

Phase-DGS: Phase-guided Dynamic Gaussian Splatting from Unsynchronized Multi-view Video

Hosung Jeon^{†1}, Jun Young Jeong¹, Sangwoon Kwak^{†1}, Joonsoo Kim¹, Sangmin Kim², Jaesik Park², Won-Sik Cheong¹, Hyon-Gon Choo¹

¹Electronics and Telecommunications Research Institute (ETRI) ²Seoul National University

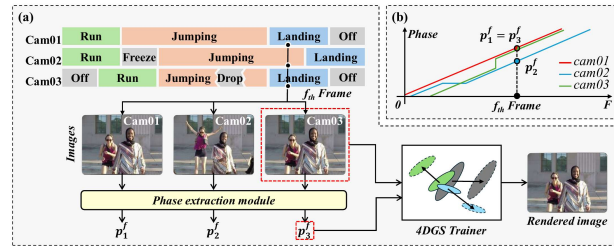
Phase-DGS estimates time from each image to train 4DGS on unsynchronized cameras.

Dynamic Gaussian Splatting (DGS) assumes every camera captures the scene at the same instant. Real rigs start late, drop frames and freeze, so frames that share an index show different moments, producing ghosting, floaters and distorted geometry. Hardware sync is costly, and data-driven alignment needs audio, rich texture or visible people.

- **Phase, not timestamp.** A per-frame phase estimated from image content replaces the timestamp input of 4DGS.
- **No phase labels.** A learnable feature bank keeps phases consistent across cameras.
- **Plug-and-play.** Works with RealTime-4DGS and FreeTimeGS without modifying the backbone.

Key Idea: estimated Phase, Not Timestamp

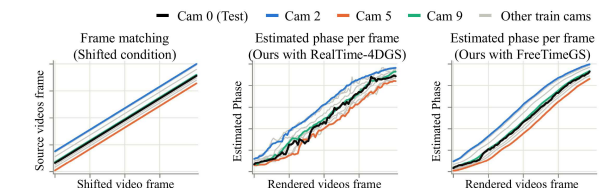
Same motion state, same phase, on any camera.



(a) Frames sharing an index show different motion states; each frame gets a phase that drives the 4DGS trainer. (b) An offset shifts the phase curve, a freeze flattens it, and a drop makes it jump.

Phases Emerge Without Labels

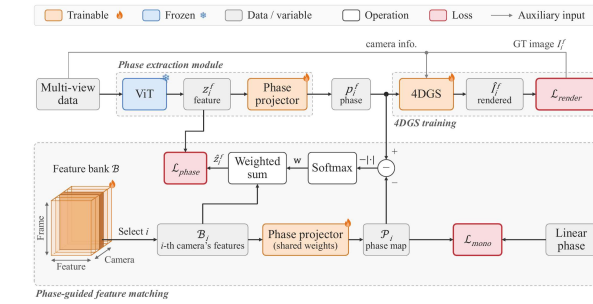
Estimated phases follow the true frame order of every camera.



Jumping, Shifted. Left: true source frame of each camera after random start offsets. Middle, right: phases estimated with RealTime-4DGS and FreeTimeGS reproduce those offsets.

Method

Phase extractor + learnable feature bank, trained end-to-end.



- **Phase extraction.** Frozen DINOv2 patch tokens are averaged into a feature z ; a small MLP maps z to a phase $p \in [0, 1]$.
- **Phase-consistency loss.** The bank feature retrieved at phase p must match z , which aligns phases across cameras.
- **Monotonic loss.** Bank phases must increase along each camera's frames.
- **Rendering loss.** The 4DGS backbone G is trained with $G(p)$ in place of $G(t)$.

Quantitative Results

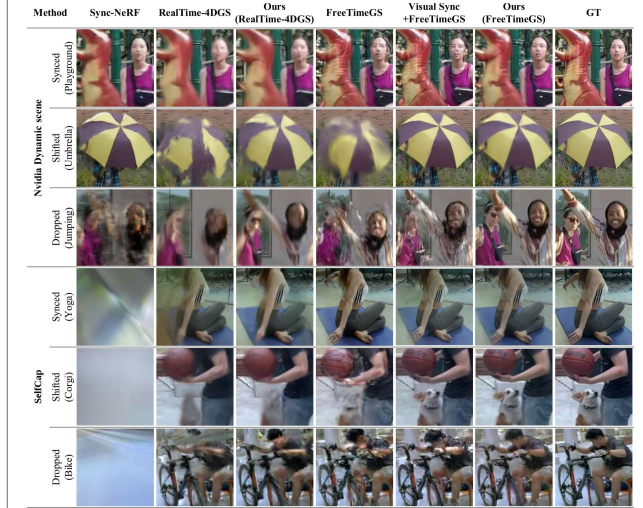
Average PSNR $\uparrow$ / LPIPS $\downarrow$ over three scenes per dataset (FreeTimeGS backbone)

Condition	Method	SelfCap		Nvidia Dynamic Scene	
		PSNR $\uparrow$	LPIPS $\downarrow$	PSNR $\uparrow$	LPIPS $\downarrow$
Synced	FreeTimeGS	34.18	0.010	31.17	0.022
	+ VisualSync	33.45	0.013	30.43	0.025
	Phase-DGS (ours)	34.73	0.012	30.42	0.030
Shifted	FreeTimeGS	27.17	0.026	23.79	0.068
	+ VisualSync	29.61	0.018	27.96	0.033
	Phase-DGS (ours)	32.95	0.014	29.84	0.037
Dropped	FreeTimeGS	27.89	0.025	25.31	0.055
	+ VisualSync	29.74	0.019	28.06	0.039
	Phase-DGS (ours)	34.38	0.012	30.67	0.030

Average over three scenes per dataset, FreeTimeGS backbone. Bold: best in each condition. RealTime-4DGS and Sync-NeRF results are in the paper.

Qualitative Results

Baselines ghost and distort moving objects; Phase-DGS keeps them sharp.



Synced, Shifted and Dropped conditions on Nvidia Dynamic Scene (top) and SelfCap (bottom).

Ablation

Both self-supervised components are essential (Dropped, FreeTimeGS, PSNR $\uparrow$).

Variant	Jumping	Bike
Phase-DGS (ours)	29.39	31.21
w/o monotonic loss	24.42	25.90
w/o feature-bank learning	24.79	27.64

Conclusion

Phase-DGS trains 4DGS from unsynchronized multi-view video by replacing timestamps with phases read from the images: no sync hardware, no phase labels, no backbone changes.

Limitation: highly periodic motion (e.g., Balloon1) makes phases ambiguous.



Project page & videos
cc-dev74.github.io/Phase-DGS