

RenderFormer-V2: Neural Rendering with Heterogeneous Scene Primitives

Chong Zeng¹, Yue Dong², Pieter Peers³, Lvmin Zhang¹, and Maneesh Agrawala¹

¹ Stanford University, USA

² Microsoft Research, China

³ College of William & Mary, USA

A Additional Implementation Details

Table 1 summarizes the key hyper-parameters of RenderFormer-V2’s two-stage transformer pipeline. Both stages share the same model dimension (768), number of attention heads (6), and Feed-Forward Network (FFN) dimension (3072). The sparse attention in the view-independent stage, consisting of a local sliding window (total size 512) combined with global attention sink tokens, is implemented using PyTorch’s `flex_attention` API [2]. We also employ QK-Norm [3] to stabilize the attention mechanism. The complete RenderFormer-V2 model comprises approximately 207M parameters in total.

Table 1: Architectural details of RenderFormer-V2’s two-stage transformer pipeline.

	View-indep.	View-dep.
Layers	12	6
Model Dimension	768	768
Attention Heads	6	6
Attention Type	Sparse Self-Attn.	Cross-Attn. + Swin Self-Attn.
FFN Dimension	3072	3072
FFN Activation		SwiGLU
Normalization		RMSNorm

B Qualitative and Quantitative Comparison

Figure 1 and 2 show qualitative comparisons for all the scenes from the main submission with respect to reference Blender Cycles path-traced renderings. For each scene we also show a difference image (scaled $5\times$ to better show the differences) and a FLIP error image [1]. In addition, we list the total number of tokens per scene, and the PSNR, SSIM, LPIPS and FLIP errors.

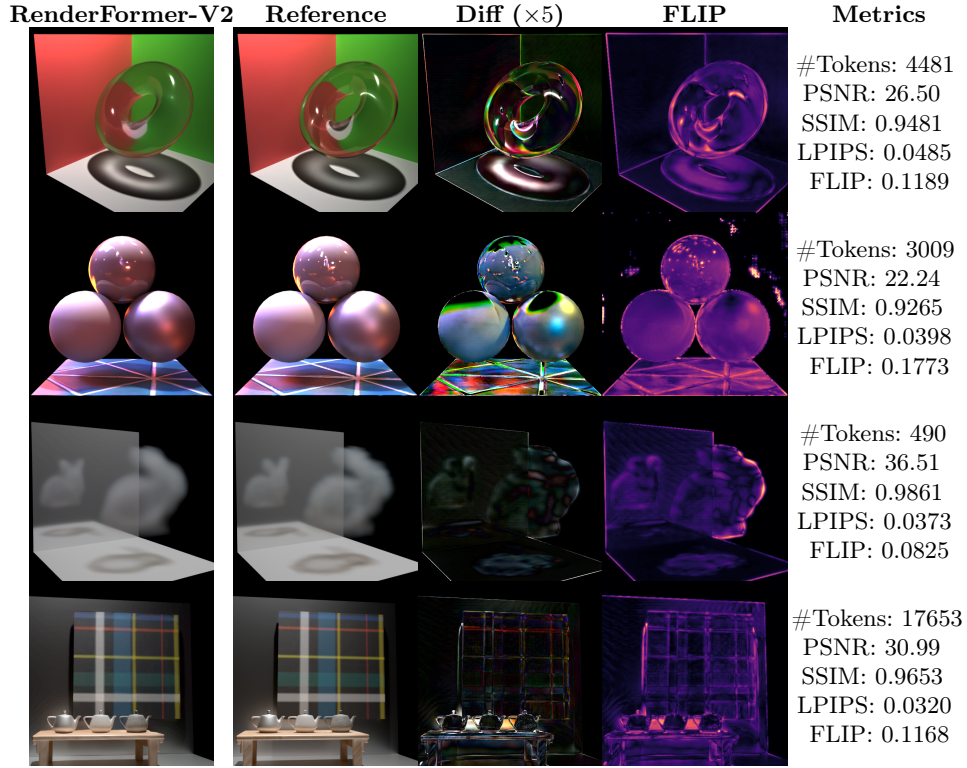


Fig. 1: Scenes from the main submission rendered with RenderFormer-V2 and compared to path-traced reference images.

1. **Transparent Torus:** (Figure 1, 1st row) the differences are mainly visible on high curvature areas of the refractive torus, as well as in the intensity of the caustic on the ground plane.
2. **Environment Lit Spheres:** (2nd row) the differences are mainly due to differences at high-frequency edges in the image at texture / environment map pixel edges. Furthermore, we can also observe a slight overall brightness difference.
3. **Smoky Bunny:** (3rd row) Due to differences in how anti-aliasing is handled (Blender Cycles uses adaptive filtering), larger differences are visible at high frequency edges in the rendered images.
4. **Three Teapot Scene:** (4th row) Similar to the previous scene, differences are mainly concentrated at high frequency edges in the image. Another area of difference is the inter-reflection below the painting; RenderFormer-V2 assumes the back of the painting is also textured and reflected onto the wall.
5. **Displacement Mapped Cornell Cube:** (Figure 2, 1st row). Again most differences are in high-frequency areas, due to (1) differences in filtering, and (2) RenderFormer-V2 sometimes misses small details.

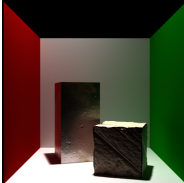
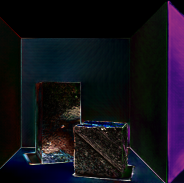
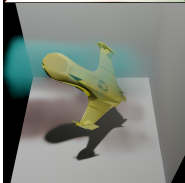
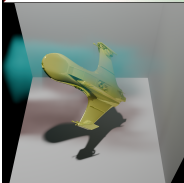
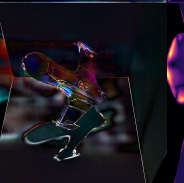
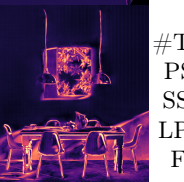
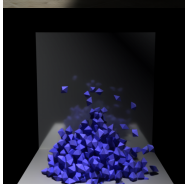
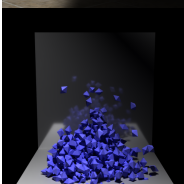
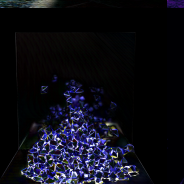
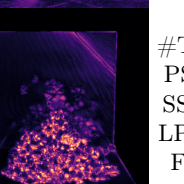
RenderFormer-V2	Reference	Diff ($\times 5$)	FLIP	Metrics
				#Tokens: 5633 PSNR: 30.97 SSIM: 0.9297 LPIPS: 0.0319 FLIP: 0.1845
				#Tokens: 36353 PSNR: 27.05 SSIM: 0.9633 LPIPS: 0.0648 FLIP: 0.1605
				#Tokens: 29276 PSNR: 24.34 SSIM: 0.9171 LPIPS: 0.0712 FLIP: 0.1992
				#Tokens: 3993 PSNR: 26.41 SSIM: 0.9243 LPIPS: 0.0474 FLIP: 0.1031

Fig. 2: Scenes from the main submission rendered with RenderFormer-V2 and compared to path-traced reference images.

6. **Spaceship in Smoke:** (2nd row) The main differences are due to minor inaccuracies in reflected directions (i.e., shifted or missing highlights).
7. **Dinner Scene:** (3rd row) Again, the main differences are at high frequency edges in the rendering, as well as slightly more blurred shadows.
8. **Cube Pile Scene:** (4th row) Similar as in prior scenes; the main differences are at high frequency edges.

C Comparisons with RenderFormer

RenderFormer-V2 shares some architectural similarities with RenderFormer [4]. To better assess the differences and improvements, we perform an in-depth comparison.

C.1 Architecture

Figure 3 contrasts the RenderFormer architecture with RenderFormer-V2’s architecture. The key differences are:

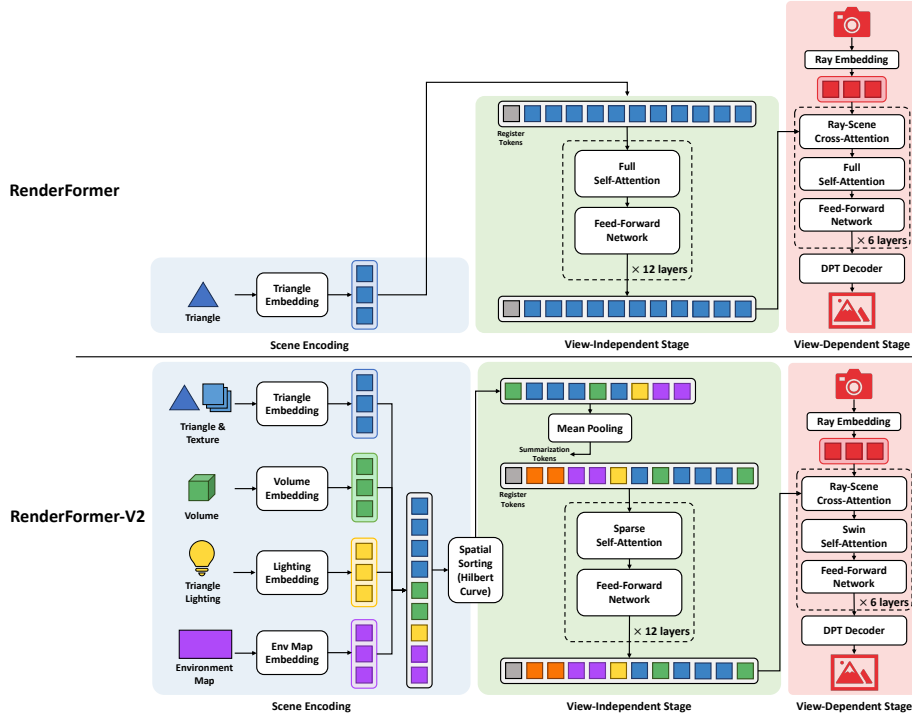


Fig. 3: RenderFormer-V2 and RenderFormer Model Architecture.

1. **Input Tokens.** RenderFormer encodes all primitives (including light sources) in a homogeneous triangle token. RenderFormer-V2 supports heterogeneous tokens, each with its own encoding procedure, supporting a wide range of primitives ranging from triangles, volume elements, and environment maps. In addition, RenderFormer-V2 adds support for spatially varying materials (with displacement mapping) and uses a BRDF-model-agnostic material specification.
2. **View-independent Attention.** Whereas RenderFormer utilizes a full (dense) self-attention, RenderFormer-V2 uses a sliding windowed attention combined with attention sinks. The sinks have rendering-aware semantics and include: global registers, light sources, and summarization tokens to model long-range light transport.
3. **View-dependent Attention.** RenderFormer uses full self-attention between all ray-bundle tokens. This is costly and it is unlikely that pixels far away in the image will be meaningfully related to nearby pixels. Therefore, RenderFormer-V2 uses SWIN attention instead.

By extending the supported token types and by adopting a novel sparse attention architecture, RenderFormer-V2 enables more accurate rendering of more complex scenes with richer visual effects.

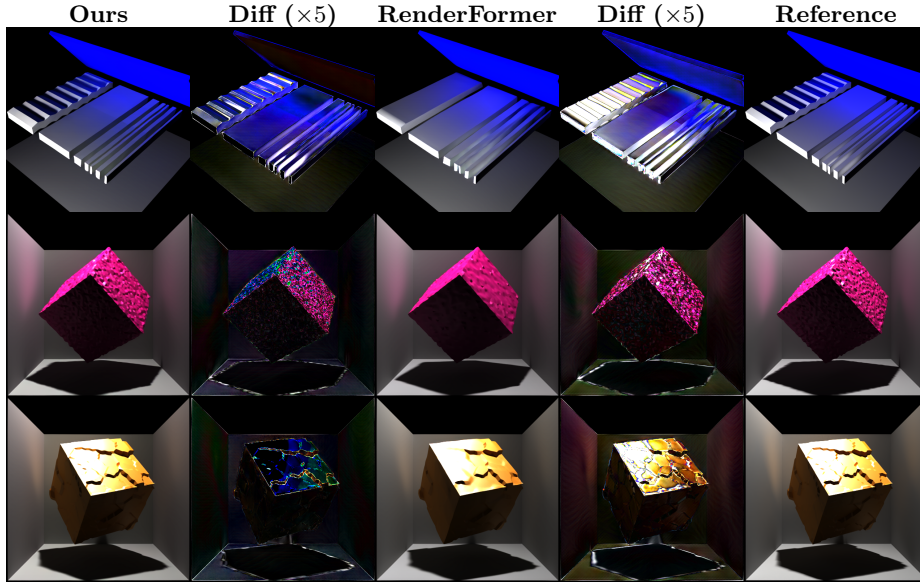


Fig. 4: Visual comparison of RenderFormer-V2 and RenderFormer against a path-traced reference. Difference maps (scaled by $\times 5$) are shown to highlight rendering errors.

Table 2: Comparison of high-resolution render accuracy between RenderFormer and RenderFormer-V2 (with and without high-resolution fine-tuning). The metrics are computed on renderings at 2048×2048 resolution.

Model	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FLIP $\downarrow$
RenderFormer	26.2423	0.9473	0.0886	0.1332
RenderFormer-V2	28.0810	0.9648	0.0503	0.1172
RenderFormer-V2 (finetuned)	30.7304	0.9742	0.0244	0.0952

C.2 Qualitative Comparison

Figure 4 qualitatively compares three scenes rendered with RenderFormer-V2 vs. RenderFormer vs. a path-tracer reference rendering (difference images are scaled $5\times$ to better highlight discrepancies). In all cases, we observe that RenderFormer-V2 more accurately renders fine geometrical details such as the corrugated structures and the narrow gaps between cuboids in the first row, as well as the high-frequency geometric patterns in the second and third rows. The difference images further show that RenderFormer-V2 yields smaller rendering errors overall.

C.3 Resolution Scalability

Thanks to the locality of the SWIN attention in the view-dependent stage, RenderFormer-V2 exhibits better scalability with respect to changes in render-

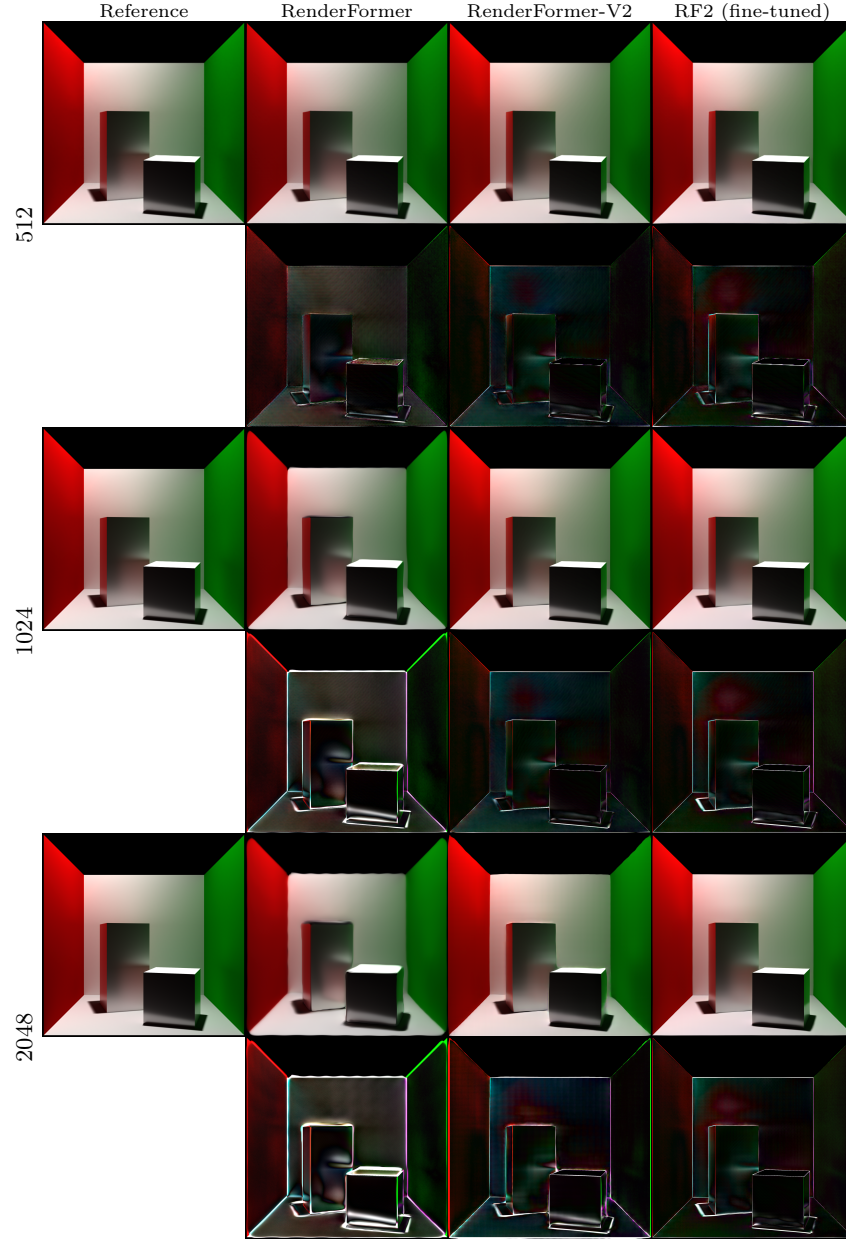


Fig. 5: Resolution scaling comparison of RenderFormer vs RenderFormer-V2 (with and without high-resolution fine-tuning).

Table 3: Comparison between RenderFormer-V2 (ours) and RenderFormer across different triangle counts. For each triangle count and each metric, the better result is highlighted in bold.

Tri Count	Model	PSNR $\uparrow$	SSIM $\uparrow$	LPIPS $\downarrow$	FLIP $\downarrow$
4K	RenderFormer-V2	33.84	0.9776	0.0198	0.1025
	RenderFormer	34.51	0.9823	0.0244	0.0838
8K	RenderFormer-V2	33.00	0.9682	0.0305	0.1230
	RenderFormer	32.89	0.9708	0.0358	0.1079
16K	RenderFormer-V2	29.96	0.9457	0.0502	0.1932
	RenderFormer	29.36	0.9418	0.0604	0.1555
32K	RenderFormer-V2	28.07	0.9205	0.0729	0.2387
	RenderFormer	24.17	0.8451	0.1500	0.3080
64K	RenderFormer-V2	26.79	0.9020	0.0873	0.2609
	RenderFormer	18.72	0.6651	0.3070	0.5313
128K	RenderFormer-V2	25.82	0.8847	0.0999	0.3220
	RenderFormer	15.88	0.4604	0.4772	0.7466

resolution. We validate this quantitatively (Table 2) and qualitatively (Figure 5) by comparing the accuracy of RenderFormer vs. RenderFormer-V2 vs. a resolution fine-tune of RenderFormer-V2. RenderFormer shows noticeably higher errors when rendering at higher resolutions (1024 and 2048). Moreover, fine-tuning RenderFormer-V2 at 2048 resolution further improves RenderFormer-V2’s ability to recover fine details under high-resolution settings.

C.4 Full Detailed Metrics

Finally, for completeness, in Table 3 we show the full rendering quality comparison with RenderFormer which was summarized in the main submission in the error plot (Fig. 6).

D Additional Results

Figure 6 shows 8 scenes used by Zeng et al. [4] to demonstrate the capabilities of RenderFormer, which RenderFormer-V2 can also handle without issues.

Finally, we show additional complex room scenes in Figure 7 with fine geometrical details, complex light transport, and textures.

We refer to the supplementary video for additional results demonstrating RenderFormer-V2’s capabilities as well as its stability to changes in scene and camera parameters.

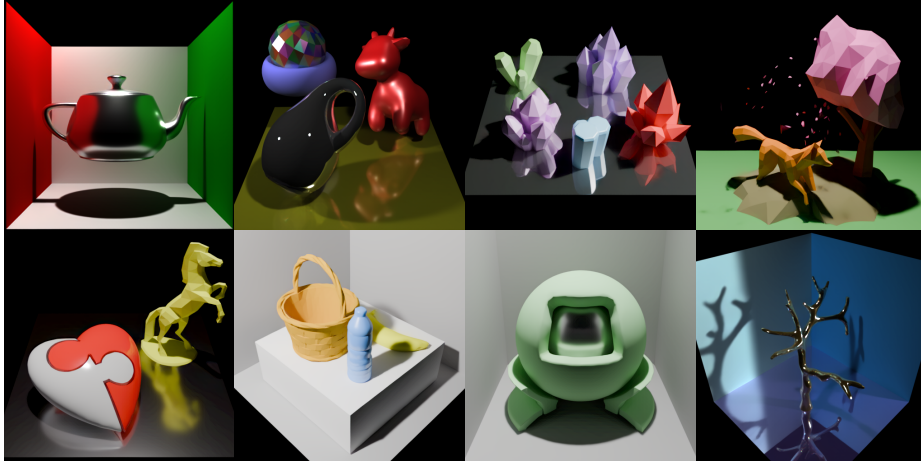


Fig. 6: The 8 test scenes from RenderFormer [4] rendered with RenderFormer-V2.



Fig. 7: Additional room scenes rendered with RenderFormer-V2.

E Ablation Test Scenes

Figure 8 shows example scenes from our test set used for the ablation studies. These procedurally generated scenes contain diverse textures, a variety of test objects, large scene token counts, as well as volumetric effects, area lighting, and environment lighting.

F Limitations

Figure 9 illustrates the limitations of RenderFormer-V2 in rendering high-resolution textures on large triangles. Since each triangle is associated with a fixed-resolution texture embedding (32×32), the effective texture resolution becomes low when the triangle covers a large surface area. By subdividing the mesh to increase the number of triangles and thus reduce triangle size, the texture quality can be significantly improved.

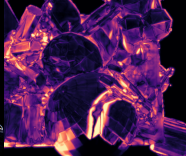
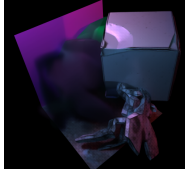
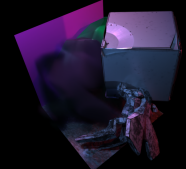
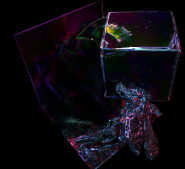
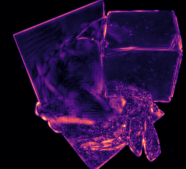
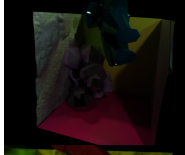
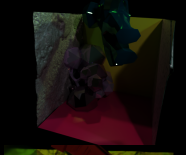
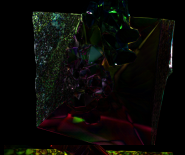
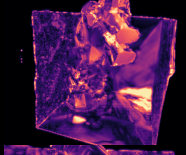
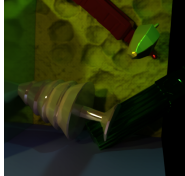
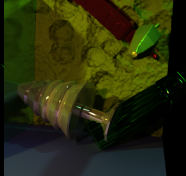
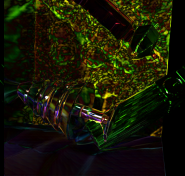
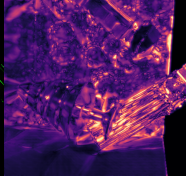
Pred	Reference	Diff ($\times 5$)	FLIP	Metrics
				PSNR: 29.07 SSIM: 0.9285 LPIPS: 0.04980 FLIP: 0.2044
				PSNR: 30.92 SSIM: 0.9353 LPIPS: 0.04512 FLIP: 0.1125
				PSNR: 34.99 SSIM: 0.9506 LPIPS: 0.06065 FLIP: 0.1719
				PSNR: 28.92 SSIM: 0.8827 LPIPS: 0.1240 FLIP: 0.2515

Fig. 8: A variety of procedurally generated ablation scenes rendered with RenderFormer-V2 and compared to path-traced reference images. We also report PSNR, SSIM, LPIPS, and FLIP errors.

References

1. Andersson, P., Nilsson, J., Akenine-Möller, T., Oskarsson, M., Åström, K., Fairchild, M.D.: FLIP: A Difference Evaluator for Alternating Images. Proceedings of the ACM on Computer Graphics and Interactive Techniques **3**(2), 15:1–15:23 (2020)
2. Dong, J., FENG, B., Guessous, D., Liang, Y., He, H.: Flexattention: A programming model for generating fused attention variants. In: Eighth Conference on Machine Learning and Systems (2025), <https://openreview.net/forum?id=2QMYV4bA0R>
3. Henry, A., Dachapally, P.R., Pawar, S., Chen, Y.: Query-key normalization for transformers (2020), <https://arxiv.org/abs/2010.04245>
4. Zeng, C., Dong, Y., Peers, P., Wu, H., Tong, X.: Renderformer: Transformer-based neural rendering of triangle meshes with global illumination. In: Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers. SIGGRAPH Conference Papers '25 (2025). <https://doi.org/10.1145/3721238.3730595>, <https://doi.org/10.1145/3721238.3730595>

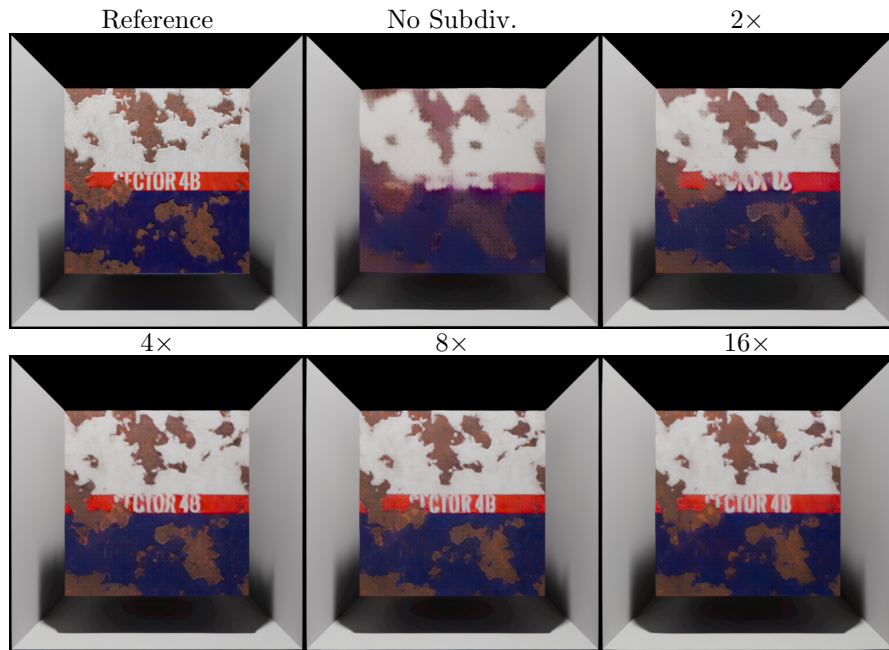


Fig.9: Limitations: large textured triangle can result in loss of texture sharpness. Increasing triangle subdivision level improves texture quality, with finer details and fewer texture artifacts at higher subdivision levels.