MeshGS: Adaptive Mesh-Aligned Gaussian Splatting for High-Quality Rendering Supplementary Material

Jaehoon Choi[1], Yonghan Lee[1], Hyungtae Lee[3],
Heesung Kwon[2], and Dinesh Manocha[1]

[1] University of Maryland, College Park, USA
[2] DEVCOM Army Research Laboratory, Adelphi, USA
[3] BlueHalo, Rockville, USA
{kevchoi,lyhan12}@umd.edu, hyungtae.lee@bluehalo.com,
heesung.kwon.civ@army.mil, dm@cs.umd.edu

1 Implementation Details

We utilize the geometry reconstruction and mesh processing part from the LTM [1]. Leveraging the strong smoothness regularization [2], this method shows consistently smooth surfaces on both ground and wall structures. Additionally, due to the incorporation of contraction function [7], background regions are effectively reconstructed. However, we do not utilize the optimization step, resulting in a mesh surface that lacks detailed structures and often appears smoothed due to the limited number of vertex. Our method is implemented using PyTorch [6] and builds on Gaussian rasterization provided by the original 3DGS [5]. For scene composition, we initially obtain the textured human mesh from Mixamo and export the compositional mesh result from the Blender engine. Subsequently, we employ our rasterization code to render the compositional scene using the exported mesh composition.

2 Comparison to 2D Gaussian Splatting

Recently, a 2DGS [4] was released, showing remarkable performance in mesh reconstruction and novel view synthesis. In Table 1 and Table 2, we compare our method with 2D GS in terms of rendering performance. Unlike approaches that bind 3D splats to an underlying mesh, this research prioritizes aligning 2D Gaussian splats with the underlying geometry without requiring prior 3D geometry. This paper demonstrate that 2D Gaussian splats surpass 3D Gaussian splats in representing geometry. Our Gaussian splats binding method can leverage the 2D Gaussian splats representation to bind splats to a given mesh.

3 Additional Qualitative Comparisons

Supplementary material includes all rendering results on the test set, provided as a downloadable zip file. To accommodate file size limitations (100mb for supplementray material), we downscale the resolution of rendering images. In the

Method	Mesh	Rendering	Outdoor (PSNR ↑ / SSIM ↑ / LPIPS ↓) Bicycle	Garden	Stump	Mean
2DGS [4]	O	Splat	24.6 / 0.71 / 0.30	26.6 / 0.83 / 0.16	26.0 / 0.76 / 0.29	25.7 / 0.76 / 0.25
MeshGS (Ours)	O	Splat	24.4 / 0.71 / 0.26	26.8 / 0.85 / 0.12	25.8 / 0.74 / 0.25	25.7 / 0.77 / 0.21
MeshGS* (Ours)	O	Splat	**24.9** / **0.75** / **0.25**	**26.8** / **0.86** / **0.12**	**26.3** / **0.76** / **0.22**	**26.0** / **0.79** / **0.20**

Table 1: Quantitative Comparison on outdoor mip-NeRF 360 Dataset. MeshGS* did not utilize any regularization techniques to tightly align Gaussian splats; instead, it solely applied image loss for training.

Method	Mesh	Rendering	Indoor (PSNR ↑ / SSIM ↑ / LPIPS ↓) Room	Counter	Kitchen	Bonsai	Mean
2DGS [4]	O	Splat	30.8 / 0.91 / **0.22**	**28.1** / 0.89 / 0.22	**30.1** / **0.92** / **0.14**	**31.3** / 0.93 / 0.25	**30.1** / 0.91 / **0.19**
MeshGS (Ours)	O	Splat	30.3 / 0.90 / 0.24	28.0 / 0.89 / **0.21**	29.7 / 0.90 / 0.17	30.7 / **0.93** / 0.20	29.6 / **0.91** / 0.20
MeshGS* (Ours)	O	Splat	**30.9** / **0.93** / 0.23	27.8 / **0.90** / 0.21	29.3 / 0.90 / 0.17	30.5 / 0.93 / **0.19**	29.6 / 0.91 / 0.20

Table 2: Quantitative Comparison on indoor mip-NeRF 360 Dataset. MeshGS* did not utilize any regularization techniques to tightly align Gaussian splats; instead, it solely applied image loss for training.

manuscript, we show the mesh reconstruction results for Garden, Stump, and Bicycle scenes (refer to Fig. 2, Fig.4, and Fig. 6) due to the limited space. we visualize the results for the remaining scenes on both Mip-NeRF 360 dataset and Deep Blending dataset. We present both the mesh reconstruction and rendering outcomes for these scenes, depicted in Fig. 1 and Fig. 2, respectively. Additionally, we show visualizations for the mesh reconstruction and rendering results of the Playroom and Dr. Johnson scenes from the Deep Blending dataset, showcased in Fig. 3 and Fig. 4, respectively.

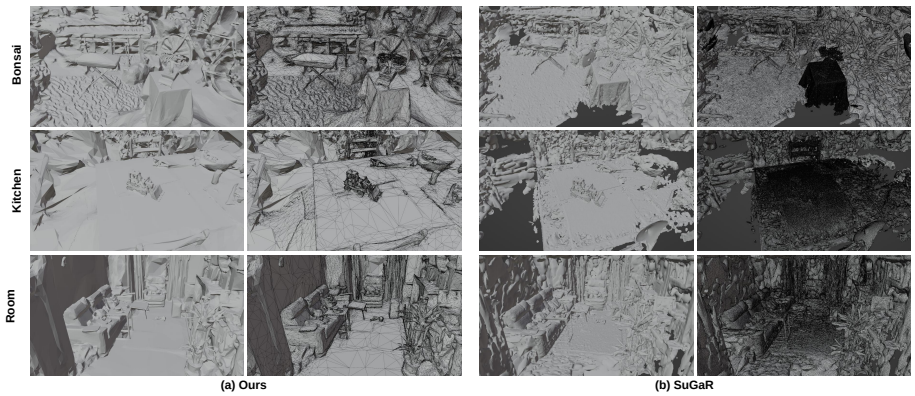

Fig. 1: Qualitative comparison between (a) Ours and (b) SuGaR [3]. Our method and SuGaR show shading mesh and wireframe mesh extracted without texture, respectively. In SuGaR, the dark grey regions denote empty areas where the mesh fails to represent the scene. Compared to ours, SuGaR's mesh inadequately captures background regions

Fig. 2: Qualitative Comparisons with Existing Methods. We visually compare our method with SuGaR [3], 3DGS [5], and LTM [1]. Red arrows emphasize subtle differences in rendering quality

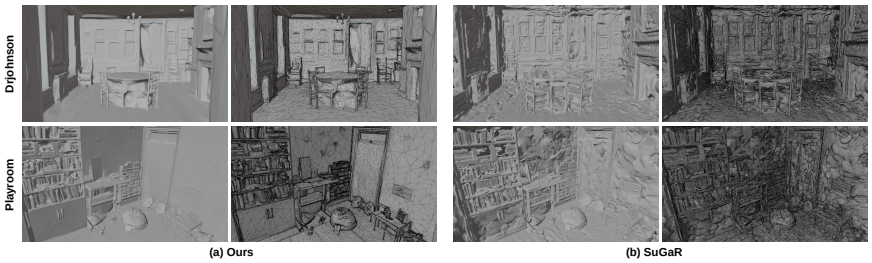

Fig. 3: Qualitative comparison between (a) Ours and (b) SuGaR [3]. Our method and SuGaR show shading mesh and wireframe mesh extracted without texture, respectively.

Fig. 4: Qualitative Comparisons with Existing Methods. We visually compare our method with SuGaR [3], 3DGS [5], and LTM [1]. Red arrows emphasize subtle differences in rendering quality.

References

1. Choi, J., Shah, R., Li, Q., Wang, Y., Saraf, A., Kim, C., Huang, J.B., Manocha, D., Alsisan, S., Kopf, J.: Ltm: Lightweight textured mesh extraction and refinement of large unbounded scenes for efficient storage and real-time rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5053–5063 (2024)
2. Gropp, A., Yariv, L., Haim, N., Atzmon, M., Lipman, Y.: Implicit geometric regularization for learning shapes. In: Proceedings of the 37th International Conference on Machine Learning. ICML'20, JMLR.org (2020)
3. Guédon, A., Lepetit, V.: Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5354–5363 (2024)
4. Huang, B., Yu, Z., Chen, A., Geiger, A., Gao, S.: 2d gaussian splatting for geometrically accurate radiance fields. arXiv preprint arXiv:2403.17888 (2024)
5. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics **42**(4) (2023)
6. Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al.: Pytorch: An imperative style, high-performance deep learning library. Advances in neural information processing systems **32** (2019)
7. Yariv, L., Hedman, P., Reiser, C., Verbin, D., Srinivasan, P.P., Szeliski, R., Barron, J.T., Mildenhall, B.: Bakedsdf: Meshing neural sdfs for real-time view synthesis. arXiv (2023)