

GSFix3D: Diffusion-Guided Repair of Novel Views in Gaussian Splatting

Supplementary Material

6. Data Preparation

6.1. Novel View Selection for Replica

The Replica dataset [29] contains high-quality reconstructions of diverse indoor scenes, featuring clean dense geometry and high-resolution textures. We leverage the provided 3D models and the official Replica SDK to render novel view images, which serve as ground truth for quantitative evaluation in Sec. 4.2. We adopt the same camera intrinsics and image resolution (1200×680) as the training views provided by [43] when generating novel views. Since these trajectories do not cover the entire scene, certain areas in the reconstructed map remain unobserved or under-constrained, often exhibiting artifacts. To assess inpainting performance, we deliberately select novel views that include such missing or artifact-prone regions, where existing pipelines (e.g., SplaTAM, RTG-SLAM, GSFusion) typically fail. This results in an intentionally challenging novel view repair task. Examples of selected novel views from the Replica dataset are shown in Fig. 3, Fig. 4, Fig. 6, Fig. 7, Fig. 8, Fig. 9 and Fig. 10. We will release the curated Replica novel view dataset to support reproducible research.

6.2. Real-World Data Collection

We collect a stereo image sequence inside a ship’s ballast water tank using an Intel RealSense D455 camera. The dataset contains 1,068 grayscale images at a resolution of 640×480 for both the left and right stereo cameras. Stereo images and recorded IMU data are fed into OKVIS2 [12] to estimate camera poses for each left stereo image, though the estimated poses may contain errors. Leveraging recent advancements in depth estimation, we employ Foundation-Stereo [34] to generate smoother and higher-quality depth maps (corresponding to the left stereo images) from the stereo pairs. The post-processed data, including left stereo images, depth maps, and camera poses, are then used as input to GSFusion [33] for the initial 3DGS reconstruction. Since no ground truth is available for this self-collected dataset, we randomly select five novel views in the reconstructed scene where shadow-like floaters caused by pose inaccuracies are prominent. For each, we use a nearby captured training view as a reference “ground truth” to qualitatively assess our method’s performance. Examples of the captured data and selected novel views are shown in Fig. 5 and Fig. 11. This real-world dataset will be released to support reproducible research.

Method	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
SplaTAM	23.03	0.791	0.311
SplaTAM + DIFIX	23.06	0.789	0.220
SplaTAM + DIFIX-finetune †	<u>24.67</u>	<u>0.829</u>	0.133
SplaTAM + GSFixer	25.11	0.831	<u>0.188</u>
RTG-SLAM	19.54	0.777	0.341
RTG-SLAM + DIFIX	19.43	0.762	0.245
RTG-SLAM + DIFIX-finetune †	<u>24.44</u>	<u>0.812</u>	0.156
RTG-SLAM + GSFixer	24.80	0.824	<u>0.204</u>
GSFusion (gs)	24.58	0.838	0.308
GSFusion (gs) + DIFIX	24.34	0.818	0.193
GSFusion (gs) + DIFIX-finetune †	<u>24.87</u>	0.834	0.142
GSFusion (gs) + GSFixer	24.79	0.833	0.196
GSFusion (mesh+gs) + GSFixer	25.30	<u>0.837</u>	<u>0.183</u>

Table 5. More comparisons of diffusion-based repair methods on the ScanNet++ dataset. † indicates that we fine-tuned this model based on the original DIFIX model. The best result is highlighted in **bold**, and the second-best is underlined. The text inside () indicates the format of the reconstruction used.

Method	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
SplaTAM	23.82	0.833	0.267
SplaTAM + DIFIX	22.97	0.790	0.262
SplaTAM + DIFIX-finetune †	<u>25.45</u>	0.867	0.149
SplaTAM + GSFixer	25.67	<u>0.839</u>	<u>0.215</u>
RTG-SLAM	25.00	0.860	0.247
RTG-SLAM + DIFIX	24.02	0.811	0.214
RTG-SLAM + DIFIX-finetune †	<u>25.47</u>	<u>0.858</u>	0.156
RTG-SLAM + GSFixer	26.27	0.843	0.228
GSFusion (gs)	22.10	0.844	0.296
GSFusion (gs) + DIFIX	21.81	0.772	0.273
GSFusion (gs) + DIFIX-finetune †	23.12	0.847	0.185
GSFusion (gs) + GSFixer	<u>23.87</u>	0.830	0.251
GSFusion (mesh+gs) + GSFixer	25.98	<u>0.845</u>	<u>0.219</u>

Table 6. More comparisons of diffusion-based repair methods on the Replcia dataset. † indicates that we fine-tuned this model based on the original DIFIX model. The best result is highlighted in **bold**, and the second-best is underlined. The text inside () indicates the format of the reconstruction used.

7. Additional Results

7.1. More DIFIX Variants

DIFIX and DIFIX-ref [35] are diffusion models pretrained on 80k noisy-clean real image pairs created using their proposed dataset curation strategies, whereas our GSFixer is only pretrained on two synthetic datasets with randomly added Gaussian noise and blur, followed by fine-tuning on a small amount of clean captured data. To demonstrate the effectiveness and efficiency of our training strategy, we also

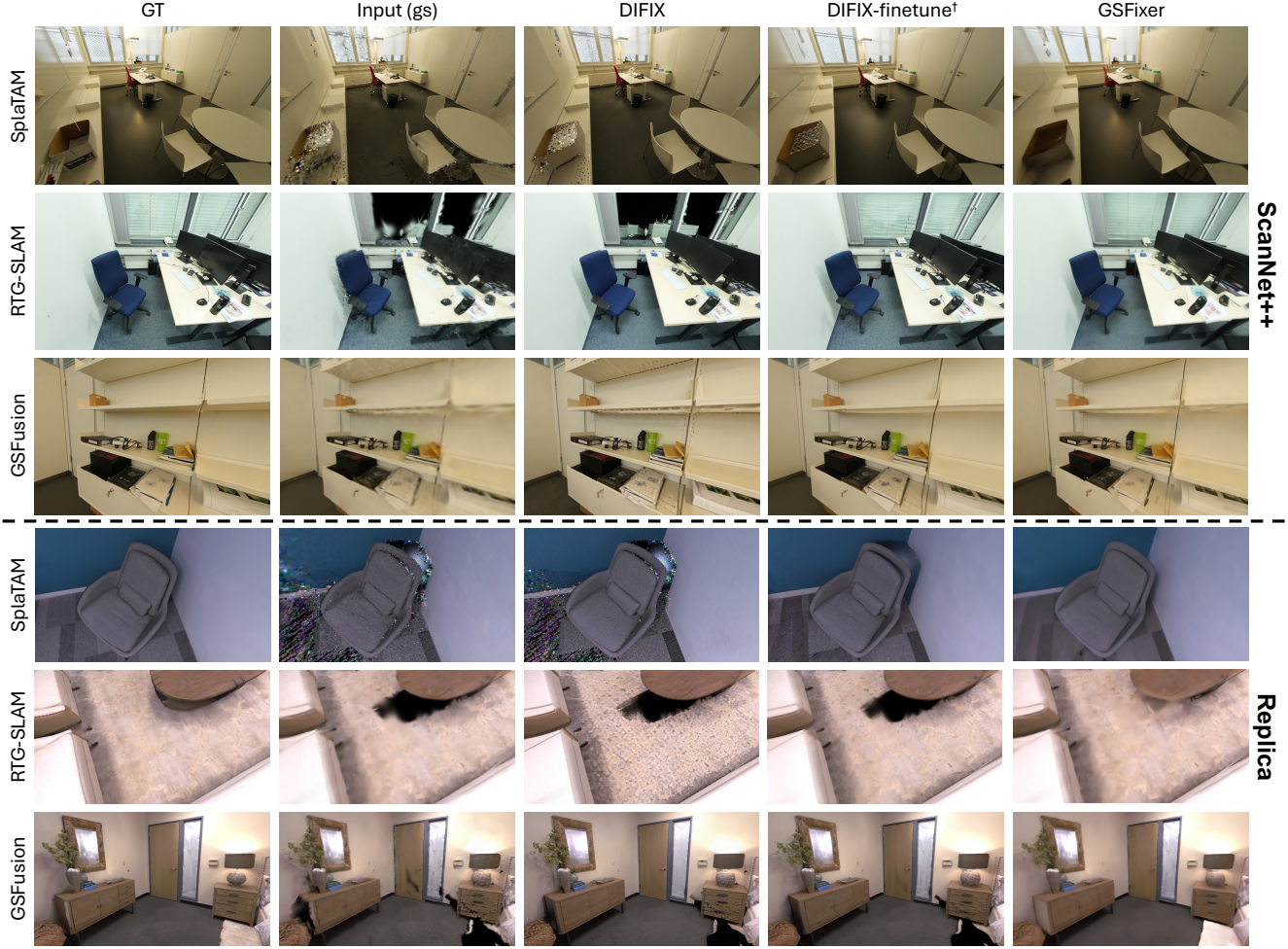


Figure 8. More qualitative comparisons of diffusion-based repair methods on the ScanNet++ and Replica datasets. All examples use only 3DGS reconstructions as the input source. Zoom in to better observe how GSFixer effectively removes artifacts and fills in large holes, where both DIFIX and DIFIX-finetune[†] fail to produce satisfactory results.

fine-tune the original DIFIX model on the same scene data for 800 iterations on the ScanNet++ dataset and 400 iterations on the Replica dataset.

Due to the higher GPU memory demands of training DIFIX, we used an NVIDIA A40 GPU with 48GB VRAM to get DIFIX-finetune, while our GSFixer is trained on a 24GB NVIDIA RTX 4500 Ada GPU. Results are presented in Tab. 5 and Tab. 6. As expected, DIFIX-finetune shows improved performance over the original DIFIX on both datasets. However, it still falls short of GSFixer, particularly in PSNR, which correlates with lower visual quality and is clearly visible in Fig. 8. For instance, in the ScanNet++ dataset, the colorful floaters in the SplatAM example are slightly suppressed by DIFIX-finetune, but not fully removed (also seen in the GSFusion example). In the RTG-SLAM example, DIFIX-finetune fills in the missing window area with content, but the result lacks the tex-

ture consistency achieved by GSFixer. Similarly, in the Replica dataset, although DIFIX-finetune reduces some artifacts compared to the original DIFIX, it still fails to inpaint large visible holes, which is an essential capability for novel view repair.

In conclusion, our fine-tuning protocol not only delivers superior performance in challenging scenarios but also requires significantly fewer computational resources and minimal dataset curation, making it both effective and efficient.

7.2. More Ablation Studies on Image Conditions

Although SplatAM and RTG-SLAM do not produce meshes, we reuse the mesh extracted from GSFusion to render conditional images for fine-tuning and inference of GSFixer. As shown in Tab. 7, GSFixer conditioned on dual-input consistently outperforms the single-input variant (conditioned only on 3DGS) across both datasets for

Dataset	Method	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
ScanNet++	SplaTAM	23.03	0.791	0.311
	SplaTAM (gs) + GSFixer	25.11	0.831	0.188
	SplaTAM (mesh*+gs) + GSFixer	25.12	0.832	0.185
	RTG-SLAM	19.54	0.777	0.341
	RTG-SLAM (gs) + GSFixer	24.80	0.824	0.204
	RTG-SLAM (mesh*+gs) + GSFixer	25.05	0.827	0.191
Replica	SplaTAM	23.82	0.833	0.267
	SplaTAM (gs) + GSFixer	25.67	0.839	0.215
	SplaTAM (mesh*+gs) + GSFixer	26.49	0.845	0.198
	RTG-SLAM	25.00	0.860	0.247
	RTG-SLAM (gs) + GSFixer	26.27	0.843	0.228
	RTG-SLAM (mesh*+gs) + GSFixer	26.53	0.848	0.212

Table 7. More ablations of input image conditions on the ScanNet++ and Replica datasets. * denotes that the mesh reconstruction used for both SplaTAM and RTG-SLAM comparisons is borrowed from the GSFusion method.

Dataset	Method	PSNR \uparrow	SSIM \uparrow	LPIPS \downarrow
ScanNet++	SplaTAM	23.03	0.791	0.311
	SplaTAM (mesh*+gs) + GSFixer	25.12	0.832	0.185
	SplaTAM (mesh*+gs) + GSFix3D	25.21	0.836	0.218
	RTG-SLAM	19.54	0.777	0.341
	RTG-SLAM (mesh*+gs) + GSFixer	25.05	0.827	0.191
	RTG-SLAM (mesh*+gs) + GSFix3D	25.39	0.837	0.233
Replica	SplaTAM	23.82	0.833	0.267
	SplaTAM (mesh*+gs) + GSFixer	26.49	0.845	0.198
	SplaTAM (mesh*+gs) + GSFix3D	27.07	0.862	0.218
	RTG-SLAM	25.00	0.860	0.247
	RTG-SLAM (mesh*+gs) + GSFixer	26.53	0.848	0.212
	RTG-SLAM (mesh*+gs) + GSFix3D	27.18	0.868	0.236

Table 8. More comparisons of GSFixer and GSFix3D on the ScanNet++ and Replica datasets. * denotes that the mesh reconstruction used for both SplaTAM and RTG-SLAM comparisons is borrowed from the GSFusion method.

SplaTAM and RTG-SLAM. These results reinforce the trends observed in Sec. 4.4. For instance, in the ScanNet++ dataset (see Fig. 9), floaters persist in the SplaTAM example and missing thin structures in the RTG-SLAM example are successfully corrected when mesh input is included. Similarly, in the Replica dataset (see Fig. 9), carpet textures in the SplaTAM example and the shape of the vase in the RTG-SLAM example are better preserved with the help of complementary information from the mesh input, which is otherwise lost in the 3DGS-only setting.

7.3. More GSFix3D Comparisons

To further demonstrate the flexibility of the GSFix3D framework, we incorporate reconstructions from SplaTAM and RTG-SLAM into our pipeline for novel view repair in 3D space. We reuse the extracted mesh from the GSFusion method to enable the dual-input setting, allowing us to fully leverage GSFixer’s potential. The results in Tab. 8 align with our main experiments on GSFusion presented in Sec. 4.2. The improvement in GSFix3D is attributed to the multi-view constraints applied during the optimization of 3D representations. Additional qualitative examples are provided in Fig. 10.

7.4. More Real-World Tests

7.4.1 Self-collected Ship Data

In Fig. 11, we present additional novel view repair results on our self-collected ship dataset. Since 3DGS is highly sensitive to pose inaccuracies, erroneous poses from multiple views can introduce shadow-like floaters in the scene, resulting in more severe artifacts in novel views. Despite this challenge, our methods, GSFixer and GSFix3D, successfully learn the artifact distribution from the captured training data through our proposed fine-tuning protocol, enabling effective removal in both the 2D image space and the 3D scene representation. This in-the-wild test further highlights the robustness of our approach.

7.4.2 Outdoor Scenario

To further demonstrate adaptability across scenes and reconstruction pipelines, we select a challenging real-world outdoor scene from the FAST-LIVO dataset [41], covering building exteriors and archway corridors. The data is captured with a hard-synchronized LiDAR and camera setup, and we reconstruct the 3DGS scene using a LiDAR-Inertial-Camera Gaussian Splatting SLAM system [11] (Gaussian-LIC). Note that pose errors may still occur, producing inaccurate maps and broken geometries. We fine-tune GSFixer on the captured RGB data and the 3DGS renderings from Gaussian-LIC. As shown in Fig. 12, our method manages to fill in visual holes on the brick ground, recover distant buildings and sky, and correct broken structures such as the arch wall and deep corridor, further validating GSFix3D’s robustness in challenging real-world scenarios.

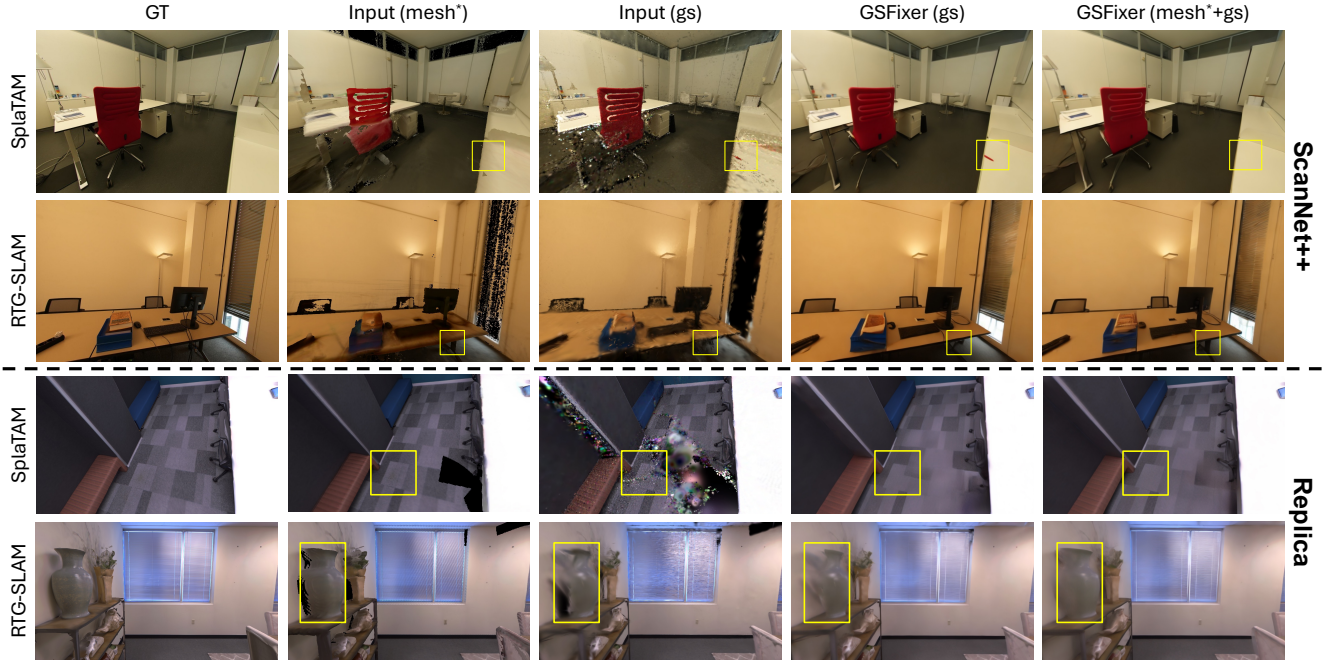


Figure 9. More qualitative ablations of input image conditions on the ScanNet++ and Replica datasets. The mesh reconstruction used for both SplatTAM and RTG-SLAM comparisons is borrowed from the GSFusion method. Zoom in to better observe how artifacts (highlighted by yellow boxes) present in the single-input settings are effectively mitigated with the dual-input configuration.

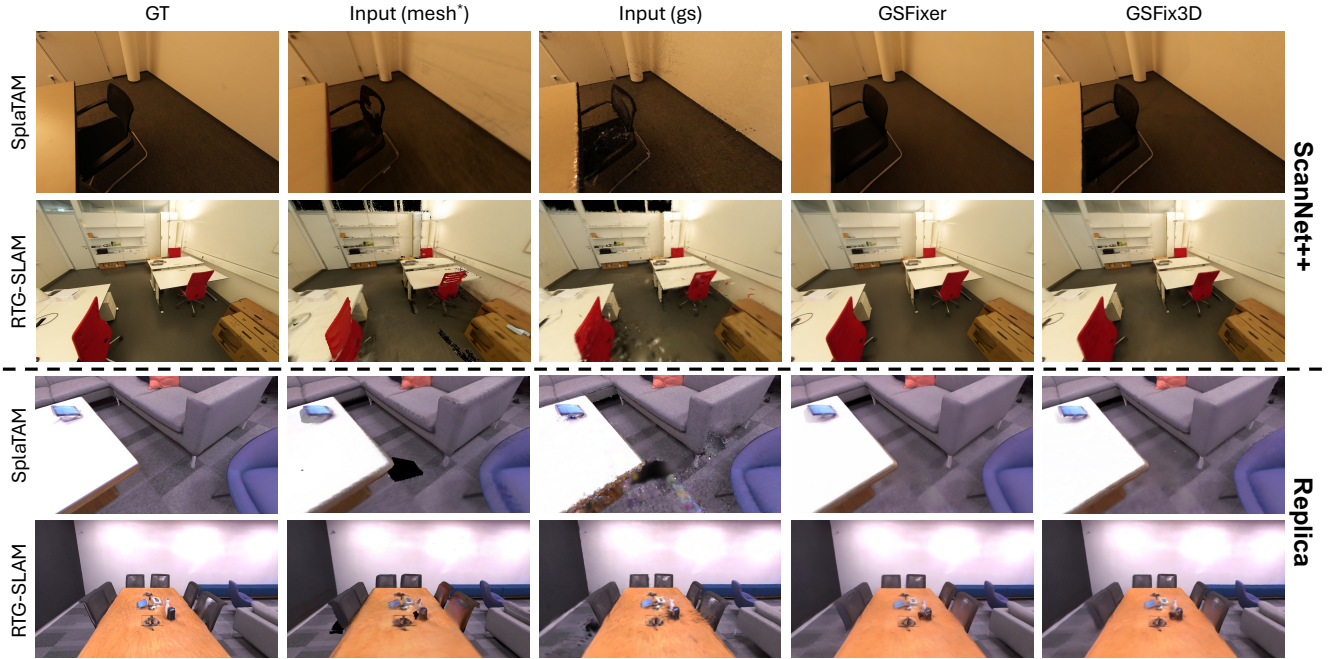


Figure 10. More qualitative comparisons between GSFixer and GSFix3D on the ScanNet++ and Replica dataset. Both mesh and 3DGS reconstructions are used as input sources. Zoom in to better observe how the 2D visual improvements from GSFixer are effectively distilled into the 3D space by GSFix3D.

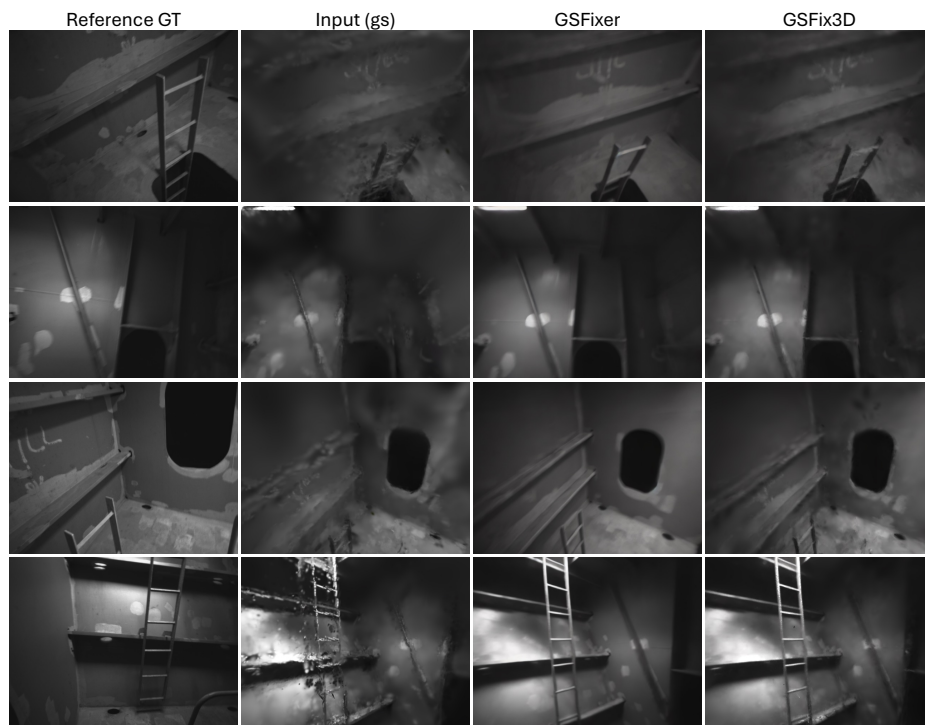


Figure 11. Novel view repair on self-collected ship data. Our method is robust to pose errors, effectively removing shadow-like floaters.



Figure 12. Novel view repair on a challenging outdoor scene from the FAST-LIVO dataset [41]. The initial reconstruction is generated by a LiDAR-Inertial-Camera Gaussian Splatting SLAM system [11], which may contain pose errors and produce inaccurate maps. Our method manages to repair those broken geometries to some extent.