0.839	0.204	29.33	0.873	27.76	0.862		23.57	0.695	0.234	25.04	0.655	22.07	0.579	
164	26.52	0.829	0.232	24.75	0.789	25.60	0.793		23.16	0.711	0.262	19.86	0.464	21.74	0.595	
171	26.97	0.833	0.249	24.05	0.652	26.22	0.812		24.39	0.748	0.273	22.28	0.568	22.51	0.639	
200	27.80	0.835	0.193	N/A	N/A	28.08	0.832		25.28	0.745	0.211	N/A	N/A	24.11	0.643	
209	29.07	0.866	0.185	28.20	0.802	28.84	0.836		26.18	0.785	0.204	25.42	0.694	25.07	0.683	
359	27.95	0.859	0.168	26.98	0.795	27.25	0.839		24.43	0.723	0.190	24.06	0.629	22.72	0.647	
529	29.90	0.893	0.133	24.25	0.736	27.80	0.856		27.40	0.826	0.146	24.18	0.712	24.84	0.745	
916	25.88	0.819	0.181	28.16	0.792	26.95	0.839		22.73	0.678	0.208	25.44	0.637	22.75	0.634	
Average	27.69	0.847	0.193	26.53	0.777	27.31	0.834		24.64	0.739	0.216	23.75	0.623	23.23	0.646	

Table A5: Detailed Appearance Evaluation for Scene Reconstruction and Novel View Synthesis on the NuScenes Dataset [1]. This table presents PSNR, SSIM, and LPIPS metrics for the full image, as well as for 'human' and 'vehicle' classes, for both scene reconstruction and novel view synthesis tasks on various NuScenes sequences.

Scene	Scene Reconstruction								Novel View Synthesis							
	Full Image			human		vehicle			Full Image			human		vehicle		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	PSNR↑	SSIM↑		PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	PSNR↑	SSIM↑	
0	29.26	0.840	0.311	N/A	N/A	23.33	0.640		25.94	0.770	0.341	N/A	N/A	22.29	0.531	
3	27.94	0.841	0.241	21.82	0.632	N/A	N/A		24.30	0.737	0.269	20.45	0.505	N/A	N/A	
31	27.65	0.849	0.224	32.93	0.784	22.54	0.683		23.98	0.723	0.250	30.48	0.657	19.05	0.421	
233	33.71	0.801	0.482	N/A	N/A	23.55	0.686		32.81	0.777	0.488	N/A	N/A	22.37	0.645	
551	24.89	0.748	0.409	20.77	0.470	21.66	0.683		22.54	0.670	0.437	19.99	0.407	18.76	0.471	
621	31.91	0.939	0.067	26.68	0.846	26.40	0.845		29.73	0.895	0.078	24.94	0.789	24.35	0.765	
Average	29.23	0.836	0.289	25.55	0.683	23.50	0.707		26.55	0.762	0.310	23.97	0.590	21.36	0.567	

Table A6: Detailed Appearance Evaluation for Scene Reconstruction and Novel View Synthesis on the Waymo Open Dataset [3]. This table details PSNR, SSIM, and LPIPS metrics for full images and specific object classes ('human', 'vehicle') during scene reconstruction and novel view synthesis on selected Waymo sequences.

Scene	Scene Reconstruction						Novel View Synthesis							
	Full Image			human		vehicle	Full Image			human		vehicle		
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	PSNR↑	SSIM↑
63	29.95	0.905	0.226	29.44	0.783	22.32	0.726	27.32	0.854	0.258	26.93	0.613	18.49	0.447
66	31.73	0.931	0.198	29.94	0.845	20.78	0.681	28.92	0.890	0.215	26.84	0.738	19.55	0.591
70	31.09	0.909	0.260	N/A	N/A	21.35	0.740	28.35	0.862	0.287	N/A	N/A	21.20	0.674
73	32.13	0.931	0.198	31.26	0.868	23.33	0.806	29.16	0.889	0.214	N/A	N/A	21.84	0.706
74	29.85	0.914	0.207	30.99	0.874	24.44	0.849	27.13	0.862	0.228	27.82	0.740	22.23	0.740
77	32.60	0.937	0.165	32.28	0.852	25.42	0.855	29.77	0.895	0.178	29.02	0.730	22.12	0.721
78	31.30	0.922	0.181	33.54	0.896	25.27	0.812	28.39	0.869	0.197	30.74	0.809	22.86	0.683
79	31.42	0.902	0.228	31.91	0.850	26.06	0.843	28.14	0.832	0.251	28.98	0.726	23.08	0.702
88	25.09	0.787	0.269	22.10	0.670	21.90	0.738	21.95	0.623	0.305	20.19	0.362	18.46	0.444
149	32.31	0.926	0.202	N/A	N/A	24.04	0.791	29.73	0.891	0.215	N/A	N/A	23.37	0.728
Average	30.75	0.906	0.213	30.18	0.830	23.49	0.784	27.89	0.847	0.235	27.22	0.674	21.32	0.644

Table A7: Detailed Appearance Evaluation for Scene Reconstruction and Novel View Synthesis on PandaSet [7]. This table outlines PSNR, SSIM, and LPIPS metrics for full image and object-specific ('human', 'vehicle') evaluations in both scene reconstruction and novel view synthesis tasks using PandaSet nighttime sequences.

Scene	Scene Reconstruction						Novel View Synthesis							
	Full Image			human		vehicle		Full Image			human		vehicle	
	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	PSNR↑	SSIM↑	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	PSNR↑	SSIM↑
0	25.72	0.871	0.181	25.54	0.822	26.58	0.825	22.83	0.765	0.199	23.73	0.734	23.31	0.682
1	22.18	0.769	0.302	21.76	0.681	22.87	0.798	20.20	0.655	0.323	18.22	0.472	18.48	0.478
2	27.36	0.888	0.166	25.33	0.785	28.93	0.884	24.83	0.803	0.183	22.32	0.620	23.63	0.669
3	26.55	0.893	0.123	23.24	0.811	25.58	0.875	24.69	0.826	0.141	21.99	0.756	23.10	0.769
4	22.65	0.804	0.272	22.73	0.720	23.76	0.812	20.55	0.685	0.290	19.95	0.518	19.58	0.493
5	24.19	0.848	0.207	23.69	0.760	23.75	0.790	22.07	0.738	0.226	21.12	0.598	20.32	0.580
6	23.56	0.795	0.275	19.80	0.588	21.05	0.757	21.86	0.703	0.292	18.15	0.446	17.76	0.498
8	23.47	0.850	0.183	25.63	0.787	23.66	0.851	21.25	0.756	0.199	20.57	0.531	19.58	0.628
9	26.48	0.925	0.091	22.44	0.700	24.42	0.853	24.91	0.875	0.100	21.04	0.614	21.96	0.732
Average	24.68	0.849	0.200	23.35	0.739	24.51	0.827	22.58	0.756	0.217	20.79	0.588	20.86	0.615

Table A8: Detailed Appearance Evaluation for Scene Reconstruction and Novel View Synthesis on the Argoverse2 Dataset [6]. This table shows PSNR, SSIM, and LPIPS for full image and object-focused ('human', 'vehicle') assessments across scene reconstruction and novel view synthesis on the Argoverse2 dataset.

Scene	Method	Reconstruction			Novel View Synthesis			Geometry		
		PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	CD ↓	RMSE ↓	Depth ↓
152	ChatSim	25.37	0.811	0.251	1.592	3.637	0.085	22.72	0.694	0.276
	Ours	27.09	0.825	0.231	1.309	3.540	0.040	23.80	0.703	0.256
164	ChatSim	24.08	0.788	0.299	1.198	2.615	0.186	22.07	0.708	0.321
	Ours	26.16	0.802	0.273	0.972	2.523	0.101	23.79	0.718	0.293
171	ChatSim	25.00	0.801	0.310	1.420	2.981	0.105	23.31	0.736	0.326
	Ours	26.95	0.813	0.289	1.114	2.895	0.057	24.77	0.744	0.307
200	ChatSim	24.85	0.799	0.230	1.690	3.740	0.035	23.47	0.732	0.246
	Ours	27.54	0.816	0.220	1.289	3.619	0.017	25.43	0.745	0.235
209	ChatSim	25.77	0.818	0.268	1.841	3.805	0.118	24.30	0.763	0.281
	Ours	27.93	0.833	0.243	1.445	3.691	0.061	25.93	0.776	0.255
359	ChatSim	25.76	0.823	0.220	1.498	3.701	0.073	23.46	0.714	0.238
	Ours	27.17	0.834	0.208	1.317	3.647	0.036	24.31	0.719	0.227
529	ChatSim	27.19	0.864	0.191	1.071	2.853	0.077	25.70	0.812	0.201
	Ours	29.60	0.878	0.159	0.692	2.749	0.018	27.35	0.822	0.169
916	ChatSim	22.81	0.742	0.249	2.145	4.745	0.173	21.19	0.641	0.271
	Ours	23.95	0.759	0.232	1.750	4.635	0.102	21.96	0.655	0.253

Table A9: Comparative Performance Analysis against ChatSim on the NuScenes Dataset. This table provides a scene-by-scene comparison of our method ('Ours') with the ChatSim baseline across scene reconstruction (PSNR, SSIM, LPIPS), novel view synthesis (PSNR, SSIM, LPIPS), and geometry (CD, RMSE, Depth) metrics.

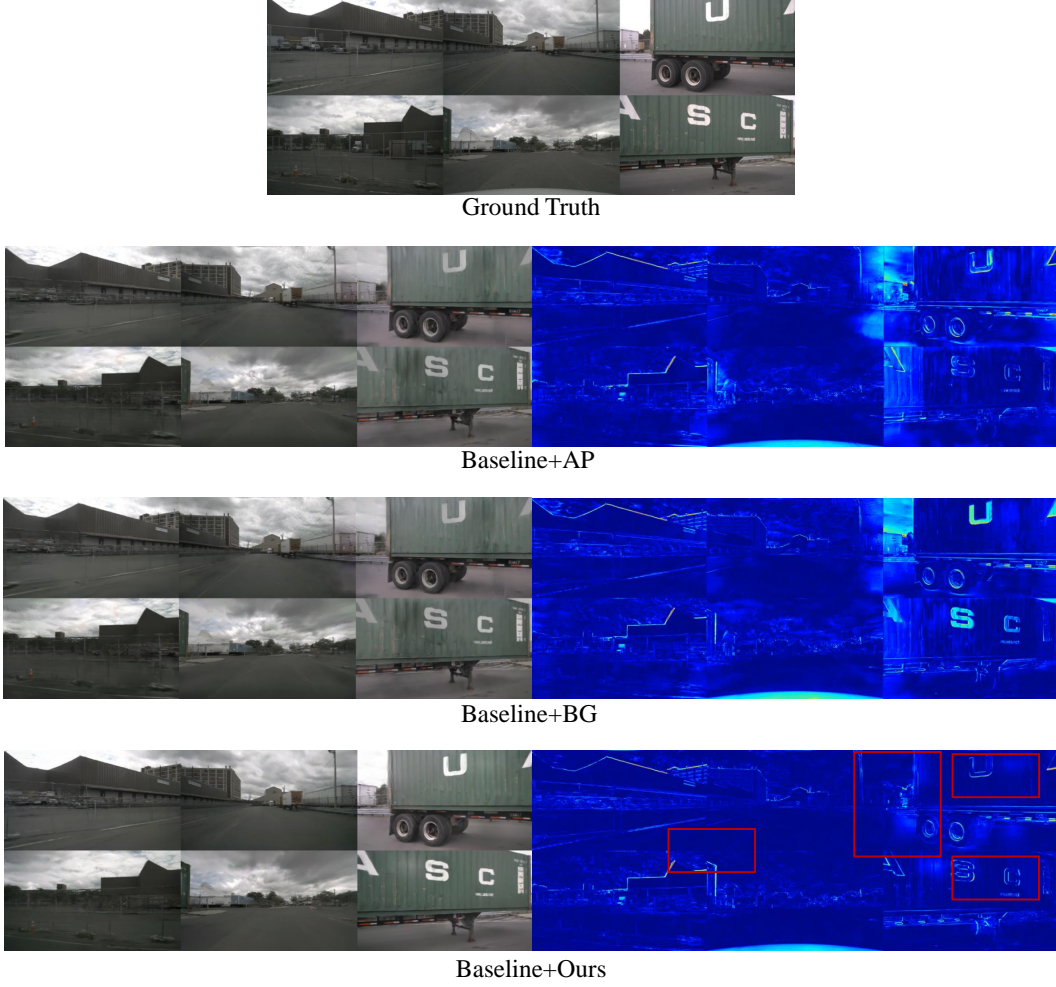


Figure A1: **Qualitative Comparison of Photometric Correction with Baseline Methods.** These figures visualize the output of our method ('Baseline+Ours') against the ground truth, a baseline with appearance codes ('Baseline+AP'), and a baseline with a single bilateral grid ('Baseline+BG'). The accompanying error maps (blue indicating lower error, red higher) and highlighted red boxes demonstrate our method's superior ability to handle complex illumination and reduce artifacts compared to traditional approaches.

the guidance map $I^g(u, v)$ is derived by applying the $GrayScale(\cdot)$ function to the rendered image $I^r(u, v)$. This function converts RGB color values to a scalar luminance intensity, normalized to the range $[0, 1]$. We chose luminance as the guidance signal because it effectively represents spatial brightness variations, capturing shadows, highlights, and illumination gradients – key factors contributing to photometric inconsistencies. This aligns with previous works in image filtering and bilateral processing [2].

The "slice" operation retrieves per-pixel transformations based on this guidance map. For each grid level l and pixel location (u, v) , the luminance value $d = I^g(u, v)$ acts as a query into the grid's intensity dimension $D^{(l)}$. As described in main paper:

$$\begin{aligned} \bar{A}^{(l)}(u, v) &= \sum_{i,j,k} w_{i,j,k}(u, v, d) \mathcal{A}^s(i, j, k), \\ w_{i,j,k}(u, v, d) &= \tau(W^{(l)} \cdot u - i) \tau(H^{(l)} \cdot v - j) \tau(D^{(l)} \cdot d - k), \end{aligned} \quad (\text{A2})$$

the level-specific affine transformation $\bar{A}^{(l)}(u, v)$ is computed through trilinear interpolation. The weights $w_{i,j,k}(u, v, d)$ are determined by the linear interpolation kernel $\tau(t) = \max(1 - |t|, 0)$, ensuring smooth blending of neighboring affine transformation coefficients $\mathcal{A}^s(i, j, k)$ around the

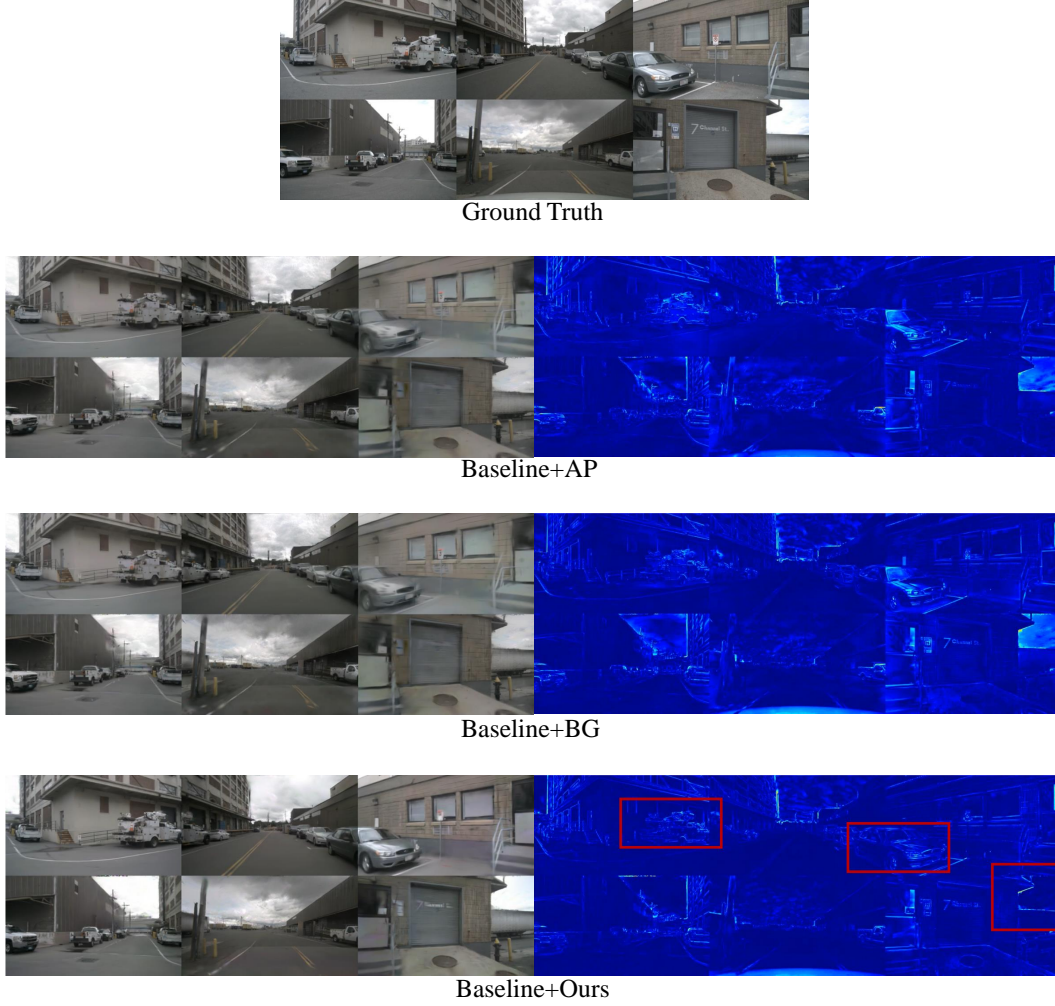


Figure A2: **Qualitative Comparison of Photometric Correction with Baseline Methods.**

Scene	Method	Reconstruction			Novel View Synthesis			Geometry		
		PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow	CD \downarrow	RMSE \downarrow	Depth \downarrow
152	StreetGS	25.49	0.813	0.254	1.611	3.648	0.089	22.73	0.695	0.280
	Ours	27.34	0.828	0.231	1.343	3.560	0.038	23.90	0.706	0.256
164	StreetGS	25.04	0.813	0.273	1.187	2.605	0.179	22.55	0.724	0.296
	Ours	27.41	0.827	0.248	0.956	2.523	0.096	24.29	0.734	0.269
171	StreetGS	25.62	0.811	0.305	1.355	2.958	0.115	23.63	0.743	0.323
	Ours	27.79	0.822	0.281	1.184	2.910	0.067	25.21	0.751	0.300
200	StreetGS	25.32	0.811	0.230	1.878	3.925	0.041	23.75	0.739	0.247
	Ours	28.12	0.829	0.219	1.418	3.817	0.020	25.67	0.752	0.235
209	StreetGS	26.53	0.840	0.243	1.932	3.813	0.125	24.77	0.779	0.258
	Ours	29.05	0.854	0.217	1.481	3.697	0.061	26.53	0.791	0.231
359	StreetGS	26.08	0.831	0.211	1.539	3.710	0.075	23.53	0.716	0.230
	Ours	27.63	0.843	0.199	1.299	3.659	0.039	24.45	0.723	0.218
529	StreetGS	27.50	0.871	0.186	1.065	2.883	0.063	25.96	0.819	0.197
	Ours	30.09	0.884	0.158	0.718	2.795	0.020	27.69	0.827	0.169
916	StreetGS	24.34	0.787	0.223	2.268	4.813	0.175	22.21	0.680	0.245
	Ours	25.80	0.804	0.204	1.781	4.709	0.106	23.11	0.692	0.226

Table A10: Comparative Performance Analysis against StreetGS on the NuScenes Dataset. This table presents a detailed per-scene comparison of our approach ('Ours') with the StreetGS baseline, evaluating scene reconstruction (PSNR, SSIM, LPIPS), novel view synthesis (PSNR, SSIM, LPIPS), and geometry (CD, RMSE, Depth) metrics.

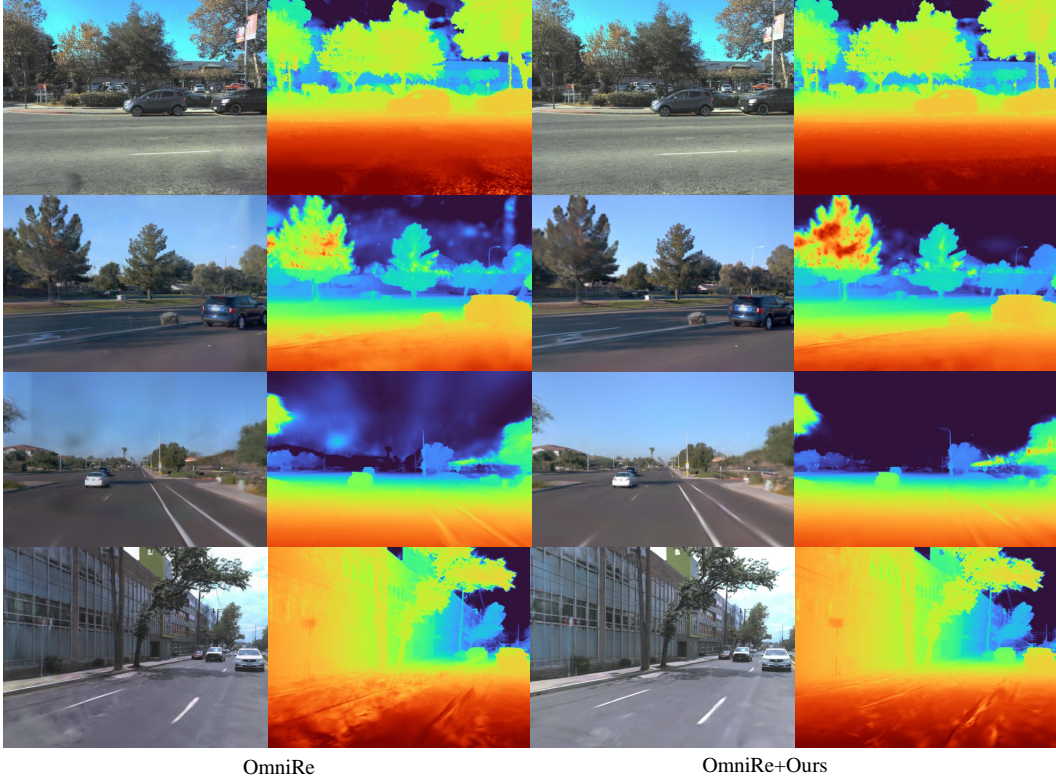


Figure A3: **Qualitative Comparison with the OmniRe.** This figure showcases a side-by-side visual comparison of renderings and depth maps produced by the original OmniRe framework versus OmniRe integrated with our proposed method ('OmniRe+Ours'). The results highlight the enhancements in image fidelity and depth map coherence achieved by our approach.

queried luminance value d . $W^{(l)}$, $H^{(l)}$, and $D^{(l)}$ represent the dimensions of the bilateral grid at level l . This formulation explicitly shows how the guidance map drives the spatially-varying photometric correction by modulating the retrieved affine transformations at each pixel. Furthermore, the "slice" operation, which retrieves per-pixel transformations, is a crucial step in our framework. As detailed in Eq. (A2), this operation utilizes trilinear interpolation to smoothly blend affine transformations associated with grid vertices around the queried luminance value. The linear interpolation kernel $\tau(t) = \max(1 - |t|, 0)$ ensures a continuous and differentiable transformation retrieval process, which is essential for stable optimization during training. The weights $w_{i,j,k}(u, v, d)$ determine the contribution of each neighboring affine transformation $\mathcal{A}^s(i, j, k)$, effectively creating a localized and context-aware photometric adjustment based on the guidance map at each pixel location. This spatially-varying adaptation is key to addressing complex photometric inconsistencies beyond global transformations.

The main paper emphasized the efficiency gained by hierarchical fusion and downsampled guidance maps. Here, we provide further details on the computational advantages of our approach. A naive full-resolution "slice" operation at each scale would be computationally expensive. For a grid of size $W^{(l)} \times H^{(l)} \times D^{(l)}$ at level l , and an image of resolution $W_{\text{img}} \times H_{\text{img}}$, performing a full-resolution slice would require approximately $O(W_{\text{img}} \cdot H_{\text{img}})$ trilinear interpolations per level. Summing over L levels, the total cost becomes $O(L \cdot W_{\text{img}} \cdot H_{\text{img}})$.

To mitigate this, we employ downsampled guidance maps $I^{g(l)}$ of size $W^{(l)} \times H^{(l)}$. The "slice" operation is then performed at this reduced resolution. The resulting low-resolution coefficient fields are upsampled to the original image resolution using bilinear interpolation, which is computationally less expensive than trilinear interpolation. The dominant computational cost now shifts to the trilinear interpolation performed at the downsampled resolution, becoming approximately $O(\sum_{l=0}^{L-1} W^{(l)} \cdot H^{(l)})$.

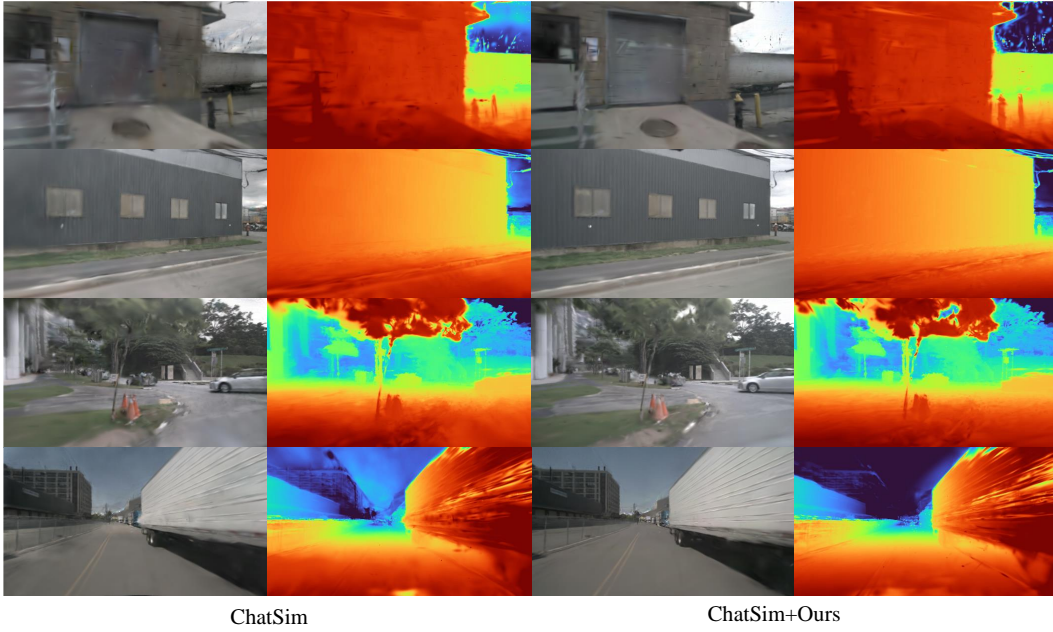


Figure A4: **Qualitative Comparison with the ChatSim.** Visual results comparing the ChatSim framework with ChatSim augmented by our method ('ChatSim+Ours'). The rendered images and corresponding depth maps illustrate the improvements in visual quality and geometric detail provided by our multi-scale bilateral grid framework.

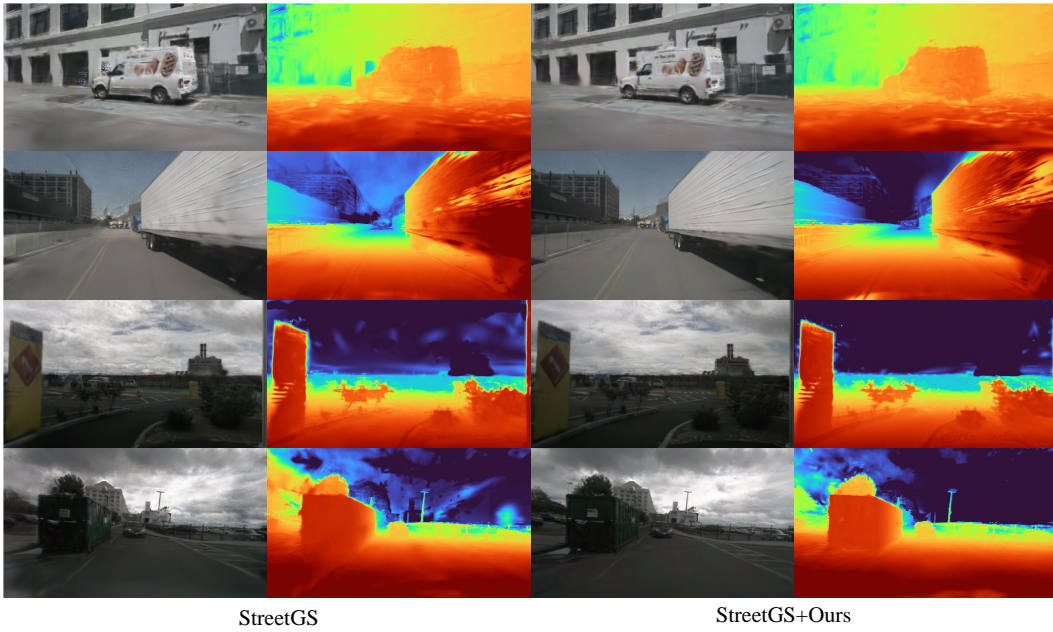


Figure A5: **Qualitative Comparison with the StreetGS.** This figure presents a visual comparison between the StreetGS framework and StreetGS enhanced with our method ('StreetGS+Ours'). The displayed images and depth maps demonstrate the capability of our approach to improve rendering realism and depth accuracy in challenging driving scenarios.

59 $H^{(l)}$). By choosing appropriate downsampling factors for each level, we can significantly reduce the
60 overall computational burden. For example, if we halve the resolution of the guidance map at each
61 level, the computational cost of slicing is drastically reduced.

62 Furthermore, the hierarchical fusion:

$$I^e = \mathcal{T}^{(L-1)} \circ \mathcal{T}^{(L-2)} \circ \dots \circ \mathcal{T}^{(0)}(I^r), \quad (\text{A3})$$

63 where $\mathcal{T}^{(l)}(x) = \bar{M}^{(l)} \odot x + \bar{T}^{(l)}$, allows for efficient combination of transformations. Function
64 composition is computationally efficient, especially for affine transformations, and the sequential
65 nature of the fusion allows for a coarse-to-fine refinement process. This hierarchical structure not
66 only improves efficiency but also facilitates learning scale-specific photometric corrections.

67 Our hierarchical fusion strategy provides an intuitive way to understand the scale-dependent nature
68 of the learned photometric corrections. At the coarsest scale (e.g., $l = 0$), the transformation $\mathcal{T}^{(0)}$
69 captures a global or near-global affine transformation, similar in spirit to appearance codes. This
70 can be viewed as learning a transformation $\bar{A}^{(0)}$ that is approximately constant across the image,
71 effectively correcting for global photometric biases.

72 Subsequent scales, $\mathcal{T}^{(1)}, \mathcal{T}^{(2)}, \dots, \mathcal{T}^{(L-1)}$, then learn residual transformations. For instance, $\mathcal{T}^{(1)}$
73 refines the globally corrected image from $\mathcal{T}^{(0)}$ by applying transformations that capture regional
74 variations. Mathematically, we can express the refined image after the first two scales as:

$$I^{(2)} = \mathcal{T}^{(1)} \circ \mathcal{T}^{(0)}(I^r) = \mathcal{T}^{(1)}(I^{(1)}) \quad (\text{A4})$$

75 where $I^{(1)} = \mathcal{T}^{(0)}(I^r)$. This shows that $\mathcal{T}^{(1)}$ is learning to correct the already globally corrected
76 image $I^{(1)}$, focusing on regional discrepancies not addressed by the coarse scale. This hierarchical,
77 residual refinement continues up to the finest scale, allowing for increasingly localized and detailed
78 photometric adjustments. This scale-dependent approach enables the framework to effectively
79 decouple global and local photometric variations, leading to improved photometric consistency and
80 geometric accuracy.

81 A1.2 Expanded Details on Training

82 We jointly optimize our multi-scale Gaussian scene representation by minimizing a composite loss
83 function that combines reconstruction accuracy with regularization terms:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{recon}} + \lambda_{\text{TV}} \mathcal{L}_{\text{TV}} + \lambda_{\text{circle}} \mathcal{L}_{\text{circle}} \quad (\text{A5})$$

84 where $\mathcal{L}_{\text{recon}}$ is the primary reconstruction loss, and \mathcal{L}_{TV} and $\mathcal{L}_{\text{circle}}$ are regularization terms, with λ_{TV}
85 and λ_{circle} being their respective weighting factors.

86 A1.2.1 Reconstruction Loss

87 The core reconstruction loss, $\mathcal{L}_{\text{recon}}$, drives the accurate reproduction of both RGB and depth informa-
88 tion. It is defined as:

$$\mathcal{L}_{\text{recon}} = \lambda_r \mathcal{L}_1 + (1 - \lambda_r) \mathcal{L}_{\text{SSIM}} + \lambda_d \mathcal{L}_d + \lambda_o \mathcal{L}_o \quad (\text{A6})$$

89 Here, \mathcal{L}_1 and $\mathcal{L}_{\text{SSIM}}$ measure the difference between the rendered and ground truth images using L1
90 loss and Structural Similarity Index Measure (SSIM), respectively. λ_r balances these two image-space
91 losses. \mathcal{L}_d represents the loss between the rendered depth and the ground truth LiDAR depth. Finally,
92 \mathcal{L}_o is an opacity regularization term that encourages alignment with a non-sky mask, ensuring that
93 opacity is concentrated on relevant scene geometry.

94 A1.2.2 Adaptive Total Variation Regularization Smoothness

95 To encourage smoothness and reduce noise in the optimized grid representations while preserving
96 important details, we incorporate a Total Variation (TV) regularization term, \mathcal{L}_{TV} , at each level of the
97 multi-scale representation. This term penalizes large gradients in the grid features:

$$\mathcal{L}_{\text{TV}} = \sum_l k^{(l)} \cdot \frac{1}{|\mathcal{A}^{(l)}|} \sum_{i,j,k} \sum_{\mathbb{D} \in \{x,y,z\}} \left\| \Delta_{\mathbb{D}} \mathcal{A}^{(l)}(i,j,k) \right\|_2^2 \quad (\text{A7})$$

98 The adaptive weight $k^{(l)}$ for each level l is proportional to the grid size, as defined by:

$$k^{(l)} = a\sqrt{H^{(l)} \cdot W^{(l)} \cdot D^{(l)}} + b \quad (\text{A8})$$

99 where $H^{(l)}, W^{(l)}, D^{(l)}$ are the dimensions of the grid at level l , and a and b are hyperparameters.
 100 This adaptive weighting strategy applies lighter regularization to lower-resolution (coarse) grids,
 101 which capture global structures and edges, thereby preserving essential features. Conversely, higher-
 102 resolution (fine) grids, which are more susceptible to noise and overfitting high-frequency details,
 103 receive stronger smoothing for improved stability and generalization.

104 **A1.2.3 Circle Regularization for Photometric Consistency**

105 While our optimization prioritizes geometric reconstruction, this can sometimes lead to discrepancies
 106 between the rendered image I^r and the ground truth image I^{gt} , potentially degrading image quality,
 107 especially in novel views. To mitigate this and constrain the noise introduced by photometric
 108 corrections, we introduce a circle regularization loss, $\mathcal{L}_{\text{circle}}$:

$$\mathcal{L}_{\text{circle}} = \sum_{(u,v) \in \mathcal{S}} \|I^r(u,v) - \bar{A}^{-1}(I^{gt}(u,v))\|_2^2 \quad (\text{A9})$$

109 where \mathcal{S} denotes the set of pixel coordinates. This loss encourages the rendered image I^r to be
 110 reconstructible from the ground truth image I^{gt} via an inverse appearance transformation \bar{A}^{-1} . This
 111 effectively tightens the permissible space for photometric corrections, preventing overly aggressive
 112 alterations that could harm perceptual quality.

113 **A1.2.4 Coarse-to-Fine Optimization Strategy**

114 We employ a coarse-to-fine optimization strategy by utilizing level-dependent learning rates ($1 \times$
 115 $10^{-5}, 3 \times 10^{-5}, 1 \times 10^{-4}$ from coarse to fine). Specifically, higher learning rates are assigned to
 116 coarser grids, with progressively lower rates applied to finer grids within the pyramid.

117 This approach is crucial for stable and effective training. The coarse grids, responsible for capturing
 118 the global appearance and structure, can rapidly learn the overall scene illumination and color
 119 palette during the initial training stages due to their higher learning rates. Subsequently, the finer
 120 grids, operating with lower learning rates, focus on refining high-frequency photometric details
 121 hierarchically. This staged optimization enhances stability by ensuring that global scene consistency
 122 is established before intricate details are incorporated, ultimately leading to a more robust and accurate
 123 scene representation.

124 **A1.3 Dynamic Rendering for Real-World ISP Adaptation**

125 Consistent multi-view training is crucial for accurate geometry reconstruction. However, autonomous
 126 driving applications demand compatibility with dynamic real-world Image Signal Processor (ISP)
 127 pipelines. These real-world ISPs are inherently heterogeneous, varying across different cameras and
 128 changing over time due to factors such as evolving lighting conditions and sensor variations. This
 129 heterogeneity introduces a significant domain gap: synthetic training data, typically rendered with a
 130 fixed ISP, differs substantially from real-world imagery captured using these dynamic ISPs.

131 To bridge this domain gap and enhance the realism of rendered images under dynamic ISP conditions,
 132 we employ an interpolation strategy within our multi-scale bilateral grid framework. This approach is
 133 designed to adapt the rendering process to novel test images that simulate such dynamic ISP effects.

134 Our strategy involves two key practical techniques:

135 (1) **Temporal Proximity Search:** For a novel test image, we first identify the two training timestamps
 136 from the same camera that are temporally closest to the test image’s timestamp. This leverages the
 137 inherent temporal coherence of ISP parameters, as ISP settings tend to change smoothly over short
 138 periods for a given camera.

139 (2) **Scale-Specific Bilateral Grid Interpolation:** We then perform linear interpolation specifically
 140 on the coarse and medium-scale bilateral grids ($\mathcal{A}^{(0)}, \mathcal{A}^{(1)}$) derived from these two identified training
 141 timestamps.

The effectiveness of this temporal interpolation stems from the strong correlation between ISP parameters, timestamps, and camera position. Under temporally and spatially proximate conditions, scene illumination and composition generally remain relatively consistent. Consequently, variations in ISP settings within such close temporal and spatial proximity can often be approximated by localized linear photometric transformations. The interpolated grids thus enable the output to adapt to dynamic ISP characteristics, effectively reducing the domain gap.

Let t_{novel} be the timestamp of a novel test image. We identify the two temporally nearest training timestamps from the same camera, denoted as t_1 and t_2 , such that $t_1 \leq t_{\text{novel}} \leq t_2$. Let $\mathcal{A}_t^{(l)}$ represent the bilateral grid tensor at level $l \in \{0, 1, 2\}$ learned during training for a timestamp t . The interpolation for the coarse and medium levels ($l \in \{0, 1\}$) is formulated as:

$$\hat{\mathcal{A}}^{(l)} = \omega \mathcal{A}_{t_1}^{(l)} + (1 - \omega) \mathcal{A}_{t_2}^{(l)} \quad (\text{A10})$$

where $\hat{\mathcal{A}}^{(l)}$ is the interpolated bilateral grid tensor at level l for the novel timestamp t_{novel} . The temporal interpolation weight, ω , is determined by the proximity of t_{novel} to t_1 and t_2 :

$$\omega = \frac{t_2 - t_{\text{novel}}}{t_2 - t_1} \quad (\text{A11})$$

This weighting ensures that $\hat{\mathcal{A}}^{(l)}$ is influenced more by the grid of the closer timestamp. For instance, if $t_{\text{novel}} = t_1$, then $\omega = 1$, resulting in $\hat{\mathcal{A}}^{(l)} = \mathcal{A}_{t_1}^{(l)}$. Conversely, if $t_{\text{novel}} = t_2$, then $\omega = 0$, and $\hat{\mathcal{A}}^{(l)} = \mathcal{A}_{t_2}^{(l)}$.

The fine-scale grid ($l = 2$) is not used during interpolation. The rationale is that the fine-scale grid is primarily intended to capture scene-intrinsic details rather than global ISP variations, making interpolation at this level unnecessary and potentially counterproductive. By utilizing these interpolated grids $\hat{\mathcal{A}}^{(0)}, \hat{\mathcal{A}}^{(1)}$, we proceed with the multi-scale bilateral grid framework as detailed in the main paper. This allows us to render the image for the novel timestamp t_{novel} while effectively adapting our rendering pipeline to the challenges posed by dynamic ISP conditions.

A1.4 Detailed Analysis of Appearance Codes and Bilateral Grids

This section concisely analyzes the limitations of appearance codes and traditional bilateral grids for photometric correction in dynamic autonomous driving scenarios, motivating our multi-scale approach.

Appearance codes use a global affine transformation for photometric correction:

$$I^e = A_{3 \times 4} \times I^r. \quad (\text{A12})$$

Designed for scene-wide adjustments of brightness/contrast, global methods assume uniform correction is sufficient. However, this fails in complex real-world driving environments with localized photometric variations from specularities, complex illumination, and dynamic lighting. Global transformations inherently cannot address these nuances, leading to spatially inconsistent corrections and hindering detailed 3D reconstruction. Appearance codes are thus a limited "one-size-fits-all" solution inadequate for diverse real-world photometric distortions.

Bilateral grids offer pixel-wise correction:

$$I^e(u, v) = A(u, v) \odot I^r(u, v). \quad (\text{A13})$$

While promising finer local adjustments, high-resolution bilateral grids for dynamic scenes face computational and practical challenges, limiting their effectiveness.

Computational Bottlenecks of High-Resolution Bilateral Grids.

a) **Dimensionality Explosion.** For N cameras and T time-steps, a single high-resolution grid scales as $O(N \times T \times H \times W)$. This massive parameter space drastically increases memory and computational demands, rendering per-pixel transformations across multi-view video impractical on standard hardware.

182 b) **Optimization Complexity.** Optimizing a single monolithic grid for dynamic scenes is
183 highly complex. Capturing both global scene structure and local dynamic variations requires
184 iterative algorithms to reconcile conflicting constraints (photometric consistency vs. motion
185 smoothness), leading to slow convergence, local minima, and unstable optimization.

186 **Practical Deployment Impasse.** Scene complexity variations in driving scenarios exacerbate
187 practical issues with single bilateral grids. Achieving a balance between reconstruction fidelity and
188 generalization with a single grid is highly scene-dependent, requiring manual tuning and leading to
189 key practical roadblocks:

190 a) **Overfitting vs. Undersmoothing Conundrum.** High-resolution grids overfit to noise. Low-
191 resolution grids undersmooth fine photometric details. Finding a "Goldilocks" resolution for
192 diverse scenes is extremely difficult and requires heuristic adjustments.

193 b) **Smoothness Regularization Dilemma.** Excessive regularization in single grids over-
194 smooths valid discontinuities (shadow boundaries). Insufficient regularization leads to
195 instability and artifacts like floaters. Balancing regularization for stability and detail preser-
196 vation is a significant challenge.

197 Parameter sensitivity and these trade-offs make direct application of single bilateral grids impractical
198 for scalable real-world deployments, especially in autonomous driving. Computational intractability
199 and tuning difficulty highlight their sub-optimal nature for complex photometric inconsistencies,
200 motivating our multi-scale bilateral grid framework.

201 **A2 Datasets**

202 In the main paper, we presented results on four widely-used autonomous driving datasets: Waymo,
203 NuScenes, Argoverse, and PandaSet to rigorously evaluate the robustness and generalization capabili-
204 ties of our proposed multi-scale bilateral grid framework. A core objective of our experiments was to
205 demonstrate that our method is not only effective under ideal conditions but also performs reliably
206 across a diverse range of real-world autonomous driving scenarios. By testing on datasets with varying
207 characteristics, we aim to provide strong evidence for the practical applicability and generalizability
208 of our approach. This section provides dataset-specific details regarding our evaluation protocol.

209 **A2.1 Waymo Open Dataset**

210 For the Waymo Open Dataset [3], we employ all five cameras and all the Li-
211 DAR sensors provided within the dataset. We utilize the entirety of the training
212 and validation sets without specific sequence selection, leveraging all available cam-
213 era views and LiDAR data. We conduct our experiments on the following 6 se-
214 quences: segment-10017090168044687777, segment-10061305430875486848,
215 segment-10584247114982259878, segment-15090871771939393635,
216 segment-4458730539804900192, segment-5835049423600303130. which are selected
217 according to open-source framework [4, 5].

218 **A2.2 NuScenes**

219 When evaluating on NuScenes [1], we utilize all six available cameras and all the LiDAR sensors. We
220 select the following 8 sequences for our experiments: 152, 164, 171, 200, 209, 359, 529,
221 916, which is an extension of the dataset used by [4]. Similar to other datasets, we address ego-vehicle
222 visibility by cropping the bottom 80 pixels from the back camera images.

223 **A2.3 PandaSet**

224 For PandaSet [7], we utilize the full sensor rig, consisting of six cameras and one LiDAR unit.
225 We specifically evaluate on the following 10 sequences: 063, 066, 070, 073, 074, 077, 078,
226 079, 088, 149, which are all challenging and complex nighttime scenarios. To mitigate ego-vehicle
227 artifacts, we apply a consistent crop to the back camera images, removing the bottom 260 pixels
228 which often contain views of the vehicle itself.

A2.4 Argoverse2

For evaluation on Argoverse2 [6], we leverage the seven ring cameras and both LiDAR sensors available in the dataset. In line with [4], we conduct our experiments on the following 9 sequences: 05fa5048-f355-3274-b565-c0ddc547b315, 0b86f508-5df9-4a46-bc59-5b9536dbde9f, 185d3943-dd15-397a-8b2e-69cd86628fb7, 25e5c600-36fe-3245-9cc0-40ef91620c22, 27be7d34-ecb4-377b-8477-ccfd7cf4d0bc, 280269f9-6111-311d-b351-ce9f63f88c81, 2f2321d2-7912-3567-a789-25e46a145bda, 44adf4c4-6064-362f-94d3-323ed42cfda9, 5589de60-1727-3e3f-9423-33437fc5da4b. To minimize ego-vehicle interference, we apply a bottom crop of 250 pixels to the front center, rear left, and rear right camera views, effectively removing regions that may contain the vehicle’s hood or trunk.

A3 Detail Experiments Results

This section presents a comprehensive breakdown of our model’s performance across various standard autonomous driving datasets. We provide detailed metrics for geometry and appearance evaluation, including scene reconstruction and novel view synthesis.

Geometry Evaluation. The geometric accuracy of our method was rigorously evaluated on multiple datasets. The tables below (Tab. A1, Tab. A2, Tab. A3, Tab. A4) show per-scene and average results for Chamfer Distance (CD), Root Mean Square Error (RMSE), and Depth metrics.

Appearance Evaluation. We evaluated the appearance quality for both scene reconstruction and novel view synthesis. The metrics used include Peak Signal-to-Noise Ratio (PSNR), Structural Similarity Index Measure (SSIM), and Learned Perceptual Image Patch Similarity (LPIPS). Results are often broken down for the full image, as well as for specific classes like "human" and "vehicle". Tab. A5, Tab. A6, Tab. A7, and Tab. A8 show the detail results.

Comparisons with State-of-the-Art. Tab. A9 and Tab. A10 provide comparisons with SOTA methods like ChatSim and StreetGS on the NuScenes dataset for scene reconstruction and novel view synthesis across reconstruction, novel view synthesis, and geometry metrics. These tables show per-scene breakdowns, comparing our method ("Ours") against baselines, demonstrating the efficacy of our approach.

A4 Additional Visualization Results

To further illustrate the qualitative performance of our method, we provide additional visualization results. These figures showcase comparisons against baselines and ground truth data across various challenging scenarios. Fig. A1 and Fig. A2 present visualization results comparing our method ("Baseline+Ours") with "Baseline+AP" (Appearance Codes) and "Baseline+BG" (Bilateral Grids) against the Ground Truth. The error maps (shown in blue/red) highlight areas where our method achieves lower reconstruction error, particularly in regions with complex lighting and textures, as indicated by the red boxes in the "Baseline+Ours" error maps. Fig. A3, Fig. A4, and Fig. A5 offer qualitative comparisons of our method integrated with other frameworks (OmniRe+Ours, ChatSim+Ours, StreetGS+Ours) against the original frameworks (OmniRe, ChatSim, StreetGS). The images typically show a side-by-side comparison of the rendered image and its corresponding depth map. These visualizations demonstrate improvements in rendering quality and depth prediction achieved by incorporating our approach. For example, the depth maps in "OmniRe+Ours" appear more consistent and detailed compared to "OmniRe". Similarly, "ChatSim+Ours" and "StreetGS+Ours" show enhanced visual fidelity and depth accuracy over their respective baselines.

References

- [1] 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.
- [2] Jiawen Chen, Sylvain Paris, and Frédo Durand. Real-time edge-aware image processing with the bilateral grid. *ACM Transactions on Graphics (TOG)*, 26(3):103–es, 2007.

- 278 [3] Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo,
279 Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo
280 open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*,
281 pages 2446–2454, 2020.
- 282 [4] Adam Tonderski, Carl Lindström, Georg Hess, William Ljungbergh, Lennart Svensson, and Christoffer
283 Petersson. Neurad: Neural rendering for autonomous driving. In *Proceedings of the IEEE/CVF Conference*
284 *on Computer Vision and Pattern Recognition*, pages 14895–14904, 2024.
- 285 [5] Yuxi Wei, Zi Wang, Yifan Lu, Chenxin Xu, Changxing Liu, Hao Zhao, Siheng Chen, and Yanfeng Wang.
286 Editable scene simulation for autonomous driving via collaborative llm-agents. In *Proceedings of the*
287 *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 15077–15087, 2024.
- 288 [6] Benjamin Wilson, William Qi, Tanmay Agarwal, John Lambert, Jagjeet Singh, Siddhesh Khandelwal, Bowen
289 Pan, Ratnesh Kumar, Andrew Hartnett, Jhony Kaesemodel Pontes, et al. Argoverse 2: Next generation
290 datasets for self-driving perception and forecasting. *arXiv preprint arXiv:2301.00493*, 2023.
- 291 [7] Pengchuan Xiao, Zhenlei Shao, Steven Hao, Zishuo Zhang, Xiaolin Chai, Judy Jiao, Zesong Li, Jian Wu,
292 Kai Sun, Kun Jiang, et al. Pandaset: Advanced sensor suite dataset for autonomous driving. In *2021 IEEE*
293 *international intelligent transportation systems conference (ITSC)*, pages 3095–3101. IEEE, 2021.