StreamGS: Online Generalizable Gaussian Splatting Reconstruction for Unposed Image Streams

Supplementary Material

Implementation Details

For training, we followed the frame sampling strategy in MVSplat. Context pairs are randomly sampled from video frames, with a constraint that the temporal distance between them is no less than 25 frames. 5 target frames are then selected from the frames wthin the context pair. For video datasets, frames are sampled at an interval of 6. For the multi-view image dataset MVImgNet, the interval is reduced to 2. Specially, for RE10K, the target frame is sampled outside the context pair to make the comparison more distinct.

Additional Comparisons on Higher Resolution.

Our method can naturally be extended to higher resolutions when sufficient GPU memory is available (1x 80G A100 GPU in current implementation). Results at a higher resolution of 384×384 are provided in Tab. 6.

	MVImgNet			ACID		
	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑
MVSplat	15.64	0.59	0.47	29.33	0.22	0.83
StreamGS	25.55	0.27	0.87	28.70	0.19	0.84

Table 6. Comparison of rendering quality on higher resolution.

Additional Metrics for Merging Process.

We include additional quantitative quality metrics—SSIM and LPIPS—in Tab. 7. There is indeed a degradation in reconstruction quality. However, considering the significant reduction in Gaussian primitives (36.71% and 40.48% reduction, saving about 18,420 and 20,311 primitives per frame), the quality drop is relatively minor.

	MVImgNet					ACID				
	CR ↑	RR ↑	PSNR ↑	LPIPS ↓	SSIM ↑	CR ↑	RR ↑	PSNR ↑	LPIPS ↓	SSIM ↑
w/o merge.	1.00	0%	25.67	0.22	0.82	1.00	0 %	29.25	0.13	0.86
merge all	1.58	36.71%	25.05	0.31	0.79	1.68	40.48%	28.50	0.15	0.84

Table 7. Evaluation of the efficiency improvements of the Gaussian merging process and its impact on rendering quality. "CR" and "RR" are short for compression ratio and reduction ratio for Gaussian primitves, respectively.

Ablation Studies on Framework Design

Adopting off-the-shelf MASt3R with an additional GS head (MASt3R+GSH) is a straightforward solution. However, this design introduces several limitations that our method effectively addresses. **(1) Redundant prediction and fusion challenges:** MASt3R requires processing the same image $\mathbf{I}^t$ twice—$\phi_{3D}((\mathbf{I}^t, \mathbf{I}^{t-1}))$ and $\phi_{3D}((\mathbf{I}^{t-1}, \mathbf{I}^t))$ (Eq. 3)—leading to duplicate Gaussian predictions and fusion complications. **(2) Higher complexity and latency:** Introducing a GS head increases training difficulty and delays inference. While MASt3R primarily focuses on geometry, predicting GS parameters (additional 84 channels) requires richer texture information. For example, Splatt3R uses a heavier DPT backbone to support GS prediction. In contrast, our method uses a lightweight CNN that directly takes images and matching features as input, offering a more streamlined solution. Efficiency comparisons in Tab. 8 show that our approach is faster and more parameter-efficient. **(3) Insufficient texture awareness and lower rendering quality:** MASt3R lacks texture-aware capabilities, resulting in inferior rendering, as it is trained for only geometry reconstruction. We conduct a fair comparison by training both MASt3R+GSH and our model for the same iterations, as shown in Tab. 9. Thanks to its lighter design, our model converges faster and achieves superior rendering quality.

	Coarse Predictor		Total	
	time/ms	param/M	time/ms	param/M
MASt3R + GS Head	19.63	695.29	122.35	697.16
StreamGS	**16.47**	**656.74**	**120.91**	**658.61**

Table 8. Efficiency comparison to MASt3R + GS head. Gaussian heads are borrowed from Splatt3R.

	MVImgNet			ACID		
	PSNR ↑	LPIPS ↓	SSIM ↑	PSNR ↑	LPIPS ↓	SSIM ↑
MASt3R + GS Head (15k iters)	19.46	0.45	0.73	15.15	0.42	0.60
StreamGS (15k iters)	**23.05**	**0.27**	**0.80**	**27.12**	**0.16**	**0.83**

Table 9. Comparison of novel view synthesis to MASt3R + GS Head. Both are trained with 15k iterations for fair comparison.