

3DSceneEditor: Controllable 3D Scene Editing with Gaussian Splatting

Supplementary Material

1. Implementation details

1.1. Implementation Details of 3D Scene Representation

3DSceneEditor processes 3D scenes reconstructed using 3D Gaussian Splatting [3]. Since ScanNet++ [6] does not provide an initial point cloud derived from Structure-from-Motion (SfM)—a crucial requirement for achieving high-quality results with 3D Gaussian Splatting—we sampled 1 million points from the ground-truth (GT) mesh as the initial point cloud [2] and fix the number of the Gaussians during training to prevent many Gaussians merge together. This ensures the geometry of the Gaussian sets are well-defined (shown in Fig. 1), which is critical for subsequent instance segmentation tasks.

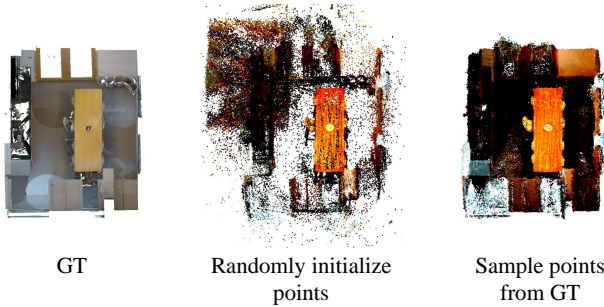


Figure 1. **Visualization of geometric results for Gaussian sets with varying initializations.** The figure illustrates the results of using randomly initialized points versus points sampled from the ground truth (GT). The Gaussian set trained with random initialization shows significant noise in the 3D geometry, making it unsuitable for instance segmentation tasks. In contrast, the Gaussian set sampled from the GT and then trained achieves much higher geometric accuracy with minimal noise, closely aligning with the GT.

Following the default configuration of 3D Gaussian Splatting, each scene is trained for 30,000 iterations, with input images exceeding 1600 pixels in width being automatically resized to 1600 pixels for computational efficiency.

For baseline methods, which utilize Instruct-Pix2Pix [1] for scene editing, input images are resized to 512×512 pixels as required, while all other hyperparameters are kept consistent.

1.2. Implementation Details of Object Grounding

Instance segmentation. In our experiments, we set the default confidence threshold $c = 0.8$ for instance segmentation to achieve higher segmentation precision. However, to avoid excluding small objects (e.g., paper, cups, books), we

lower the threshold to $c = 0.3$ when targeting such challenging objects for the pre-trained model, even if this results in additional noise or slight mis-segmentation.

In Fig. 2, when the confidence threshold c is set to 0.8, each instance is segmented more completely. However, objects with a confidence below c are filtered out (highlighted by green bounding boxes). Conversely, at $c = 0.3$, more objects are successfully segmented, but additional noise appears in some instances and on the floor (highlighted by red bounding boxes).

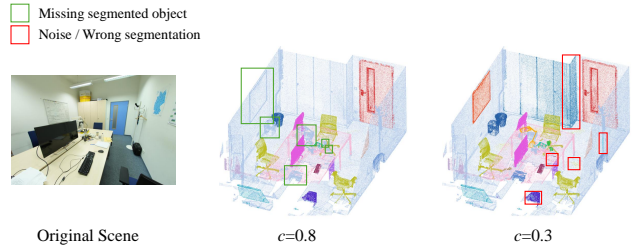


Figure 2. **Visualization of instance segmentation with different confidence threshold.** In this figure, we visualize instance segmentation of a 3D scene represented by a Gaussian set using colorful point clouds as Gaussians lack RGB attributes. Green bounding boxes highlight the missed segmented objects when the confidence threshold is set to $c = 0.8$ compared to $c = 0.3$, while red bounding boxes indicate additional noise or mis-segmentations introduced at the lower threshold.

1.3. Implementation Details of 3D Gaussians Editing

Object recoloring. We designed a color-mapping table (refer to “color-mapping.pdf”) to translate color keywords from prompts into their corresponding RGB values. This pipeline enables editing with over 200 distinct colors, ensuring precise and flexible color adjustments.

Object addition and replacement. Our pipeline leverages LGM [5], one of the fastest Gaussian-based generative backbone, for creating new objects. To ensure compatibility with 3D Gaussian Splatting, we set the spherical harmonics degree $sh = 3$, while keeping the remaining parameters unchanged. The pipeline supports inputs in the form of text-only, image-only or text + image combinations for the generative model, with each generation trained for 30 iterations.

Object relighting. To align the illumination of new objects with the scene, we first extract an HDR environment map from a scene image using DiffusionLight [4]. We then use Blender to relight the multi-view images generated by LGM [5], which are subsequently used to generate the fi-



Figure 3. More visualization result of 3DSceneEditor.

nal objects. Notably, object relighting is optional in our editing pipeline, as extracting the HDR map typically takes around 15-20 minutes, which would disrupt our interactive-rate editing workflow. The visualization of object relighting is shown in Fig. 4.

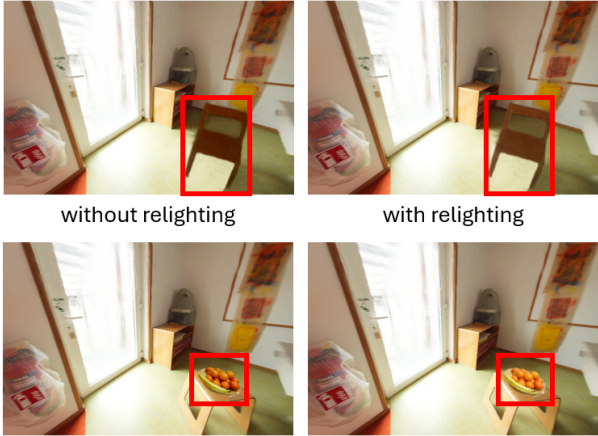


Figure 4. Results of object relighting

Detail of Spatial relation interpreting. Here we provide more detail about our spatial relation interpreting module. We assume the virtual camera is positioned at the geometric center of the scene and infer view-dependent relationships by projecting both the target and reference objects

onto a 2D plane along the camera ray direction. As illustrated in Fig. 5 (Prompt: Remove the trash can on the right side of the refrigerator), the trash can within the red bbox is selected because it lies to the right of the refrigerator (green bbox) relative to the camera ray direction.

Prompt: Remove the trash can on the right side of the refrigerator

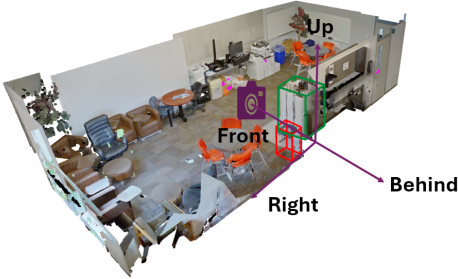


Figure 5. Visualization of spatial relation.

Adaptive optimization framework. We provide more visualization of our adaptive optimization module in Fig.6. With the help of our optimization framework, we can generate results with higher human aesthetic scores.

User study design. We conducted a user study with 42 professionals working in 3D field to assess the quality of edits, as detailed in Fig. 7. The study includes 20 questions covering 10 different editing operations. For each operation assessment, it combined with an original video, four edited

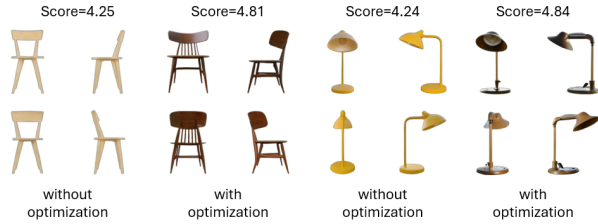


Figure 6. More visualization of adaptive optimization

versions generated by the evaluated methods, and a corresponding target editing instruction. Participants were asked to answer two key questions:

- 1) **Prompt Alignment** – Evaluating how well the edits match the given prompt.
- 2) **Editing Quality** – Selecting the video with the best quality based on their first choice.

If none of the edited scenes align with the prompt, participants can select *None of the above* to avoid a forced choice. Note that responses marked as *None of the above* will be excluded from the Editing Quality evaluation, as this question is skipped in such cases.

References

- [1] Tim Brooks, Aleksander Holynski, and Alexei A Efros. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402, 2023. 1
- [2] Yalda Foroutan, Daniel Rebain, Kwang Moo Yi, and Andrea Tagliasacchi. Does gaussian splatting need sfm initialization? *arXiv preprint arXiv:2404.12547*, 2024. 1
- [3] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023. 1
- [4] Pakkapon Phongthawee, Worameth Chinchuthakun, Nontaphat Sinsunthithet, Varun Jampani, Amit Raj, Pramook Khungurn, and Supasorn Suwajanakorn. Diffusionlight: Light probes for free by painting a chrome ball. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 98–108, 2024. 1
- [5] Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pages 1–18. Springer, 2024. 1
- [6] Chandan Yeshwanth, Yueh-Cheng Liu, Matthias Nießner, and Angela Dai. Scannet++: A high-fidelity dataset of 3d indoor scenes. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12–22, 2023. 1

3D Editing Quality Assessment

Instructions - MUST READ

A 3D editing pipeline is tasked to edit a scene provided in the first video of each questionnaire, with a *text prompt*, e.g. "Change the color of a pillow on the on the bed to DarkOrchid".

We want to evaluate the quality of the edited scenes. You will be asked to assess it from two perspectives: **prompt alignment** and **editing quality**.

- **Prompt Alignment** refers to whether the content of the results accurately reflects the instructed text prompts.

- **Editing Quality** is determined by various details in the post-editing scenes, including whether all target objects have been successfully removed, whether added or replaced objects are accurately positioned and visually realistic, and whether the video maintains clarity without local artifacts.

On each page, you will first see the initial scene (by video) and the text prompt. Then you will be asked to evaluate four edit results (by video) generated by four 3D editing methods.

Note: Please refresh if the video is not loaded.

Example of the Questionnaire

Given the initial scene shown in the following video:

and the text prompt:

"Remove the monitors on the desk".



Please watch the following four videos and assess the prompt alignment and editing quality.



Please read carefully:

1. Answer the first question, then choose one option for the second question based on your selection from the first question.
2. If none of the videos align with the prompt, select "*None of the above*" in the first question and "*N/A*" in the second question.

| | 1 | 2 | 3 | 4 | None of the above |
|---|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| Which edited results align with the prompt? (could be multiple selection) | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| Which edited result has the best editing quality | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> | <input type="radio"/> |

Get started

Figure 7. Design of our user study.