CSG-Fusion: Consistent Sparse-View Gaussian Splatting via Matching-based Fusion

Supplementary Material

7. Training Details

7.1. Patch-Level Mask from Pixel Matching Prior

We obtain patch-level matching mask by binning each pixel-level match $\mathcal{M} = \{(u, v)\}$ into its corresponding patches (x, y), with u belonging to patch $\mathcal{P}_x \in I^1$ and v belonging to patch $\mathcal{P}_y \in I^2$.

Algorithm 1 Patch-Level Mask from Pixel Matching Prior

Require: Image size $H \times W$, pixel-level matching pairs $\mathcal{M} = \{(u, v)\}$, patch size (P_H, P_W)
Ensure: Binary patch-level matching matrix $\mathbf{m}_{\text{patch}} \in \{0, 1\}^{N_p \times N_p}$

```
Compute $N_p = \frac{H}{P_H} \cdot \frac{W}{P_W}$
Initialize $\mathbf{m}_{\text{patch}}$ as a zero matrix
for all $(u, v) \in \mathcal{M}$ do
    $x \leftarrow$ patch index of $u$ (row: $\lfloor u_y / P_H \rfloor$, col: $\lfloor u_x / P_W \rfloor$)
    $y \leftarrow$ patch index of $v$ (row: $\lfloor v_y / P_H \rfloor$, col: $\lfloor v_x / P_W \rfloor$)
    $\mathbf{m}_{\text{patch}}(x, y) \leftarrow 1$
end for
return $\mathbf{m}_{\text{patch}}$
```

8. More Experimental Analysis

We present additional comparisons with previous SOTA pose-dependent and pose-free methods under varying degrees of image overlap: Figure 13 for high overlap, Figure 12 for medium overlap, Figure 11 for low overlap, and Figure 10 for very low overlap.

In addition, we provide further comparisons on the DTU dataset in Figure 14 and Table 4. Following the evaluation protocol of NoPoSplat [31], we report metrics without the mask averaging operation for the ScanNet++ dataset in Figure 15 and Table 3.

Finally, to illustrate the efficiency trade-offs of our method, we report the average peak GPU memory usage during inference compared with baseline methods in Figure 9. Note that all evaluation experiments are conducted on a single NVIDIA A40 GPU.

9. Discussion And Limitation

Two main limitations of our method arise from the properties of the proposed fusion strategy. First, the performance of our matching-based fusion depends on the degree of overlap between input views, leading to degraded rendering quality in low-overlap regions (Table 1). Second, while the distance-filter mechanism provides an effective trade-off, its computational cost increases with the degree of overlap in the input views. Finally, our method does not explicitly model multi-view inputs. Addressing these issues will be the focus of our future work.

Table 3. **Comparison on ScanNet++ follow NoPoSplat [31].** Different from evaluation methods in Table 1, here we compute average metrics on all pixels.

Training Data	Method	ScanNet++		
		PSNR ↑	SSIM ↑	LPIPS ↓
ScanNet++	Ours	13.556	0.490	0.385
	Splatt3R	13.025	0.478	0.393
RealEstate10K	PixelSplat	18.422	**0.720**	**0.278**
	MVSplat	17.137	0.687	0.298
	NoPoSplat	**19.431**	0.686	0.293

Table 4. **Zero-shot Comparison on DTU follow NoPoSplat [31].**

Training Data	Method	DTU		
		PSNR ↑	SSIM ↑	LPIPS ↓
ScanNet++	Ours	10.980	0.374	0.585
	Splatt3R	11.756	0.372	0.523
RealEstate10K	PixelSplat	15.069	0.539	0.342
	MVSplat	14.542	0.537	0.324
	NoPoSplat	**17.906**	**0.631**	**0.281**

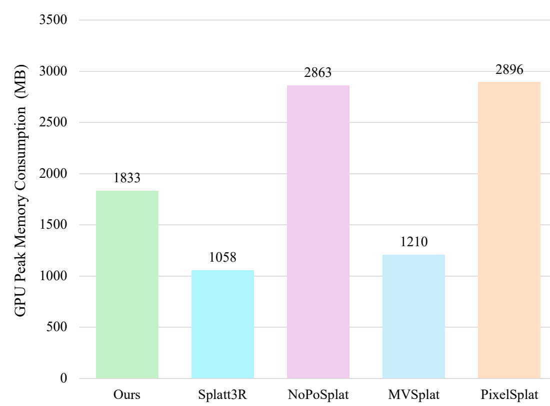

Figure 9. Comparison of peak GPU memory usage on Scannet++ dataset at evaluation stage.

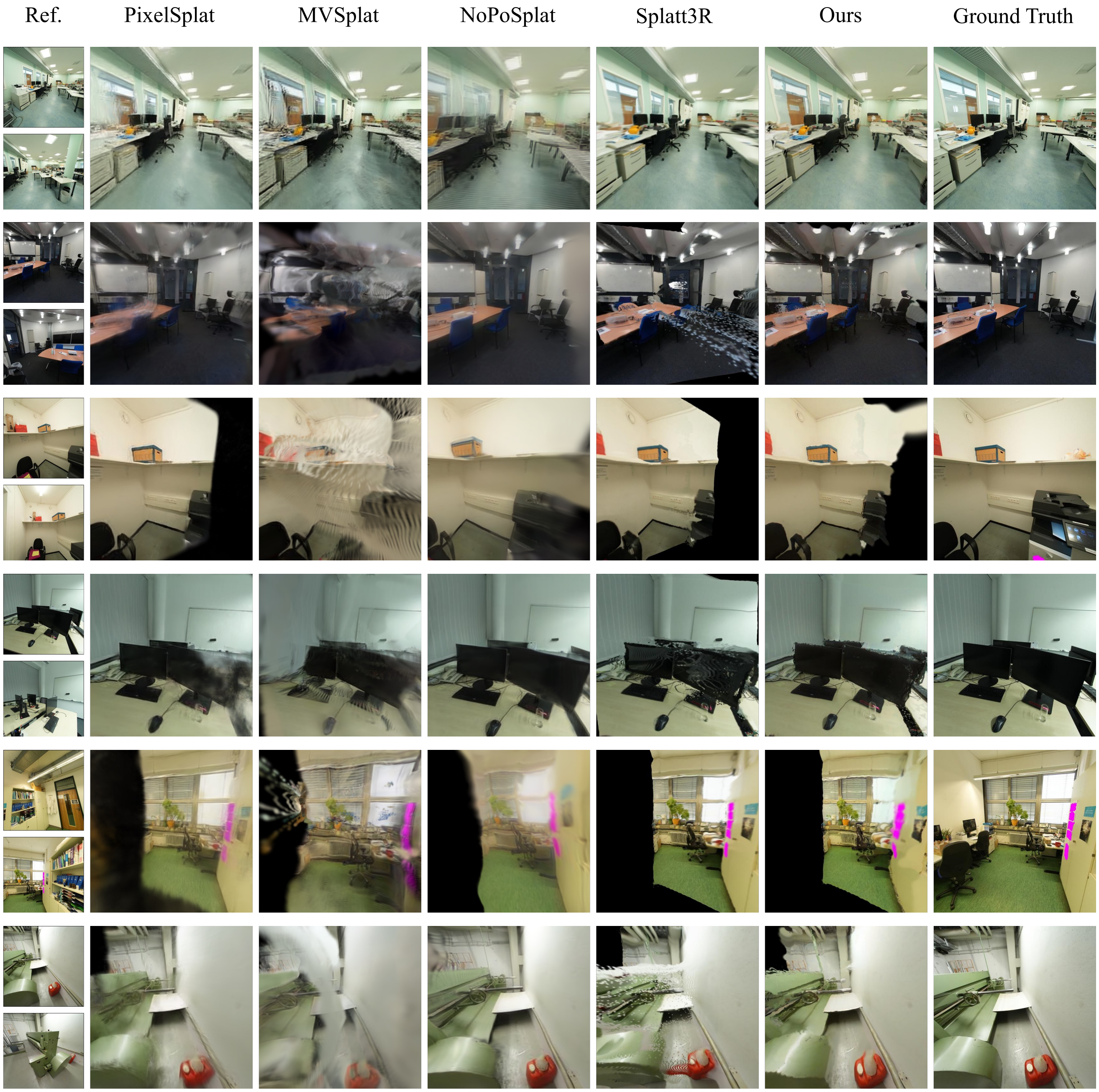

Figure 10. More comparisons of the ScanNet++ dataset with 30% overlap input views.

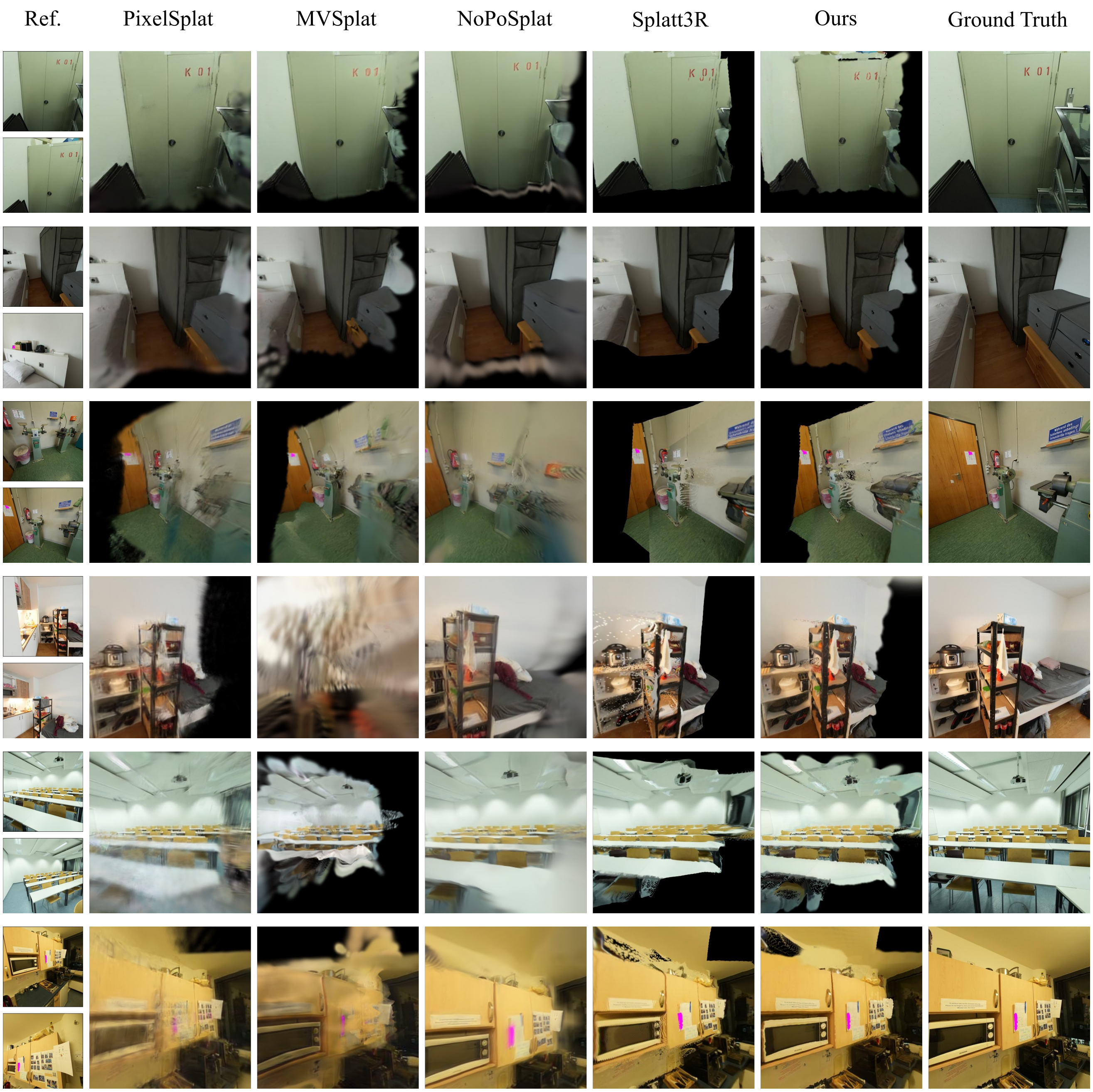

Figure 11. More comparisons of the ScanNet++ dataset with 50% overlap input views.

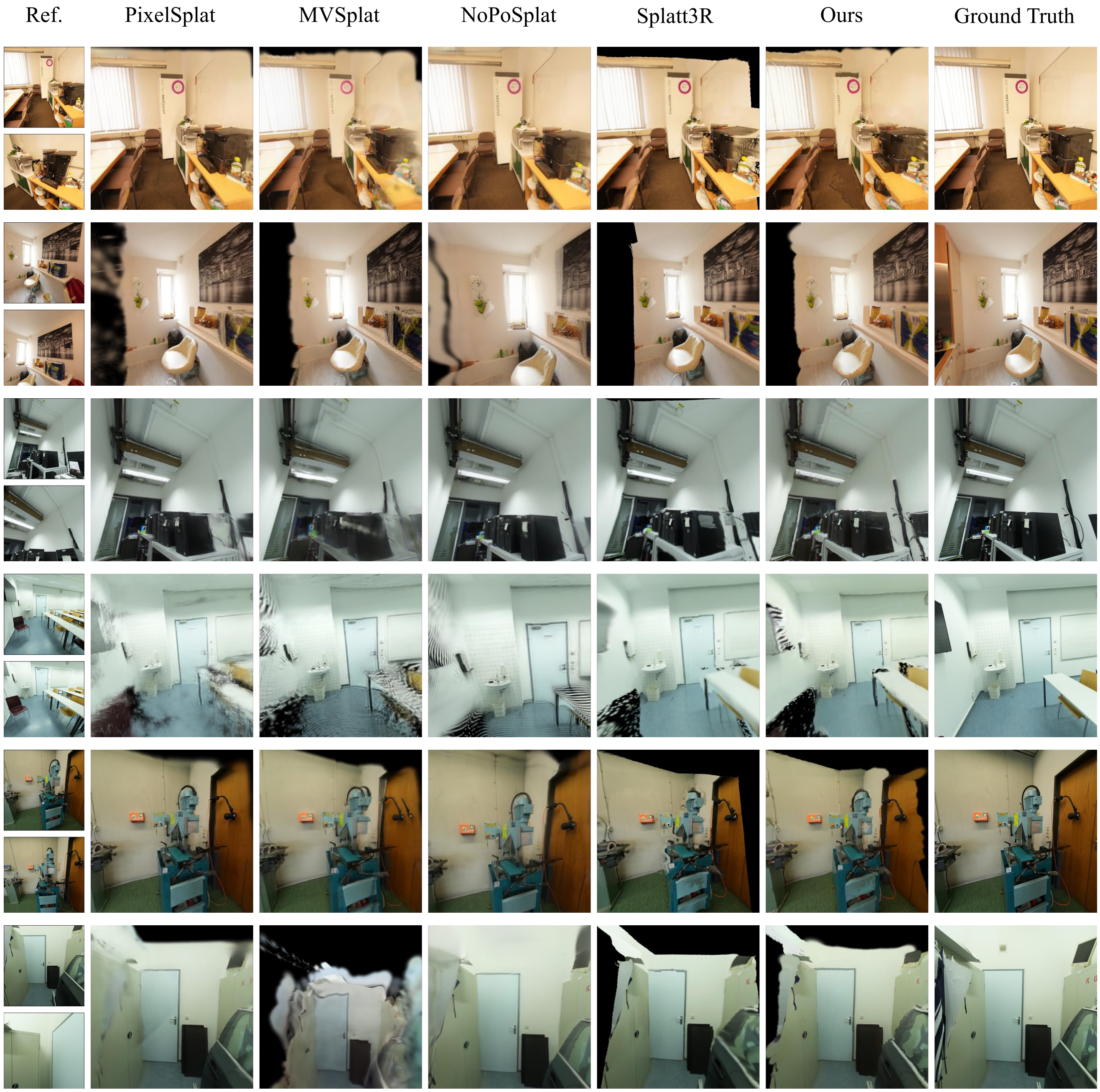

Figure 12. More comparisons of the ScanNet++ dataset with 70% overlap input views.

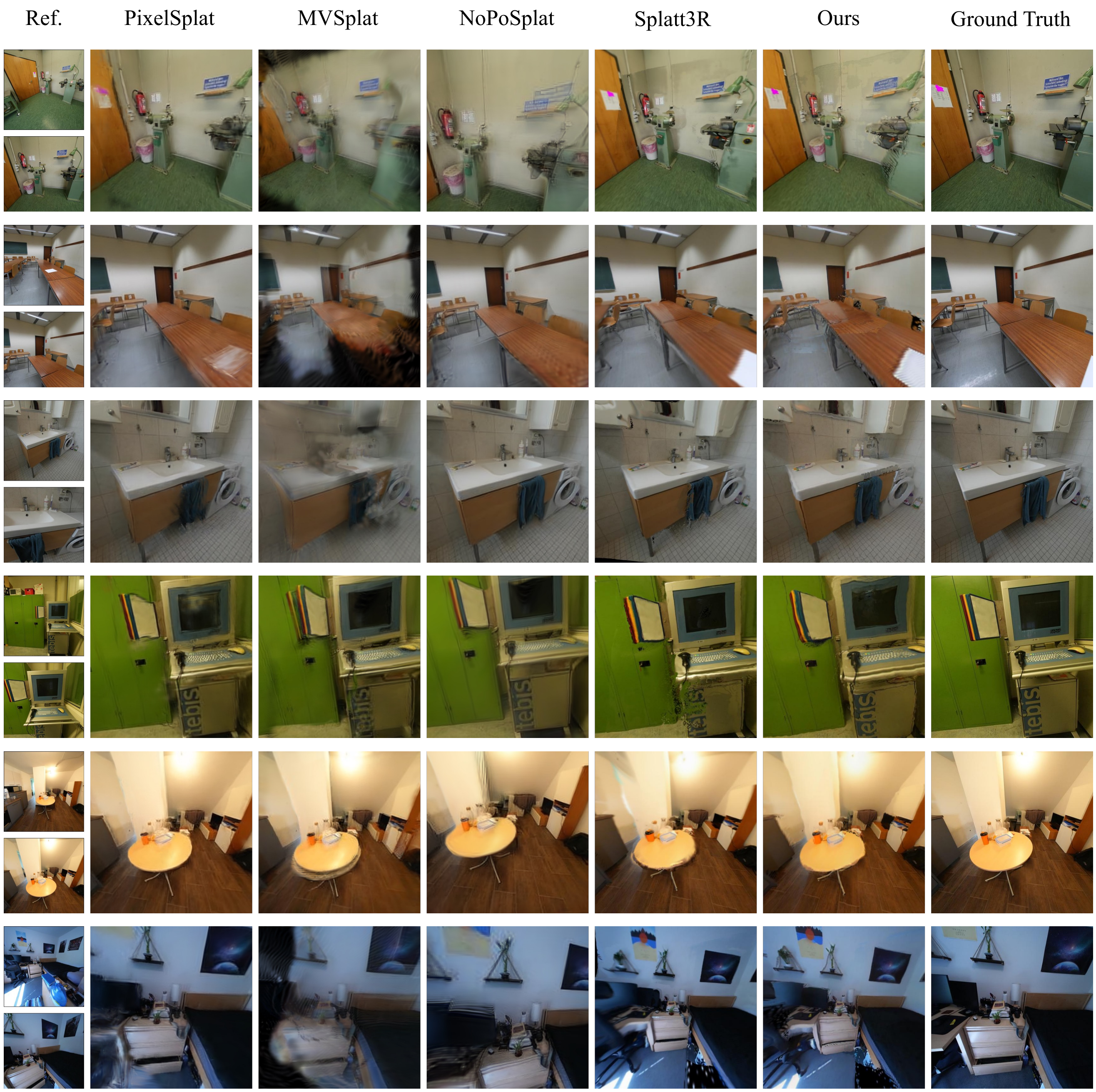

Figure 13. More comparisons of the ScanNet++ dataset with 90% overlap input views.

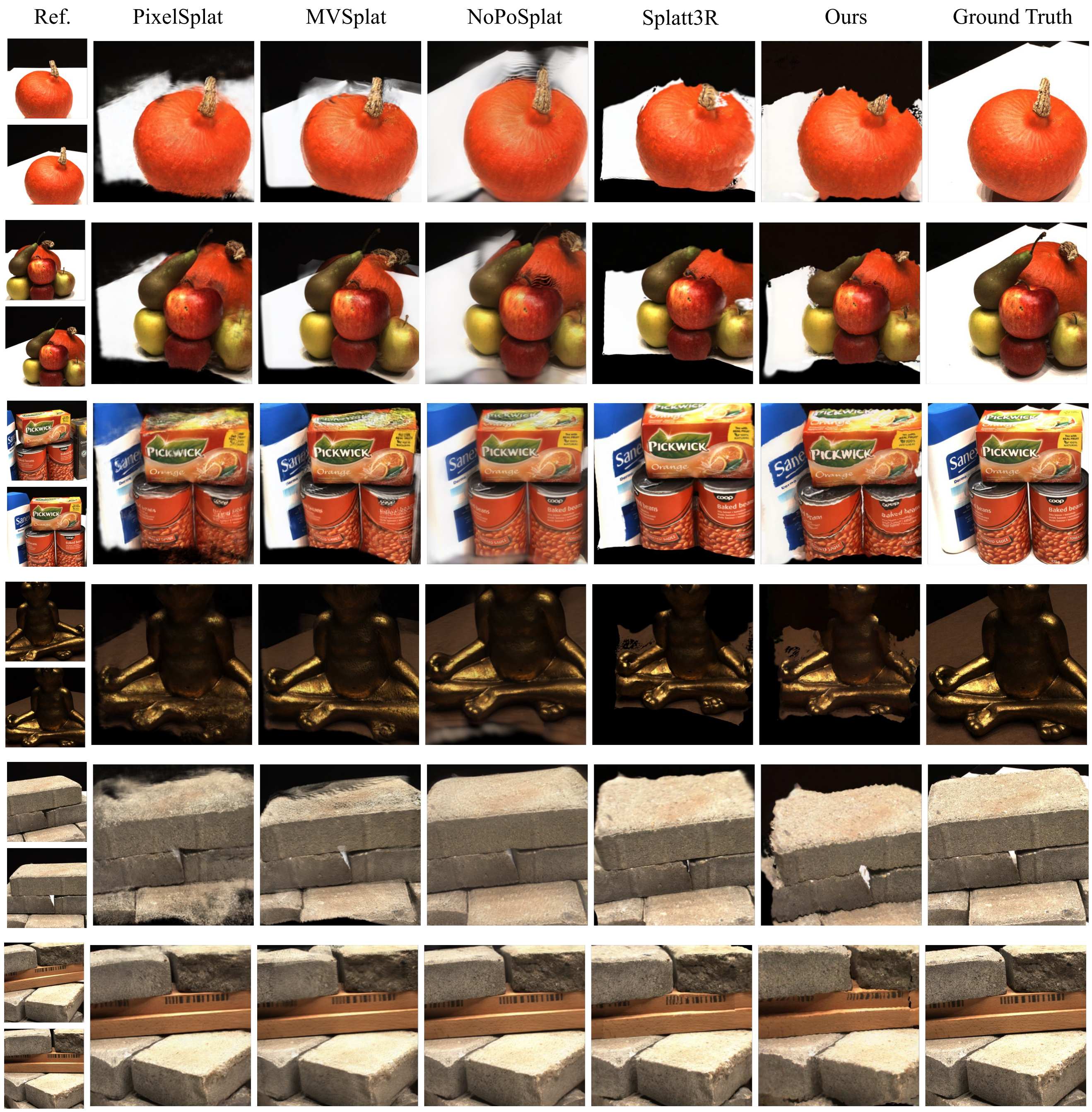

Figure 14. More comparisons of DTU dataset for Cross-dataset generalization.

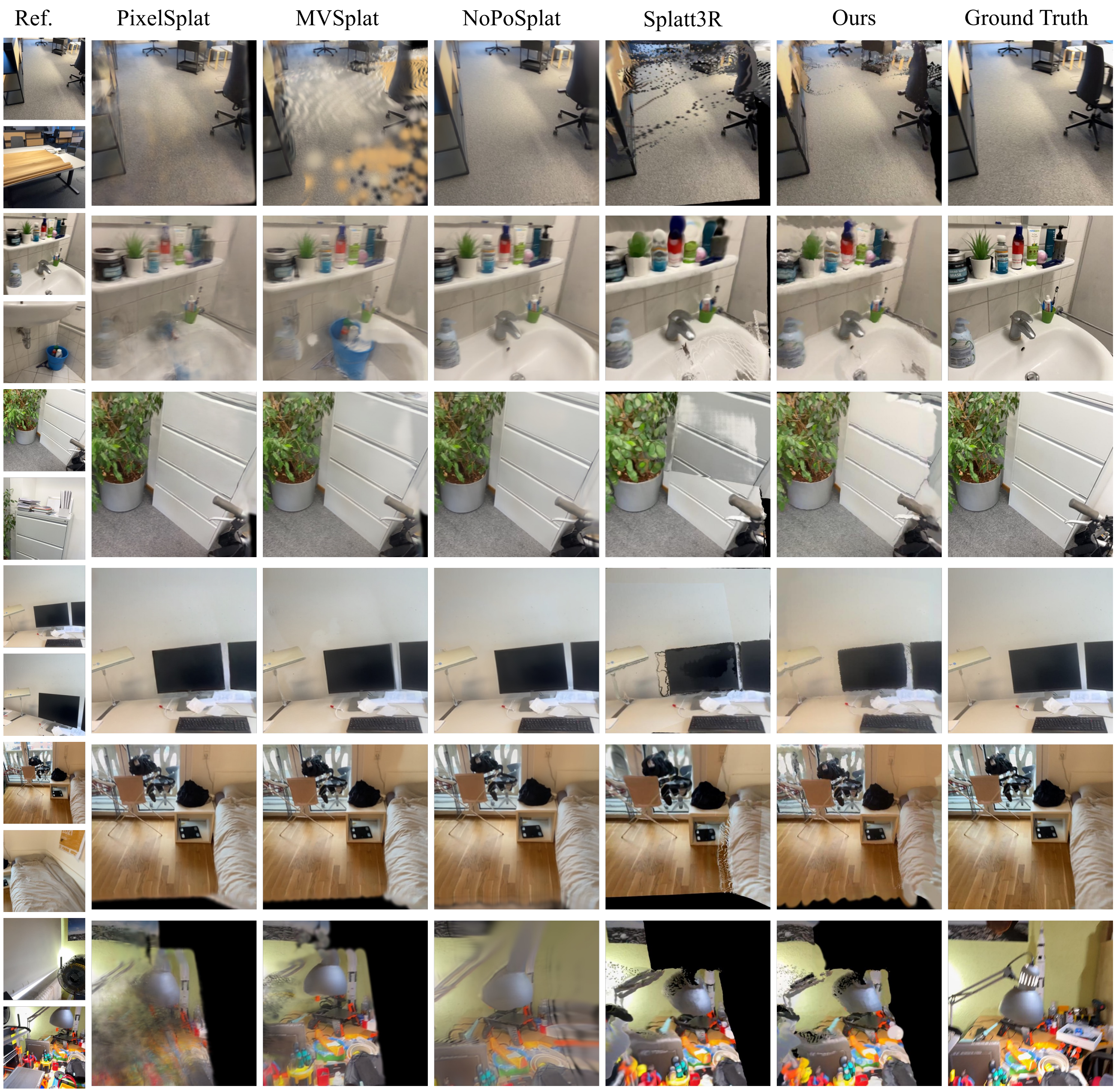

Figure 15. More comparisons of ScanNet++ dataset evaluation follow NoPoSplat [31] without masked average loss.