S2D: Sparse to Dense Lifting for 3D Reconstruction with Minimal Inputs

Supplementary Material

Ablations	PSNR ↑	LPIPS ↓
$\mathcal{L}_{LPIPS} + \mathcal{L}_2 + \mathcal{L}_{GAN} + \mathcal{L}_{CLIP}$	20.1	0.35
$\mathcal{L}_{LPIPS} + \mathcal{L}_2 + \mathcal{L}_{GAN}$	20.1	0.35
$\mathcal{L}_{LPIPS} + \mathcal{L}_2 + \mathcal{L}_{GAN} + \mathcal{L}_{DINO}$	20.5	0.33
$\mathcal{L}_{LPIPS} + \mathcal{L}_2 + \mathcal{L}_{GAN} + \mathcal{L}_{SSIM}$	21.8	0.28
$\mathcal{L}_{LPIPS} + \mathcal{L}_2 + \mathcal{L}_{GAN} + \mathcal{L}_{SSIM} + \mathcal{L}_{DINO}$	22.0	0.27

Table 4. Ablation on loss terms.

Figure 7. Training data with different levels of artifacts (line 1) and example of random perturbations (line 2).

7. Ablation on loss terms

We further conduct ablations on loss terms applied on S2D artifact fixer by evaluating the image quality conditioned on different loss choices. Apart from base losses that are already verified [29], we compare the difference of CLIP, DINO and SSIM losses. The results are reported in Table 4.

While this is not new image generation task, the original CLIP loss actually contributes nothing in training, thus removing CLIP loss makes completely no difference. Introducing SSIM loss provides more improvements with higher supervision on structural details. DINO loss also helps with small enhancement by aligning semantic features.

We also test applying different weights to each loss term, and the results remain unchanged across reasonable variation within the range of 0.4.

8. Training data

To enhance the robustness of our model, we generate an extensive dataset that covers the broadest spectrum of corruption boundaries.

The processed training data with different artifact levels is shown in Figure 7 Line 1. To demonstrate how random perturbations introduced in Sec 3.3 works, we also provide an example of render-time perturbation on correctly reconstructed 3DGS scenes in Figure 7 Line 2, where the perturbation intensity increases from left to right.

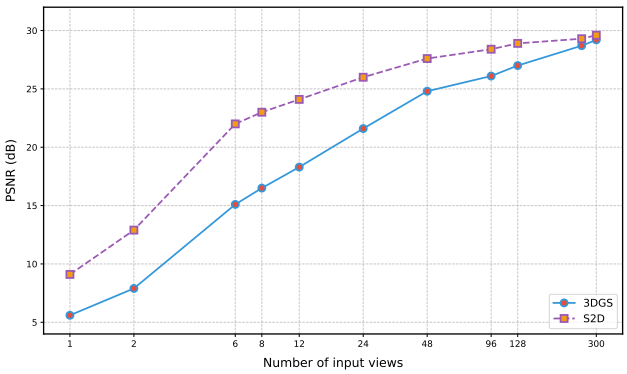

Figure 8. Reconstruction quality versus input density.

The large artifacts guide the model to recover basic object structure, while the perturbation applied on mild corruptions can further encourage detail enhancements, especially textures and edges. Therefore, our model is able to handle a large range of input intensity.

9. Evaluation on input density

We evaluate S2D's ability when faced with different amount of input images, the results on DL3DV scenes are provided in Figure 8.

As shown, our reconstruction quality improves steadily when increasing the number of input views. S2D consistently outperforms 3DGS with large improvements in sparse input and extra enhancement in dense input settings. This shows that S2D can be applied in general situations instead of fixed input views.

10. Performance report

The computational cost of S2D fixer is low, including 11.1 GB GPU usage and 1 FPS processing speed upon image resolution 1024×576 on a single RTX 4090. Under the same setting, Stable Virtual Camera (SEVA) results in 16.4 GB GPU usage and 0.08 FPS generation speed, while DIFIX is on par with S2D with 10.9 GPU usage and 1 FPS processing speed.

The reported processing speed means that the generation of novel guidance with S2D creates very small overhead comparing with original reconstruction (*e.g.* on 3DOVS dataset, with 30 seconds fixing is only $1/30$ of the total 15 minutes reconstruction), empowering the application of S2D in diverse scheme combinations while eliminating concerns about noticeable efficiency degradation.

Figure 9. Extra results of DiffusionGS and driving scene comparison.

11. More Comparison

We provide more qualitative comparison in Figure 9. For the same scenes as in Figure 4, the latest sparse-view reconstruction method DiffusionGS [5] (mainly object-centric) largely degrades with respect to in-the-wild scenes.

In line 2 we compare the side camera results on Waymo Open Dataset [38]. Operating on the same artifact image, DIFIX generates wrong car details and fuzzy road with supersaturated lane lines, while ours is more consistent.

12. More Implementation Details

Due to the page limit of the main paper, we provide more implementation details and discussion here.

Evaluation standard. While many feed-forward reconstruction researches now tend to conduct evaluation only on a few images from the original scene capture (especially sparse-view reconstruction such as 2-view reconstruction), we believe that this is insufficient to represent the actual and overall reconstruction quality of the complete scene.

Therefore, evaluation results reported in this paper are tested on **all other frames** of a scene that are not used for training or reference, identical to the evaluation standard of traditional reconstruction methods.

Evaluation on static scenes is conducted with the max pixel size of 255000 (the original setting for π^3 [54], *e.g.* 672×378) per image while remaining the original aspect ratio. This is to align with the default rendered image of point cloud for detail consistency, which is not compulsory. While many feed-forward methods only support fixed resolution or aspect ratio such as 256×256, S2D can work on any resolution and aspect ratio. In driving scenes, we evaluate the results on resolution 1024×576 as the setting in StreetCrafter [65].

Rendering for point cloud. We render point cloud images generated by VFMs through *pyrender*.

Specifically, we initialize an empty scene with black background, and generate mesh from reconstructed point cloud.

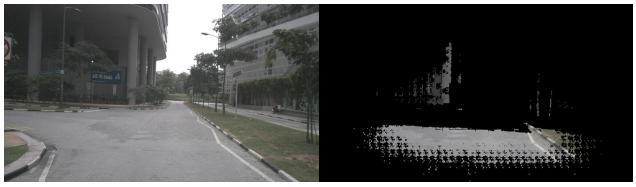

Figure 10. Point cloud rendering with single image input.

We applied *MetallicRoughnessMaterial* with *metallicFactor* set at 1.0 and *baseColorFactor* set as [1.0, 1.0, 1.0, 1.0] to all meshes to maintain the original point color.

Given cameara intrinsics and poses, the corresponding point cloud images are then rendered through *PerspectiveCamera*.

Data and training. The S2D artifact fixer is trained on our processed DL3DV-960[8, 21] dataset (train split) on 1-7K scenes for 75000 steps. The training images are augmented with random horizontal flip. All images are resized to 512×512 during training. The model is implemented in PyTorch and optimized with the AdamW optimizer, weight dtype is float32.

13. Limitation and future work

While our methods utilize VFM priors, we share the same limitation as VFMs. If the input images are both extremely sparse and low in texture, the resulting point cloud can be quite fragmentary. We show such a failure case in Figure 10 from the NuScenes dataset. This means the point cloud can hardly provide sufficient guidance and the fixing can only work on smaller artifacts. Fortunately, the VFM can be easily replaced to avoid similar situation.

The current fixing model uses point cloud for structural guidance and reference view for textural guidance, but we think the ultimate goal is to directly utilize both structural guidance and textural guidance from reference views while still work as an efficient image-level fixing. We are now working on further spatial feature extraction and cross-attention to pursue this objective.