VIDEOPORT: FROM MONOCULAR VIDEO TO LIVE HUMAN TELEPRESENCE

Anonymous authors
Paper under double-blind review

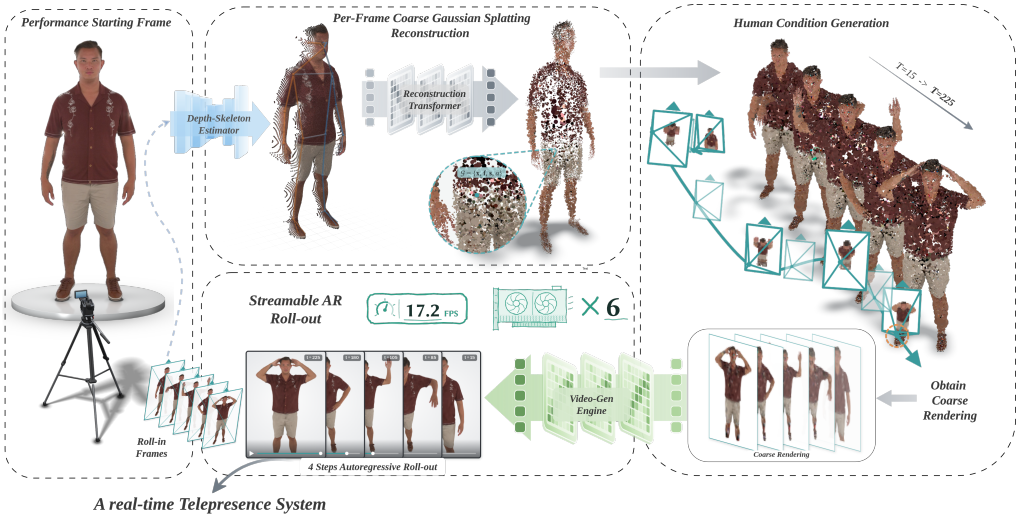


Figure 1: Streaming monocular telepresence via causal geometric video diffusion.

ABSTRACT

Human telepresence aims to enable remote viewers experience a live performance from freely chosen viewpoints, however, achieving this from a single RGB camera requires inferring surfaces and appearance that the camera cannot observe. Existing approaches face a fidelity–realism tradeoff. Feedforward 3D reconstruction is geometrically consistent, but often produces coarse appearance. Video diffusion synthesizes realistic detail, but when conditioned on its own history, drifts from the person being captured. We present *VideoPort*, a two-stage framework fixes geometry via feedforward 3D reconstruction, then injects realistic detail using causal video diffusion refinement. Reconstructed geometry guides novel-view synthesis, while surface-aligned observations provide additional appearance conditioning during training. To address error accumulation through generated history, we propose deep self-rollout supervision (DSR) with paired performance targets to train the refiner to recover from its own imperfect predictions. As a result, VideoPort preserves performance-specific appearance throughout minutes-long sequences and achieves state-of-the-art reconstruction quality on 3 benchmarks. We further deploy it as a live system that performs continuous, real-time novel-view synthesis from a single consumer RGB camera.

1 INTRODUCTION

Virtual human telepresence enables immersive remote collaboration by allowing viewers to freely observe a live performance from arbitrary viewpoints. While established systems (Orts-Escolano et al., 2016; Lawrence et al., 2024) achieve high fidelity, their reliance on synchronized multi-camera and depth rigs creates prohibitive cost and setup barriers. Making telepresence broadly accessible requires synthesizing realistic novel views from a single consumer RGB camera—a severely ill-posed

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

task demanding continuous estimation of dynamic occluded geometry and unobserved textures under complex apparel deformation.

Existing approaches handle this single-view ambiguity primarily through two regimes. Parametric kinematic models and feedforward 3D estimators (Li et al., 2020; Qiu et al., 2025b) offer fast geometric scaffolding, but rigid skinning fails to capture non-rigid clothing dynamics, causing oversmoothed surfaces and geometric tearing. Conversely, generative multiview and video diffusion models (He et al., 2024; Li et al., 2024; Zhang et al., 2024) hallucinate detailed textures, yet their unconstrained latent spaces lead to cross-view inconsistencies and severe camera drift. This indicates that realistic synthesis requires neither purely parametric rigs nor unconstrained generative priors, but an explicit dynamic 3D substrate that firmly anchors generative refinement.

Crucially, live telepresence requires causal, low-latency execution. Multi-pass denoising, per-scene optimization, and bidirectional clip batching introduce prohibitive delays for continuous communication. Furthermore, naive causal video diffusion suffers from severe exposure bias: autoregressive conditioning on past predictions compounds small geometric and textural errors, precipitating catastrophic identity and visual drift over multi-minute rollouts. To sustain continuous telepresence, the causal refiner cannot merely train on clean ground-truth history; it must directly train on its own degraded rollouts to learn active error recovery.

To address these challenges, we present *VideoPort*, a framework coupling feedforward 3D human reconstruction with causal video diffusion refinement. A skeleton-conditioned predictor reconstructs complete-body points and Gaussian attributes to guide target-view rendering, while surface-aligned reference conditioning dynamically supplies appearance from live and past observations. To prevent compounding history errors, we introduce deep self-rollout supervision (DSR), which directly supervises the refiner on its own generated rollout states. Together, these components enable continuous, high-fidelity monocular telepresence. Our contributions are:

- **Geometry-guided feedforward human reconstruction.** A skeleton-conditioned predictor of complete-body points and Gaussian attributes from visible geometry and whole-body pose, trained with temporal supervision.
- **Surface-aligned appearance conditioning.** A mechanism that projects current and past observations onto the reconstructed surface to condition the diffusion refiner on performance-specific appearance.
- **Deep self-rollout supervision (DSR) for long-horizon stability.** Adapting differentiable history replay to paired performance targets, directly supervising student rollout states to mitigate compounding autoregressive drift across multi-minute sequences.
- **Streaming monocular telepresence.** An integrated, end-to-end framework enabling continuous, real-time novel-view synthesis of live human performances from a single RGB camera.

2 Related Works

Multiview Human Telepresence. Multiview capture supplies complementary geometry and appearance for live reconstruction. Holoportation (Orts-Escolano et al., 2016) supports real-time 3D capture, transmission, and rendering, while Project Starline (Lawrence et al., 2024) combines multiview color and depth capture with a dedicated display. Their hardware and installation demands motivate monocular telepresence using learned geometry and appearance priors.

Single-view Human Digitization. PIFu Saito et al. (2019) and PIFuHD Saito et al. (2020) infer geometry using pixel-aligned implicit functions, while ICON Xiu et al. (2022) combines body priors and surface normals for challenging poses. MonoPort Li et al. (2020) targets real-time reconstruction. Generative image-to-3D methods Wang & Shi (2023); Long et al. (2024); Liu et al. (2023); Wu et al. (2024a); Tang et al. (2025); Xu et al. (2024) learn priors from object datasets such as Objaverse Deitke et al. (2022; 2023), but transfer to clothed humans remains challenging. MagicMan He et al. (2024) and PSHuman Li et al. (2024) use multiview diffusion priors, while Human-3Diffusion Xue et al. (2024) couples multiview denoising with Gaussian reconstruction. Live capture additionally requires recovering changing appearance and nonrigid motion.

Animatable Avatars and Telepresence. Single-image methods predict articulated Gaussian avatars Zhuang et al. (2024); Qiu et al. (2025a); Dong et al. (2025); Buehler et al. (2026). LHM Qiu et al. (2025a) and LHM++ Qiu et al. (2025b) use feedforward reconstruction, with LHM++ supporting pose-free inputs and real-time animation rendering. Dream, Lift, Animate Buehler et al. (2026) lifts generated views into body-aligned Gaussians, while MoGA Dong et al. (2025) fits an avatar prior to diffusion-generated views. Large-scale Codec Avatars (Li et al., 2026) enables feedforward full-body modeling. Mon3tr (Lin et al., 2026) drives a personalized Gaussian avatar built offline from multiview observations for monocular telepresence. VideoPort combines per-frame geometry with causal refinement to recover garment and hair configurations not specified by body pose alone.

Diffusion Models for 3D/4D Synthesis. LTX HaCohen et al. (2024), Hunyuan Video Kong et al. (2024), and Wan Wan et al. (2025) provide video appearance and motion priors. 4Diffusion (Zhang et al., 2024) generates multiview videos from monocular input to guide 4D reconstruction. Joint multiview generation and per-sequence optimization pose latency challenges for streaming. Difix3D+ Wu et al. (2025) uses single-step diffusion to correct Gaussian-splatting artifacts Kerbl et al. (2023), while ViewCrafter Yu et al. (2024) conditions video diffusion on coarse point-cloud renderings. ActCam Khalifi et al. (2026) uses camera-aligned pose and depth for joint human-motion and camera control. VideoPort uses feedforward geometry to guide causal refinement along requested camera trajectories.

Streaming Video Diffusion. StreamDiffusionV2 (Feng et al., 2026) uses cache management, motion-aware noise control, and execution scheduling, while FlowCache (Ma et al., 2026) accelerates generation through feature reuse and KV-cache compression. Causal Forcing methods (Zhu et al., 2026; Zhao et al., 2026) study causal adaptation and few-step distillation. Self Forcing and its extensions (Huang et al., 2025; Cui et al., 2025) train with self-generated histories. Resampling Forcing (Guo et al., 2025) trains with model-degraded histories and clean targets, while Self Gradient Forcing (Zhuang et al., 2026) differentiably re-encodes detached generated context under distribution-matching supervision. We adapt differentiable replay to paired reconstruction, supervising predictions from student rollout states against target latents at matching times and viewpoints under perturbed history. Geometry and training-time retrieved observations condition recovery; we compare paired supervision with distribution matching.

3 METHOD

3.1 SYSTEM OVERVIEW

Given a monocular RGB video $I_{1:T}$, foreground masks, and a target camera trajectory, VideoPort synthesizes the captured performance from requested viewpoints. A feedforward reconstruction module predicts a coarse Gaussian representation $\mathcal{G}_t$ independently for each input frame. Rasterizing these representations along the target trajectory produces a coarse conditioning sequence $c_{1:T}$. A causal video diffusion refiner synthesizes target views from these renderings and generated history. During training, surface-aligned appearance retrieval provides additional conditioning (memory). We train the refiner to handle errors in both its geometric guidance and its self-generated history: deep self-rollout supervision (DSR) trains recovery toward paired target frames directly from its own rollout states. Block-causal attention and a rolling cache support streaming generation.

3.2 SINGLE VIEW 3D HUMAN RECONSTRUCTION IN REAL-TIME

We combine camera-facing depth with whole-body pose to complete partial monocular observations into a coarse 3D representation. The predictor operates independently on each frame without test-time optimization, while temporal supervision encourages output points to follow the captured motion (Appendix A.3). Its Gaussian renderings provide geometric and appearance guidance for the video refiner (Figure 2).

Partial observation. A frozen DINOv3 backbone (Siméoni et al., 2025) extracts a global image feature g_t and spatial patch features. A DPT-style decoder (Ranftl et al., 2020) predicts camera-facing depth $\hat{D}_t$, while a separate head estimates 3D whole-body joints. Depth is anchored to the known capture distance, and both prediction heads remain frozen during reconstruction training.

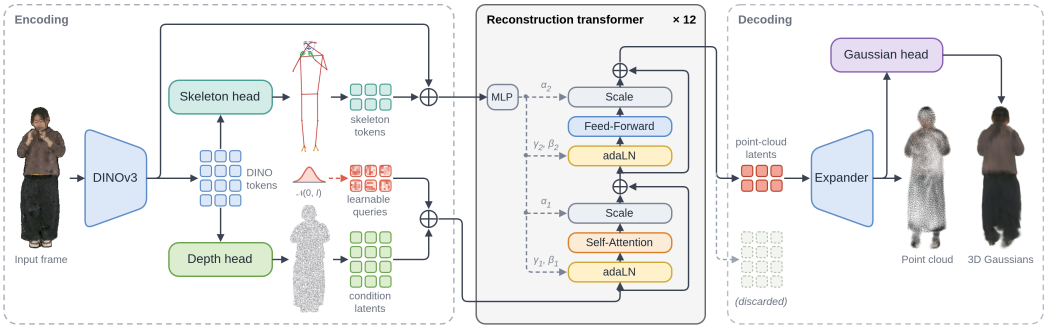


Figure 2: **Feedforward geometric and appearance reconstruction.** DINOv3 visual tokens condition depth and skeleton prediction, which modulate learned queries in the reconstruction transformer via adaLN-Zero. The retained latents are hierarchically expanded into a dense point cloud and then to 3D Gaussians to produce coarse novel-view priors.

Using fixed camera intrinsics, we back-project foreground pixels into a partial point cloud $\mathcal{C}_t$ representing the visible surface.

Reconstruction transformer. Following LVSM (Jin et al., 2025), learnable queries aggregate the partial observation into a complete representation. We subsample $\mathcal{C}_t$ via farthest-point sampling and concatenate its embedded coordinates with learned query tokens Q:

$$h_t^{(0)} = \left[Q \,\|\, \mathrm{MLP}\left(\gamma(\mathcal{C}_t^{\mathrm{fps}})\right) \right]. \tag{1}$$

Here, $\mathcal{C}_t^{\mathrm{fps}}$ is the sampled point cloud, γ denotes Fourier positional encoding, and the MLP maps encoded coordinates into the transformer embedding space. Joint self-attention exchanges information between the queries and observed geometry.

To guide completion of occluded regions, we condition on whole-body pose and global appearance. We encode joint coordinates with learned identity embeddings, contextualize them using a joint transformer, and pool the features into a skeleton descriptor $c_{\mathrm{skel},t}$. The conditioning vector is

$$f_t = W_{\mathrm{fuse}} \left[W_{\mathrm{img}} \,\mathrm{LN}(g_t) \,\|\, c_{\mathrm{skel},t} \right], \tag{2}$$

where LN denotes layer normalization. Through adaLN-Zero (Peebles & Xie, 2023), f_t predicts feature-wise shifts, scales, and residual gates for the query tokens in each block. Cloud tokens use standard layer normalization and exchange information with the modulated queries through joint self-attention. The final query features initialize hierarchical point decoding (Appendix A.3).

Decoding to Gaussians. We decode the final query features into points through two hierarchical expansion stages using predicted anchors and bounded coordinate offsets. Windowed attention over Morton-ordered features (Wu et al., 2024b) exchanges local spatial information. A per-point head predicts Gaussian opacity, rotation, anisotropic scale, color, and bounded center offsets. Appearance prediction additionally uses visibility-weighted image features sampled at projected point locations. The resulting Gaussians $\mathcal{G}_t$ are rendered along the target camera trajectory to produce $c_{1:T}$. We first train point-cloud reconstruction with geometric supervision, then attach the Gaussian head and apply multiview image supervision:

$$\mathcal{L}_{\mathrm{img},t} = \sum_{v \in \mathcal{V}_t} \ell_{\mathrm{img}}(\mathcal{R}(\mathcal{G}_t; \Pi_{t,v}), I_{t,v}), \tag{3}$$

where $\mathcal{V}_t$ indexes supervised views, $\mathcal{R}$ is the differentiable rasterizer, and $\Pi_{t,v}$ and $I_{t,v}$ denote the camera and target image for view v. All supervised views constrain the same Gaussian representation, encouraging multiview agreement. The point decoder is trained alongside the main

model, while the Gaussian head is trained separately with the main model frozen (detailed in Appendices A.3 and A.3).

3.3 3D-Aware Video to Video Translation for Human Performance Capture

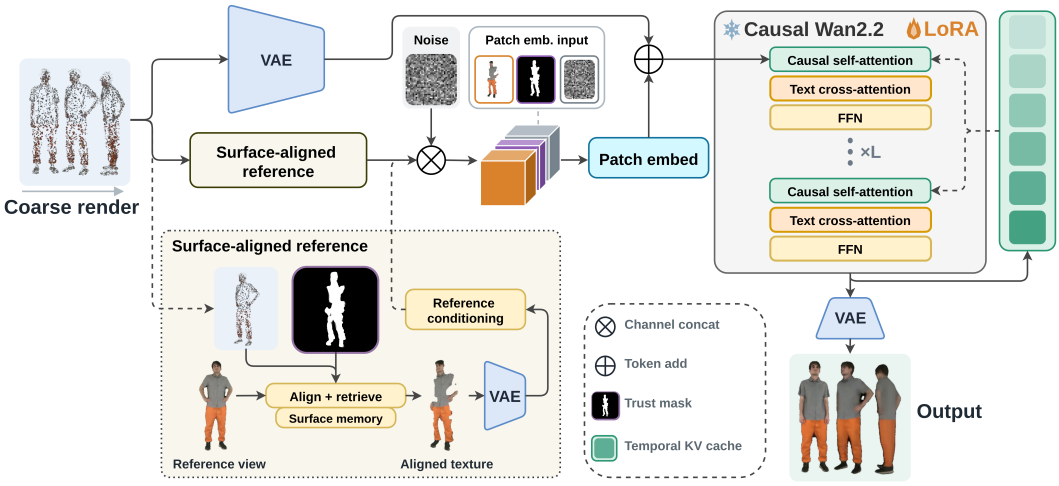


Figure 3: **Causal diffusion refinement.** Reconstructed geometry guides surface texture retrieval and trust masking to condition a LoRA-adapted Wan2.2 model alongside coarse renders. Causal attention and a rolling KV cache enable streaming generation, decoded into video by the VAE.

Geometry-conditioned video refinement. We adapt the pretrained Wan2.2-I2V-A14B model to recover clean target-view videos from $c_{1:T}$ (Figure 3). Off-the-shelf VAE or ControlNet-style Zhang et al. (2023) modules fail during continuous autoregressive rollouts, lacking the temporal mechanisms needed for stable human motion. To overcome this, our refinement integrates geometry deep within the latent space. During training, sparse, corrupted Gaussian representations produce coarse sequences paired with clean renderings of the same performance (Section A.1). The pretrained VAE encodes these sequences into conditioning latents z^c and target latents z. Retaining the native identity-conditioning pathway, for a noisy target $z_\sigma = (1-\sigma)z + \sigma\epsilon$, with $\epsilon \sim \mathcal{N}(0, I)$ and $\sigma \in [0, 1]$, the input embedding becomes:

$$h^{(0)} = \mathrm{PatchEmbed}\big(z_\sigma \,\|\, m \,\|\, z^I\big) + W_c * z^c. \tag{4}$$

Here, z^I and m denote the identity-conditioning latent and its mask. The projection W_c maps coarse-render latents onto the aligned spatiotemporal token grid. We freeze the pretrained backbone and VAE, optimizing only LoRA updates and conditioning adapters (θ) via conditional flow matching:

$$\mathcal{L}_{\mathrm{FM}} = \mathbb{E}_{\sigma,\epsilon}\left[\left\| v_\theta\big(z_\sigma, \sigma;\, z^c, \kappa, z^I\big) - (\epsilon - z)\right\|_2^2\right]. \tag{5}$$

The objective is averaged over paired training sequences. Architecture settings and training configurations are provided in Appendix A.4.

3.4 Surface-Aligned Reference Conditioning

To recover high-frequency appearance details omitted by coarse renderings, we use the reconstructed geometry to align observed textures with the target view. Direct warping supplies currently visible appearance, while a surface memory retrieves texture from regions observed in earlier frames.

Training-time geometry-guided texture retrieval. During training, we back-project target pixels using coarse depth to obtain surface points $\mathbf{X}_t(p)$. Projection into reference image r_t, with depth and coverage tests, yields warped texture and a validity mask. For historical retrieval, augmented temporal correspondences from multiview-fitted Dynamic 3D Gaussians (Luiten et al., 2024) transport these points to keyframe $s < t$:

$$\mathbf{X}_{t\to s}(p) = \mathbf{X}_t(p) + \mathcal{R}_t\left[\boldsymbol{\mu}_i^s - \boldsymbol{\mu}_i^t\right](p), \tag{6}$$

where $\boldsymbol{\mu}_i^t$ is the corrupted Gaussian center and $\mathcal{R}_t$ denotes normalized target-view rasterization. We project transported points into past reference frames, apply the same visibility tests, and fill missing texture from the most recent valid observation. Since our human estimator cannot provide surface tracking, this memory mechanisms operates solely during training.

Reference-conditioned refinement. We composite the retrieved texture over the coarse rendering and encode the result into VAE latents z^w. Separate coverage maps M^{d} and M^{m} record the fractions of each latent cell supported by direct and memory observations. Their weighted combination defines a trust mask:

$$\bar{m}_k = \min\left(1,\, M_k^{\mathrm{d}} + \lambda_{\mathrm{m}} M_k^{\mathrm{m}}\right), \tag{7}$$

where λ_{m} controls the relative trust assigned to past observations. We inject this appearance evidence through Wan's existing image-conditioning channels (Wan et al., 2025), preserving the initial identity frame. For subsequent latent frames, the conditioning latent is

$$y_k(u) = \begin{cases} z_k^w(u), & \bar{m}_k(u) > 0, \\ z_k^{\varnothing}(u), & \text{otherwise}, \end{cases} \qquad k \geq 1, \tag{8}$$

where u indexes spatial latent cells and $z^{\varnothing}$ denotes the encoded black-video baseline. The trust mask supplies the corresponding mask channels. Thus, y replaces z^I in Eq. equation 4, while the coarse-render retains geometric guidance. During training, we drop reference texture with probability 0.1 per sample, retaining identity and coarse-render conditioning so the refiner learns to operate without retrieved texture (details in Appendix A.7).

3.5 Streaming Autoregressive Video Refinement

Autoregressive Video Diffusion We adapt the refiner for streaming using block-causal attention and a rolling key-value cache (Huang et al., 2025), retaining an identity anchor and recent generated latents. Window-relative temporal rotary positions (Xiao et al., 2025) support generation beyond the training clip length. We initialize the causal model using ground-truth-history teacher forcing (Zhao et al., 2026) with Eq. equation 5, introduce reference latents through a further teacher-forced warm-up, and conclude with DSR (cache construction detailed in Appendix A.5).

Deep self-rollout supervision. Building on differentiable history replay (Zhuang et al., 2026), we formulate deep self-rollout supervision (DSR) for paired performance reconstruction. We directly supervise clean predictions against target latents under perturbed generated histories. This paired objective requires no additional distribution-matching loss. Each DSR iteration pairs gradient-free rollouts (Algorithm 1) with differentiable replay (Algorithm 2). We save intermediate inputs u_k and noise levels σ_k at sampled denoising exit steps. During replay, we re-encode detached generated histories perturbed with small Gaussian noise and predict selected frames from their saved states. Clean predictions are supervised against ground-truth latents z_k:

$$\begin{aligned} \tilde{x}_{0,k} &= u_k - \sigma_k v_\theta\Big(u_k, \sigma_k;\, \tilde{\mathcal{H}}_k, \mathrm{cond}_k\Big), \\ \mathcal{L}_{\mathrm{DSR}} &= \frac{1}{|\mathcal{S}|} \sum_{k\in\mathcal{S}} \|\tilde{x}_{0,k} - z_k\|_2^2 . \end{aligned} \tag{9}$$

Here, $\mathcal{S}$ contains supervised frame indices, $\tilde{\mathcal{H}}_k$ is the perturbed history, and cond_k contains geometric and reference conditioning. Targets z_k are encoded from uncorrupted multiview-fitted renderings (Section A.1). Replay differentiates through history re-encoding and current-frame prediction, without backpropagating through the computations that generated the recorded history (Appendix A.6).

4 Experiments and Results

Virtual human telepresence requires synthesized views to maintain metric 3D geometric alignment, preserve temporal coherence across long horizons, and operate causally at interactive rates. We empirically validate our system across three axes: (i) full-system benchmark comparisons of latency vs. novel-view fidelity; (ii) geometric and temporal accuracy of the 4D feedforward substrate; and (iii) ablation of causal autoregressive training and appearance conditioning. Augmented multiview-fitted Gaussian reconstructions provide training supervision and evaluation targets only; they are not used to construct test-time conditioning or memory correspondences.

4.1 Experimental Setup

We evaluate on synchronized studio captures from **DNA-Rendering** (Cheng et al., 2023), high-fidelity subjects from **ActorsHQ** (Işık et al., 2023), and in-the-wild dynamic sequences from **4DAnyone** (Jin et al., 2026). For ground-truth targets along continuous camera sweeps, we fit per-sequence Dynamic 3D Gaussians (Luiten et al., 2024). Fitted reconstructions provide training data and evaluation targets, but are not supplied to the model at test time, so reported inference runs disable memory retrieval. All models receive strictly monocular RGB input from a single static camera and synthesize novel views along continuous trajectories (details in Appendix A.1).

4.2 End-to-End Monocular Telepresence

Table 1 compares VideoPort against optimization-based (PERSONA), generative diffusion (Human3Diffusion, DISCO4D), kinematic (IDOL, LHM++), and volumetric feed-forward (MonoPort) baselines. VideoPort achieves the highest PSNR across all datasets and generally tops silhouette IoU (except for ActorsHQ), while operating continuously at 17.2 FPS after a brief one-time startup latency of $T_{\text{init}} = 0.81$ s.

Table 1: **Quantitative novel-view synthesis.** VideoPort is evaluated in its memory-free streaming mode, relying solely on coarse geometry and reference warp to achieve 17.2 FPS and robustness on in-the-wild monocular data. T_{init} denotes the one-time system initialization and buffering latency before continuous real-time streaming begins. Parametric rigs yield low FVD through rigid skinning that oversmooths loose garments as seen in Figure 5. Diffusion baselines incur prohibitive sampling or precomputation latency. Bold indicates best performance.

				4DAnyone					ActorsHQ					DNA-Rendering				
Method	$T_{\text{init}}\downarrow$	**FPS↑**	**Control**	PSNR↑	SSIM↑	LPIPS↓	IoU↑	FVD↓	PSNR↑	SSIM↑	LPIPS↓	IoU↑	FVD↓	PSNR↑	SSIM↑	LPIPS↓	IoU↑	FVD↓
Per-Scene Optimization																		
PERSONA	3 h	30.8	Replay	20.31	0.8803	0.1409	0.6497	162.9	22.32	0.9093	0.0984	0.7622	130.4	21.31	0.8982	0.1209	0.7014	84.6
Generative Diffusion																		
Human3Diffusion	21.30 s	0.05	Per Frame	21.59	0.8902	0.0951	0.7221	163.4	24.10	0.9229	**0.0648**	0.8612	126.3	22.17	0.9048	0.0857	0.7845	73.2
DISCO4D	22.7 m	50.8	Trajectory	20.00	0.8650	0.1610	0.5913	210.9	21.14	0.8918	0.1283	0.6910	223.6	20.70	0.8768	0.1460	0.6590	109.4
Parametric Kinematics																		
IDOL	0.55 s	**266.6**	SMPL Rig	20.87	0.8782	0.1319	0.6813	199.4	22.95	0.9106	0.0950	0.7856	128.3	22.14	0.8977	0.1132	0.7430	80.9
LHM++	0.90 s	51.8	SMPL Rig	21.25	0.8868	0.1219	0.6818	128.5	22.95	0.9145	0.0866	0.7820	**100.4**	22.32	0.9025	0.1040	0.7397	**56.6**
Generalizable Feed-Forward																		
MonoPort	**0.093 s**	10.7	Stream	22.49	0.8999	**0.0896**	0.7550	**71.2**	23.94	**0.9235**	0.0665	**0.8645**	140.1	22.69	0.9105	**0.0835**	0.8099	63.8
VideoPort (ours)	0.81 s	17.2	Stream	**24.11**	**0.9096**	0.1086	**0.7892**	199.5	**24.31**	0.9234	0.0895	0.8051	111.9	**24.71**	**0.9227**	0.0910	**0.8172**	97.9

4D consistency vs. per-frame sharpness. Generative baselines like Human3Diffusion achieve lower LPIPS on static frames (0.0648 on ActorsHQ) but suffer from extrinsic trajectory drift and severe latency (0.05 FPS). Kinematic models (IDOL, LHM++) render rapidly, but rigid SMPL skinning fails on dynamic clothing, causing geometric tearing. In contrast, VideoPort prioritizes 4D geometric adherence: synthesized views strictly follow user extrinsics without pose drift (Figure 4). While fast-motion sharpness is occasionally bounded by monocular blur, cross-view and temporal fidelity remain consistent, as seen from our user-study A.10 from an unbiased sample.

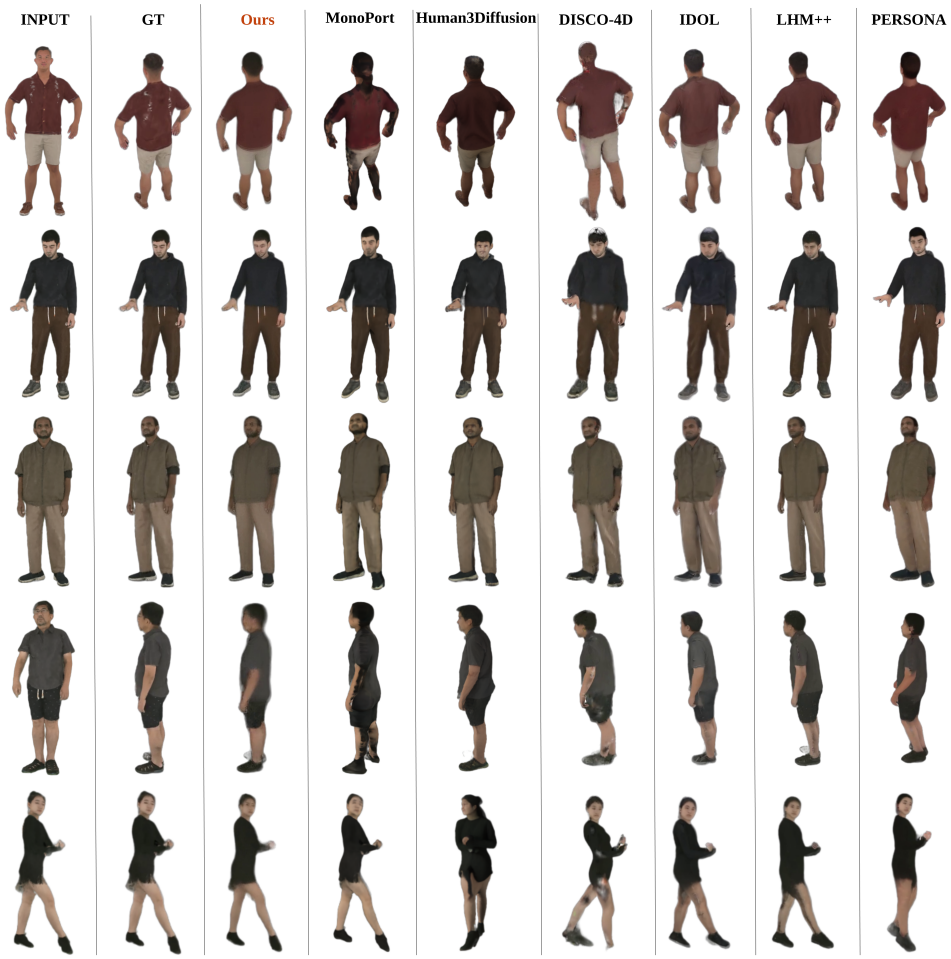


Figure 4: **Qualitative novel-view comparison.** Synthesized views at matched target viewpoints. Competing methods exhibit camera-pose drift and identity degradation. VideoPort strictly follows camera conditioning while preserving fine details (extended visuals in Appendix A.9).

4.3 Feedforward 4D Geometric Substrate

Surface accuracy and temporal tracking. Table 2 compares our unrefined geometric predictor against ground-truth dynamic gaussians. It achieves the lowest Chamfer Distance across all benchmarks (outperforming MonoPort by 27-47%) and recovers non-rigid apparel boundaries missed by parametric models (0.338 vs. 0.195 F1@1 cm on DNA-Rendering). Ablated in Appendix A.3, skeleton conditioning lowers CD to 17.36 mm, while temporal velocity supervision reduces inter-frame point churn by 45.9%. This provides a temporally coherent condition, for the causal refiner during multi-minute streaming.

4.4 Causal Refinement and Long-Horizon Stability

Mitigating autoregressive drift via DSR. In continuous streaming, autoregressive conditioning on self-generated history causes compounding drift as exposure bias pushes the model out of its training distribution. Table 3 evaluates closed-loop refinement to measure this decay, with a complete ablation in Appendix Table 9. Standard teacher-forced training collapses to 22.98 dB PSNR. Deep Self-Rollout (DSR) supervision improves PSNR by +2.53 dB over teacher forcing and +3.59 dB over DMD distillation. Backpropagating gradients through unrolled KV states yields modest gains, demonstrating that the refiner learns to actively attenuate prior hallucinations rather than blindly amplifying them as the temporal context window shifts.

Table 2: **Coarse geometric reconstruction against ground-truth surfaces.** Symmetric CD is in mm; F1 thresholds are 1 and 3 cm. Values represent per-sequence medians.

	4DAnyone			ActorsHQ			DNA-Rendering		
Method	CD↓	F1@1↑	F1@3↑	CD↓	F1@1↑	F1@3↑	CD↓	F1@1↑	F1@3↑
MonoPort	42.61	0.144	0.491	25.65	0.181	0.679	37.39	0.145	0.513
LHM++	39.55	0.169	0.558	28.29	0.215	0.667	29.97	0.195	0.657
Human3Diffusion	40.57	0.189	0.581	34.49	0.289	0.753	34.64	0.221	0.643
IDOL	42.97	0.140	0.503	28.75	0.188	0.641	32.21	0.167	0.604
PERSONA	51.00	0.123	0.461	37.24	0.178	0.612	48.46	0.141	0.507
DISCO4D	47.91	0.151	0.513	33.50	0.211	0.649	51.32	0.185	0.571
Ours	**24.84**	**0.270**	**0.745**	**18.65**	**0.364**	**0.812**	**19.77**	**0.338**	**0.808**

Impact of surface-aligned appearance memory. While baseline DSR achieves 25.51 dB, naive 2D reference warping introduces severe seam artifacts that degrade PSNR to 25.13 dB. In contrast, training with our surface-aligned 4D memory teacher boosts the oracle score to 26.93 dB. Crucially, disabling memory at inference to guarantee 17.2 FPS streaming yields a highly competitive 26.87 dB. This proves that rather than relying on test-time retrieval, the refiner successfully internalizes the teacher's appearance distribution, enabling robust, memory-free deployment.

Table 3: **Refinement ablation.** Evaluated on closed-loop rollouts with self-generated history. Differentiable Replay Differentiable history encoding provides a modest improvement. Unlike Table 1, "Full" enables inference-time surface memory via studio-tracked correspondences to isolate its upper-bound contribution. Bold denotes best causal result.

Configuration	**PSNR↑**	**SSIM↑**	**LPIPS↓**
Offline Reference Target			
Bidirectional Diffusion (Upper Bound)	27.30	0.9522	0.0434
Causal Training Paradigms			
Teacher-Forced Causal Baseline	22.98	0.9234	0.0914
DMD Distillation	21.92	0.9192	0.0923
DSR (Detached History K/V)	25.45	0.9349	0.0789
DSR (Differentiable Replay)	25.51	0.9390	0.0784
Appearance Conditioning Variants			
DSR + Direct Reference Warping	25.13	0.9347	0.0881
VideoPort Full (Memory-Free)	26.87	0.9450	0.0778
VideoPort Full (DSR + Memory)	**26.93**	**0.9456**	**0.0769**

Long horizon streaming evaluation. We additionally evaluate two-minute novel-view rollouts for two actors (Figure 12, Figure 9). The target camera follows forward, reverse, and repeated forward passes, yielding mean PSNR values of 26.35 and 24.25 dB, respectively. Quality varies with viewpoint and recovers on return passes, providing evidence of sustained reconstruction quality over the evaluated duration.

5 Conclusion

We present VideoPort, a framework for continuous monocular novel-view synthesis of human performances. The system combines feedforward geometric reconstruction, surface-aligned appearance retrieval, and causal video refinement. Deep self-rollout supervision trains the refiner against paired performance targets under its own generated-history errors to preserve appearance and motion over extended generation. Consequently, VideoPort bridges the gap between static generative fidelity and dynamic streaming viability. By anchoring causal diffusion to a metric 4D substrate, our framework circumvents the catastrophic autoregressive drift plaguing standard video models, achieving temporally stable, realistic telepresence over multi-minute rollouts.

VideoPort remains fundamentally constrained by the fidelity of its predicted geometry and temporal associations. Inaccurate geometry misaligns texture, while motion blur and coarse surfaces compound to degrade rendering sharpness. Furthermore, heavy reliance on geometric reprojection introduces a domain imbalance between high-detail observed surfaces and unobserved generative completions. A promising direction is to assimilate the generative coverage of multi-step vision foundation models into our causal framework, shifting toward unified full-body synthesis without sacrificing streaming efficiency. Enhancing geometric robustness under rapid deformation, resolving blur-induced misalignment, and reducing latency remain critical for practical live telepresence.

AI USE STATEMENT

In this work, we used generative AI tools for grammar and spelling checking the paper. We have not used generative AI tools for enhancing visual results. Additionally, we used generative AI tools for some coding tasks. We have reviewed all AI-assisted work. LLM-generated code was verified and tested for correctness. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

ETHICS STATEMENT

VideoPort is intended to support consensual human telepresence. Because it reconstructs identifiable appearance and synthesizes novel views, potential misuse includes impersonation and the creation of misleading depictions without a person's permission. Deployment should require informed consent for capture, synthesis, and sharing, clearly disclose generative refinement, and protect input recordings and stored appearance references from unauthorized access.

The reconstructed videos may contain inferred details that differ from the actual performance, particularly in occluded regions or under rapid deformation. Outputs should therefore not be treated as exact visual records. Performance on selected benchmarks also does not establish uniform accuracy across demographic groups, body shapes, clothing, or capture conditions; these limitations should inform deployment and further evaluation.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility and further research, we will open-source the complete training and inference code, model checkpoints, and the training dataset used in this work. We will also provide the configurations and implementation details necessary to reproduce our experiments and reported results. Together with the methodological and experimental details provided in the main paper and appendix, these resources are intended to enable independent reproduction and further development of our method.

REFERENCES

Marcel C Buehler, Ye Yuan, Xueting Li, Yangyi Huang, Koki Nagano, and Umar Iqbal. Dream, lift, animate: From single images to animatable gaussian avatars. In *2026 International Conference on 3D Vision (3DV)*, pp. 1414–1425. IEEE, 2026.

Wei Cheng, Ruixiang Chen, Wanqi Yin, Siming Fan, Keyu Chen, Honglin He, Huiwen Luo, Zhongang Cai, Jingbo Wang, Yang Gao, Zhengming Yu, Zhengyu Lin, Daxuan Ren, Lei Yang, Ziwei Liu, Chen Change Loy, Chen Qian, Wayne Wu, Dahua Lin, Bo Dai, and Kwan-Yee Lin. Dna-rendering: A diverse neural actor repository for high-fidelity human-centric rendering. *arXiv preprint*, arXiv:2307.10173, 2023.

Justin Cui, Jie Wu, Ming Li, Tao Yang, Xiaojie Li, Rui Wang, Andrew Bai, Yuanhao Ban, and Cho-Jui Hsieh. Self-forcing++: Towards minute-scale high-quality video generation, 2025. URL https://arxiv.org/abs/2510.02283.

Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. *arXiv preprint arXiv:2212.08051*, 2022.

Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, Eli VanderBilt, Aniruddha Kembhavi, Carl Vondrick, Georgia Gkioxari, Kiana Ehsani, Ludwig Schmidt, and Ali Farhadi. Objaverse-xl: A universe of 10m+ 3d objects. *arXiv preprint arXiv:2307.05663*, 2023.

Zijian Dong, Longteng Duan, Jie Song, Michael J Black, and Andreas Geiger. Moga: 3d generative avatar prior for monocular gaussian avatar reconstruction. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 13304–13314. IEEE, 2025.

Tianrui Feng, Zhi Li, Shuo Yang, Haocheng Xi, Muyang Li, Xiuyu Li, Lvmin Zhang, Keting Yang, Kelly Peng, Song Han, Maneesh Agrawala, Kurt Keutzer, Akio Kodaira, and Chenfeng Xu. Streamdiffusionv2: A streaming system for dynamic and interactive video generation. *Proceedings of Machine Learning and Systems*, 8:1656–1669, 2026.

Yuwei Guo, Ceyuan Yang, Hao He, Yang Zhao, Meng Wei, Zhenheng Yang, Weilin Huang, and Dahua Lin. End-to-end training for autoregressive video diffusion via self-resampling. *arXiv preprint arXiv:2512.15702*, 2025.

Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, et al. Ltx-video: Realtime video latent diffusion. *arXiv preprint arXiv:2501.00103*, 2024.

Xu He, Xiaoyu Li, Di Kang, Jiangnan Ye, Chaopeng Zhang, Liyang Chen, Xiangjun Gao, Han Zhang, Zhiyong Wu, and Haolin Zhuang. Magicman: Generative novel view synthesis of humans with 3d-aware diffusion and iterative refinement. *arXiv preprint arXiv:2408.14211*, 2024.

Xun Huang, Zhengqi Li, Guande He, Mingyuan Zhou, and Eli Shechtman. Self forcing: Bridging the train-test gap in autoregressive video diffusion. *arXiv preprint arXiv:2506.08009*, 2025.

Mustafa Işık, Martin Rünz, Markos Georgopoulos, Taras Khakhulin, Jonathan Starck, Lourdes Agapito, and Matthias Nießner. Humanrf: High-fidelity neural radiance fields for humans in motion. *ACM Transactions on Graphics (TOG)*, 42(4):1–12, 2023. doi: 10.1145/3592415. URL https://doi.org/10.1145/3592415.

Haian Jin, Hanwen Jiang, Hao Tan, Kai Zhang, Sai Bi, Tianyuan Zhang, Fujun Luan, Noah Snavely, and Zexiang Xu. Lvsm: A large view synthesis model with minimal 3d inductive bias. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=QQBPWtvtcn.

Yudong Jin, Tao Xie, Qihang Zhang, Zehong Shen, Zhen Xu, Yujun Shen, Hujun Bao, Xiaowei Zhou, and Yinghao Xu. 4danyone: Create anyone in 4d from a casual monocular video. *arXiv preprint arXiv:2608.20335*, 2026.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3d gaussian splatting for real-time radiance field rendering. *ACM Transactions on Graphics*, 42(4), 2023.

Omar El Khalifi, Thomas Rossi, Oscar Fossey, Thibault Fouque, Ulysse Mizrahi, Philip Torr, Ivan Laptev, Fabio Pizzati, and Baptiste Bellot-Gurlet. Actcam: Zero-shot joint camera and 3d motion control for video generation, 2026. URL https://arxiv.org/abs/2605.06667.

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024.

Jason Lawrence, Ryan Overbeck, Todd Prives, Tommy Fortes, Nikki Roth, and Brett Newman. Project starline: A high-fidelity telepresence system. In *ACM SIGGRAPH 2024 Emerging Technologies*, pp. 1–2. 2024.

Junxuan Li, Rawal Khirodkar, Chengan He, Zhongshi Jiang, Giljoo Nam, Lingchen Yang, Jihyun Lee, Egor Zakharov, Zhaoen Su, Rinat Abdrashitov, Yuan Dong, Julieta Martinez, Kai Li, Qingyang Tan, Takaaki Shiratori, Matthew Hu, Peihong Guo, Xuhua Huang, Ariyan Zarei, Marco Pesavento, Yichen Xu, He Wen, Teng Deng, Wyatt Borsos, Anjali Thakrar, Jean-Charles Bazin, Carsten Stoll, Ginés Hidalgo, James Booth, Lucy Wang, Xiaowen Ma, Yu Rong, Sairanjith Thalanki, Chen Cao, Christian Häne, Abhishek Kar, Sofien Bouaziz, Jason Saragih, Yaser Sheikh, and Shunsuke Saito. Large-scale codec avatars: The unreasonable effectiveness of large-scale avatar pretraining, 2026. URL https://arxiv.org/abs/2604.02320.

Peng Li, Wangguandong Zheng, Yuan Liu, Tao Yu, Yangguang Li, Xingqun Qi, Mengfei Li, Xiaowei Chi, Siyu Xia, Wei Xue, et al. Pshuman: Photorealistic single-view human reconstruction using cross-scale diffusion. *arXiv preprint arXiv:2409.10141*, 2024.

Ruilong Li, Yuliang Xiu, Shunsuke Saito, Zeng Huang, Kyle Olszewski, and Hao Li. Monocular real-time volumetric performance capture. In *European Conference on Computer Vision*, pp. 49–67. Springer, 2020.

Fangyu Lin, Yingdong Hu, Zhening Liu, Yufan Zhuang, Zehong Lin, and Jun Zhang. Mon3tr: Monocular 3d telepresence with pre-built gaussian avatars as amortization. *arXiv preprint arXiv:2601.07518*, 2026.

Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3d object. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9264–9275. IEEE, 2023.

Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9970–9980. IEEE, 2024.

Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *3DV*, 2024.

Yuexiao Ma, Xuzhe Zheng, Jing Xu, Xiwei Xu, Feng Ling, Xiawu Zheng, Huafeng Kuang, Huixia Li, Xing Wang, Xuefeng Xiao, Fei Chao, and Rongrong Ji. Flow caching for autoregressive video generation, 2026. URL https://arxiv.org/abs/2602.10825.

Guy M. Morton. A computer oriented geodetic data base and a new technique in file sequencing. Technical report, IBM Ltd., 1966.

Sergio Orts-Escolano, Christoph Rhemann, Sean Fanello, Wayne Chang, Adarsh Kowdle, Yury Degtyarev, David Kim, Philip L Davidson, Sameh Khamis, Mingsong Dou, et al. Holoportation: Virtual 3d teleportation in real-time. In *Proceedings of the 29th annual symposium on user interface software and technology*, pp. 741–754, 2016.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4172–4182. IEEE, 2023.

Lingteng Qiu, Xiaodong Gu, Peihao Li, Qi Zuo, Weichao Shen, Junfei Zhang, Kejie Qiu, Weihao Yuan, Guanying Chen, Zilong Dong, et al. Lhm: Large animatable human reconstruction model for single image to 3d in seconds. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 14184–14194. IEEE, 2025a.

Lingteng Qiu, Peihao Li, Heyuan Li, Qi Zuo, Xiaodong Gu, Yuan Dong, Weihao Yuan, Rui Peng, Siyu Zhu, Xiaoguang Han, et al. Lhm++: An efficient large human reconstruction model for pose-free images to 3d. *arXiv preprint arXiv:2506.13766*, 2025b.

René Ranftl, Katrin Lasinger, David Hafner, Konrad Schindler, and Vladlen Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2020.

Gernot Riegler and Vladlen Koltun. Free view synthesis. In *European conference on computer vision*, pp. 623–640. Springer, 2020.

Gernot Riegler and Vladlen Koltun. Stable view synthesis. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12216–12225, 2021.

Shunsuke Saito, Zeng Huang, Ryota Natsume, Shigeo Morishima, Angjoo Kanazawa, and Hao Li. Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2304–2314, 2019.

Shunsuke Saito, Tomas Simon, Jason Saragih, and Hanbyul Joo. Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 84–93, 2020.

Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3, 2025. URL https://arxiv.org/abs/2508.10104.

Jiaxiang Tang, Zhaoxi Chen, Xiaokang Chen, Tengfei Wang, Gang Zeng, and Ziwei Liu. Lgm: Large multi-view gaussian model for high-resolution 3d content creation. In *European Conference on Computer Vision*, pp. 1–18. Springer, 2025.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Peng Wang and Yichun Shi. Imagedream: Image-prompt multi-view diffusion for 3d generation. *arXiv preprint arXiv:2312.02201*, 2023.

Jay Zhangjie Wu, Yuxuan Zhang, Haithem Turki, Xuanchi Ren, Jun Gao, Mike Zheng Shou, Sanja Fidler, Zan Gojcic, and Huan Ling. Difix3d+: Improving 3d reconstructions with single-step diffusion models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 26024–26035. IEEE, 2025.

Kailu Wu, Fangfu Liu, Zhihan Cai, Runjie Yan, Hanyang Wang, Yating Hu, Yueqi Duan, and Kaisheng Ma. Unique3d: High-quality and efficient 3d mesh generation from a single image. *Advances in Neural Information Processing Systems*, 37:125116–125141, 2024a.

Xiaoyang Wu, Li Jiang, Peng-Shuai Wang, Zhijian Liu, Xihui Liu, Yu Qiao, Wanli Ouyang, Tong He, and Hengshuang Zhao. Point transformer v3: Simpler, faster, stronger. In *CVPR*, 2024b.

Guangxuan Xiao, Jiaming Tang, Jingwei Zuo, Junxian Guo, Shang Yang, Haotian Tang, Yao Fu, and Song Han. Duoattention: Efficient long-context llm inference with retrieval and streaming heads. In *International Conference on Learning Representations*, volume 2025, pp. 37228–37253, 2025.

Yuliang Xiu, Jinlong Yang, Dimitrios Tzionas, and Michael J Black. Icon: Implicit clothed humans obtained from normals. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13286–13296. IEEE, 2022.

Yinghao Xu, Zifan Shi, Wang Yifan, Hansheng Chen, Ceyuan Yang, Sida Peng, Yujun Shen, and Gordon Wetzstein. Grm: Large gaussian reconstruction model for efficient 3d reconstruction and generation. In *European Conference on Computer Vision*, pp. 1–20. Springer, 2024.

Yuxuan Xue, Xianghui Xie, Riccardo Marin, and Gerard Pons-Moll. Human 3diffusion: Realistic avatar creation via explicit 3d consistent diffusion models. *arXiv preprint arXiv:2406.08475*, 2024.

Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis, 2024. URL https://arxiv.org/abs/2409.02048.

Haiyu Zhang, Xinyuan Chen, Yaohui Wang, Xihui Liu, Yunhong Wang, and Yu Qiao. 4diffusion: Multi-view video diffusion model for 4d generation. *Advances in Neural Information Processing Systems*, 37:15272–15295, 2024.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3813–3824. IEEE, 2023.

Min Zhao, Hongzhou Zhu, Kaiwen Zheng, Zihan Zhou, Bokai Yan, Xinyuan Li, Xiao Yang, Chongxuan Li, and Jun Zhu. Causal forcing++: Scalable few-step autoregressive diffusion distillation for real-time interactive video generation. *arXiv preprint arXiv:2605.15141*, 2026.

Hongzhou Zhu, Min Zhao, Guande He, Hang Su, Chongxuan Li, and Jun Zhu. Causal forcing: Autoregressive diffusion distillation done right for high-quality real-time interactive video generation. *arXiv preprint arXiv:2602.02214*, 2026.

Junhao Zhuang, Shiyi Zhang, Yuxuan Bian, Yaowei Li, Yawen Luo, Yijun Liu, Weiyang Jin, Songchun Zhang, Xianglong He, Xuying Zhang, et al. Self gradient forcing: Native long video extrapolation. *arXiv preprint arXiv:2607.20368*, 2026.

Yiyu Zhuang, Jiaxi Lv, Hao Wen, Qing Shuai, Ailing Zeng, Hao Zhu, Shifeng Chen, Yujiu Yang, Xun Cao, and Wei Liu. Idol: Instant photorealistic 3d human creation from a single image. *arXiv preprint arXiv:2412.14963*, 2024.

Laurent Zwald and Sophie Lambert-Lacroix. The berhu penalty and the grouped effect, 2012. URL https://arxiv.org/abs/1207.6868.

A Appendix

A.1 Dataset Augmentation.

To obtain flexible supervision from these recordings, we fit dynamic Gaussian representations Luiten et al. (2024) to DNA-Rendering sequences. These representations support user-defined camera trajectories, temporally aligned multiview rendering, and geometric corruptions that simulate errors in our coarse predictor.

We define a set of $2{,}000$ camera trajectories, including static, orbital-arc, return, oscillating, spline, and dolly motions. We additionally introduce five augmentations: ray-aligned depth jitter, low-frequency depth deformation with global offset and scale changes, geometric over-smoothing, view-angle-dependent degradation, and temporal wobble. For each action sequence, we sample 50 trajectory-augmentation combinations. The resulting dataset contains 940 performance sequences, each comprising 149 frames, yielding $46{,}520$ training videos and $1{,}500$ held-out videos. The refiner is trained on the augmented dataset, while the point-cloud estimator is trained on the non augmented dataset.

A.2 System overview.

At deployment, a hardware frame grabber continuously ingests the live monocular video stream. The end-to-end system is distributed across 7 workstation-grade NVIDIA RTX 6000 Ada GPUs—hardware substantially below current datacenter state-of-the-art accelerators (e.g., H100 or B200). Individual pipeline components run asynchronously at differing latencies: live segmentation and feedforward coarse 3D reconstruction operate at a faster rate than the 14B-parameter diffusion refiner, which runs an accelerated 4-step flow-matching schedule on 6 GPUs in paralllel. After a brief one-time system initialization and buffer fill ($T_{\text{init}} = 0.81\,\text{s}$), pipelining these heterogeneous stages yields an aggregate, continuous end-to-end throughput of 17.2 FPS at 512×512 resolution. The headline quality and speed use the same configuration, report steady-state latency, and the same hardware. This allows for the user to immerse himself/herself in the demo and experience a live novel view synthesis of their performance with a stable latency of around 17-18 FPS (a minimal drop to 17.2 FPS), guided by a simple mouse movement in our rendering interface.

A.3 Single View 3D Human Reconstruction in Real-time

Observation encoding. The model reconstructs each frame independently from an RGB image I_t, a foreground mask M_t, and camera calibration. A frozen DINOv3 ViT-B/16 encoder processes the 512×512 image once:

$$(g_t, F_t) = \text{Enc}_{\text{DINOv3}}(I_t). \tag{10}$$

The class token g_t summarizes the image. Spatial features F_t from blocks 3, 6, 9, and 12 feed pretrained, frozen depth and joint heads. Their surface and pose estimates, together with g_t, condition complete-body reconstruction. Training images and masks are rendered from ground-truth 4D Gaussians while deployment uses a photograph and its supplied mask.

Depth prediction and calibration. We use a fixed, level camera $0.90\,\text{m}$ above the floor and $2.825\,\text{m}$ from the subject axis, with a horizontal field of view of $43.3°$. Its intrinsics are $\mathbf{K}$, and $\mathbf{E}_t$ maps world coordinates to camera coordinates. A similarity transform fitted to shared camera centers aligns the training Gaussians, joints, and cameras in one metric coordinate system.

A DPT-style decoder predicts camera-space XYZ coordinates. All three receive supervision; reconstruction uses depth. We calibrate its scale using the known camera-to-aim distance $d_{\text{aim},t}$. The visible surface is generally closer than the aim point inside the body, so we subtract an empirical $25\,\text{mm}$ offset retained from depth-model development. A learned adjustment handles frame-dependent differences:

$$\begin{aligned} s_t &= \text{clip}\left(\frac{d_{\text{aim},t} - 0.025\,\text{m}}{\text{median}_{p:M_t(p)=1}\, \tilde{D}_t(p)},\, 0.5,\, 2 \right), \\ D_t^{\text{cal}}(p) &= s_t \tilde{D}_t(p), \qquad \hat{D}_t(p) = e^{\delta_t} D_t^{\text{cal}}(p). \end{aligned} \tag{11}$$

Table 4: Main loss weights of the observation module after warm-up. Smaller auxiliary terms are omitted: edge-aware smoothness (0.05), side-view coverage (0.5), joint visibility and joint-surface coupling (0.1 each), the auxiliary camera heads (1-2), and the training-only signed-distance field with its six companion terms.

Observation loss	Weight
Depth prediction	
Point map: berHu	0.85
Point map: normal consistency	0.15
Log-scale correction	1
Visible-surface Chamfer	1
Side-view depth	1
Depth-error motion consistency	2
Joint prediction	
Heatmap KL divergence	1
Joint depth / mean 3D error	1 each
Joint-motion consistency	1
Side-view joint reprojection	0.5
Joint-group multipliers	
Hands / feet / body / face	2 / 1.5 / 1 / 0.1

Here, p indexes image pixels and $\tilde{D}_t$ is the raw depth prediction. An MLP reads pooled image features and camera parameters to predict $\delta_t = 0.08 \tanh(a_t)$, where a_t is its scalar output. This allows approximately 8% scaling; zero output initialization leaves depth unchanged initially.

For supervision, we find the scalar $c_t^\star$ that minimizes absolute depth error after multiplying every foreground value in D_t^{cal} by that scalar. We train δ_t toward $\log c_t^\star$, since its exponential is the applied scale. This target is recomputed each step but held fixed during backpropagation. The smooth-ℓ_1 loss uses threshold 0.01.

Depth supervision. Raw XYZ predictions receive berHu Zwald & Lambert-Lacroix (2012), surface-normal, and edge-aware smoothness losses. The corrected depth receives three geometric constraints: Chamfer distance to the visible ground-truth surface, agreement of rendered depth and coverage at $\pm 60^\circ$, and consistency of depth errors along ground-truth motion. The front-surface Chamfer uses a 1 cm Huber threshold. Table 4 lists the loss weights.

An auxiliary field predicts signed distances around the surface using a four-layer, width-128 MLP. It samples $8{,}192$ foreground rays at 32 positions, spanning 0.25 m before to 0.60 m behind the prediction. Front, back, and side renders define targets truncated at 8 cm. We supervise distances, changes along rays, and the extracted front surface, including normal and smoothness terms. Surface matching uses 98%-trimmed Chamfer and squared-cost Sinkhorn on $4{,}096$ points per cloud. Gradients through sample positions update depth. An auxiliary camera head predicts focal lengths, principal point, and camera-up direction. Neither auxiliary head runs at inference.

Whole-body joint prediction. For each of 133 joints, the skeleton head predicts location scores $H_{t,j}$, depth offsets $Z_{t,j}$, and visibility logits on a 128×128 grid. A spatial softmax gives the location distribution:

$$P_{t,j}(p) = \frac{\exp(H_{t,j}(p))}{\sum_{p'} \exp(H_{t,j}(p'))}. \tag{12}$$

We express grid locations p in input-image pixels. Their probability-weighted mean gives the joint's image position; the same weights give its depth offset:

$$\begin{aligned} \hat{\mathbf{p}}_{t,j} &= \sum_p P_{t,j}(p)\, p, \\ \hat{z}_{t,j} &= z_{\mathrm{ref},t} + \sum_p P_{t,j}(p) Z_{t,j}(p), \\ \hat{\mathbf{X}}_{t,j} &= \mathbf{E}_t^{-1} \pi_{\mathbf{K}}^{-1}(\hat{\mathbf{p}}_{t,j}, \hat{z}_{t,j})\,. \end{aligned} \tag{13}$$

Here, $\pi_{\mathbf{K}}^{-1}$ back-projects pixels and depth, and $\mathbf{E}_t^{-1}$ maps the result to world coordinates. The reference $z_{\mathrm{ref},t}$ is the foreground median of the supplied depth map, detached to prevent joint losses changing depth.

Locations receive Gaussian heatmap supervision with standard deviation two bins; depths and 3D positions receive ℓ_1 supervision, including for occluded in-frame joints. Additional losses constrain visibility, motion, and side-view projections. Joint-surface coupling penalizes visible joints more than 2 cm in front of the surface. Table 4 gives weights and joint-group multipliers.

Reconstruction transformer. We back-project foreground pixels of the corrected depth map to form a visible point cloud:

$$\mathcal{C}_t = \left\{ \mathbf{E}_t^{-1} \pi_{\mathbf{K}}^{-1}\Big(p, \hat{D}_t(p)\Big) : M_t(p) = 1 \right\}. \tag{14}$$

where $M_t(p) = 1$ indicates that pixel p belongs to the person in frame t. We sample $15{,}000$ foreground pixels, allowing repeats, then select $1{,}024$ points by farthest-point sampling. Their coordinates and sine/cosine features at frequencies $2^0, \ldots, 2^7$ form the encoding ϕ. An MLP converts these encodings into 768-dimensional tokens:

$$h_t^{(0)} = \Big[Q \parallel \mathrm{MLP}_{\mathrm{point}}\Big(\phi(\mathcal{C}_t^{\mathrm{fps}})\Big) \Big]. \tag{15}$$

Here, $\mathcal{C}_t^{\mathrm{fps}}$ is the sampled cloud, Q contains 512 learned queries, and $\parallel$ concatenates tokens. Twelve blocks with 12 attention heads and MLP width 3,072 exchange information between queries and geometry. Predicted depths and joints are used during training to expose reconstruction to upstream errors.

Pose enters through a separate joint encoder:

$$e_{t,j} = \mathrm{MLP}_{\mathrm{joint}}\Big([\phi(\hat{\mathbf{X}}_{t,j}) \parallel v_{t,j}] \Big) + E_{\mathrm{id}}[j]. \tag{16}$$

Here, concatenation is along the feature dimension, $E_{\mathrm{id}}[j]$ is a learned joint-identity embedding, and $v_{t,j} = 1$ marks each predicted joint as valid. Visibility predictions and velocities are not inputs. Two transformer layers map $e_{t,j}$ to contextual joint features $\tilde{e}_{t,j}$. We pool these features and combine them with the image summary:

$$\begin{aligned} c_{\mathrm{skel},t} &= W_{\mathrm{pool}} \frac{1}{133} \sum_j \mathrm{LN}(\tilde{e}_{t,j}), \\ f_t &= W_{\mathrm{fuse}} [W_{\mathrm{img}} \,\mathrm{LN}(g_t) \parallel c_{\mathrm{skel},t}]. \end{aligned} \tag{17}$$

The W matrices are learned projections and LN denotes layer normalization. In each reconstruction block ℓ, f_t predicts query shifts $\beta_{\ell,t}$, scales $\gamma_{\ell,t}$, and residual gates $\alpha_{\ell,t}$. For query tokens q and cloud tokens c,

$$\begin{aligned} \tilde{q} &= (1 + \gamma_{\ell,t}) \odot \mathrm{LN}(q) + \beta_{\ell,t}, \\ [a_q \parallel a_c] &= \mathrm{MHSA}([\tilde{q} \parallel \mathrm{LN}_c(c)]), \\ q' &= q + \alpha_{\ell,t} \odot a_q, \qquad c' = c + a_c. \end{aligned} \tag{18}$$

Here, MHSA is multi-head self-attention, a_q, a_c are its output slices, and $\odot$ denotes elementwise multiplication. The MLP sublayer uses its own modulation parameters. This adaLN-Zero conditioning is zero-initialized and applies only to queries.

Hierarchical point decoding. Each final query $u_{t,q}$ predicts an anchor, eight child points, and twelve points per child:

$$\begin{aligned} \mathbf{x}_{t,q}^{(0)} &= \tanh(\psi_0(u_{t,q})), \\ \mathbf{x}_{t,q,i}^{(1)} &= \mathbf{x}_{t,q}^{(0)} + r_1 \tanh\Big(\psi_1(f_{t,q,i}^{(1)})\Big), \\ \mathbf{x}_{t,q,i,j}^{(2)} &= \mathbf{x}_{t,q,i}^{(1)} + r_2 \tanh\Big(\psi_2(f_{t,q,i,j}^{(2)})\Big). \end{aligned} \tag{19}$$

The features $f^{(1)}, f^{(2)}$ combine parent features, learned child codes, and position embeddings; each ψ_ℓ is layer normalization followed by a linear projection. Anchors lie in $[-1, 1]^3$ m, and $r_1 = 0.35$ m and $r_2 = 0.06$ m bound offsets along each axis. This produces $512 \times 8 \times 12 = 49{,}152$ points.

Morton ordering Morton (1966), implemented after inspiration by Wu et al. (2024b), groups nearby parent positions into windows of 64 tokens, allowing local attention without attention over the entire cloud. The first level uses two width-768 blocks and also attends to front-cloud and joint tokens. The second uses one width-256 block and reads offsets to the eight nearest front points and four nearest joints. These additional reads use zero-initialized residual projections. After attention, we restore the original order. Thus, index k identifies the same query descendant in every frame, enabling the temporal supervision below. We denote its position by $\hat{\mathbf{x}}_{t,k}$.

Reconstruction supervision. Ground truth contains up to 40,960 Gaussian centers $\mathcal{P}_t^\star = \{\mathbf{x}_{t,m}^\star\}_m$, with the same centers tracked throughout each sequence. The losses constrain surface accuracy, coverage, local detail, and temporal motion.

Huber Chamfer $\mathcal{L}_{\mathrm{CD}}$ measures nearest-neighbor distances in both directions. We apply $h_\epsilon(d) = d^2/(2\epsilon)$ below $\epsilon = 1$ cm and $d - \epsilon/2$ otherwise, and double the ground-truth-to-prediction term to emphasize coverage. Sinkhorn divergence $\mathcal{L}_{\mathrm{S}}$ compares the clouds as uniform point distributions, discouraging many predictions from accumulating on a few target points. It uses Euclidean cost, blur $\max(0.005, 0.1 \cdot 0.995^n)$ m, and annealing ratio $\min(0.95, 1 - 0.5 \cdot 0.995^n)$ at optimizer step n.

Because these losses decrease at different rates, we adjust the Sinkhorn weight every 200 steps:

$$\lambda_{\mathrm{S}}^\star = \lambda_{\mathrm{CD}} \frac{\bar{C}}{\bar{S}} \frac{1 - T}{T}. \tag{20}$$

Here, $\bar{C}$ and $\bar{S}$ are the unweighted losses averaged over the preceding interval and all devices; T is the desired Chamfer share of their weighted sum. Each update changes the current weight by at most 20%, within $[0.02, 20]$. Table 5 gives the targets. The controller regulates loss values.

The part loss $\mathcal{L}_{\mathrm{part}}$ supervises 15 anatomical regions separately. Points take the nearest ground-truth bone label; head-region points within 8 cm of a facial joint receive the face label. Each region uses unsquared Chamfer with doubled coverage, then contributes according to Table 4. Regions with fewer than 16 points in either cloud are excluded, leaving entirely missing parts unsupervised by this term.

Rendered-depth losses constrain other viewpoints. We render fixed 8 mm splats at 512×512 from four orbit views spaced by 90° and twelve distinct cameras selected from the 48-camera rig. The group means are $\mathcal{L}_{\mathrm{orbit}}$ and $\mathcal{L}_{\mathrm{rig}}$: smooth-$\ell_1$ depth errors with a 5 cm threshold, evaluated on ground-truth foreground pixels with valid predicted depth.

Two losses constrain the coarse point distribution on a 5 cm grid. $\mathcal{L}_{\mathrm{empty}}$ measures distance to the nearest cube containing ground-truth points. $\mathcal{L}_{\mathrm{mass}}$ compares the fraction of points in each cube using half the summed absolute differences. Trilinear assignment makes these fractions differentiable; out-of-grid points enter an extra bin. Grid bounds exclude ground-truth outliers beyond 2 m from the frame median and add a 0.5 m margin.

Temporal flow supervision. The fixed output order lets us supervise each point's motion between frames. For output point k, we find the nearest tracked ground-truth center in the preceding frame:

$$i_t(k) = \arg\min_m \|\hat{\mathbf{x}}_{t-1,k} - \mathbf{x}_{t-1,m}^\star\|_2. \tag{21}$$

We accept this match only when the distance is below 5 cm. Let $\mathcal{V}$ contain all accepted matches (t, k) in the batch, and let $\Delta\mathbf{x}_{t-1,m}^\star = \mathbf{x}_{t,m}^\star - \mathbf{x}_{t-1,m}^\star$ be the tracked displacement. For $|\mathcal{V}| > 0$, the loss is

$$\mathcal{L}_{\mathrm{flow}} = \frac{1}{|\mathcal{V}|} \sum_{(t,k)\in\mathcal{V}} \left\| \hat{\mathbf{x}}_{t,k} - \hat{\mathbf{x}}_{t-1,k} - \Delta\mathbf{x}_{t-1,i_t(k)}^\star \right\|_2. \tag{22}$$

Both frames use the same camera, are predicted independently, and receive gradients. Temporal information is needed only during training.

Table 5: Settings used during reconstruction training; 1k denotes 1,000 optimizer steps. $\Delta\phi$ is camera azimuth relative to the input view. Warm-up ramps multiply the listed loss weights. The offset Chamfer weight increases from 0.05 to 0.25 over its first 500 steps. The effective batch changes from $7 \times 4 \times 2$ to $5 \times 4 \times 3$ (GPUs, pairs per GPU, accumulation steps).

Point-cloud reconstruction - 35k steps, 903 sequences				
Setting	0-8k	8-17k	17-25k	From 25k
Chamfer weight (Huber, $\delta = 10$ mm, coverage $\times 2$)	1.0	1.0	1.0	1.5
Transport: target chamfer share	$0.73 \rightarrow 0.40$	0.40	$0.40 \rightarrow 0.77$	0.81
Slot-flow weight	1.0	2.0	2.0	2.0
Part-chamfer weight	0.3	0.3	0.3	0.3
Rendered-depth weights (4 orbit / 12 studio-rig)	0.5/0.25	0.5/0.25	0.5/0.25	0.5/0.25
Voxel free-space weight (mass part $\times 0.05$)	0.1	0.1	0.1	0.1
Effective batch (frame pairs)	56	56	60	60
Appearance head - 6k steps, reconstruction frozen at 35k				
Setting	0-0.5k	0.5-2k	2-6k	From 6k
Pixel / perceptual weights	$\rightarrow 0.5$ / $\rightarrow 0.4$	0.5/0.4	0.5/0.4	0.5/0.4
Spill / sphere-prior weights	0.1 / $\rightarrow 0.001$	0.1/0.001	0.1/0.001	0.1/0.001
Opacity floor / radius cap (mm)	0.5/12	0.5/12	0.5/12	0.5/12
Elongation cap / back-view weight	8/$\times 2$	8/$\times 2$	8/$\times 2$	8/$\times 2$
Silhouette matte filter	off	off	on	on
Center offsets (ray / tangent, mm)	off	off	off	15/10
Effective batch (frames)	96	96	96	96

The geometric objective combines the terms defined above:

$$\begin{aligned}\mathcal{L}_{\text{recon}} = \lambda_{\text{CD}}\mathcal{L}_{\text{CD}} + \lambda_{\text{S}}(t)\,\mathcal{L}_{\text{S}} + \lambda_{\text{flow}}\mathcal{L}_{\text{flow}} \\ + \lambda_{\text{part}}\mathcal{L}_{\text{part}} + \lambda_{\text{orbit}}\mathcal{L}_{\text{orbit}} + \lambda_{\text{rig}}\mathcal{L}_{\text{rig}} \\ + \lambda_{\text{empty}}\mathcal{L}_{\text{empty}} + \lambda_{\text{mass}}\mathcal{L}_{\text{mass}}.\end{aligned} \tag{23}$$

At the end of training $\lambda_{\text{CD}} = 1.5$, $\lambda_{\text{flow}} = 2$, $\lambda_{\text{part}} = 0.3$, $\lambda_{\text{orbit}} = 0.5$, $\lambda_{\text{rig}} = 0.25$, $\lambda_{\text{empty}} = 0.1$ and $\lambda_{\text{mass}} = 5 \times 10^{-3}$. The transport weight is the only one never set by hand: every 200 steps it is moved toward the value that holds the chamfer at a prescribed share T of the weighted pair,

$$\lambda_{\text{S}}^{\star} = \lambda_{\text{CD}}\,\frac{\bar{\mathcal{L}}_{\text{CD}}}{\bar{\mathcal{L}}_{\text{S}}}\,\frac{1-T}{T}, \qquad \lambda_{\text{S}} \leftarrow \text{clip}\left(\lambda_{\text{S}}^{\star},\, 0.8\lambda_{\text{S}},\, 1.2\lambda_{\text{S}}\right), \tag{24}$$

with $\bar{\mathcal{L}}$ the means of the raw terms over the last 200 steps and T taking the values 0.73, 0.40 and 0.81 in three phases beginning at steps 0, 3,500 and 21,000. All remaining weights are ramped linearly from zero over their opening steps - flow and part over steps 200-1,000, the four rendered-depth and voxel weights over 1,000-2,000 - and are constant thereafter; λ_{CD} and λ_{flow} are raised once each during training (Table 5).

Gaussian attributes and appearance. A per-point head predicts Gaussian color, opacity, rotation, size, and center offsets from decoded features and projected RGB, convolutional, and DINO features. Visibility compares point depth with the front surface and cloud z-buffer (20 mm tolerance, 5 mm softness). Visible colors are corrected from sampled pixels; hidden colors use point features and the image summary. Projection coordinates and visibility weights are detached.

The Gaussian center and axis scales are

$$\boldsymbol{\mu}_{t,k} = \hat{\mathbf{x}}_{t,k} + \Delta\mathbf{x}_{t,k}, \tag{25}$$

$$\mathbf{s}_{xt,k} = \varrho_{t,k} \begin{bmatrix} e^{a_{t,k}/2} \\ e^{-a_{t,k}/2} \\ e^{-b_{t,k}} \end{bmatrix}. \tag{26}$$

Here, ϱ controls size, a controls elongation, and b controls thickness. Final bounds are $\varrho \in [1, 4]$ mm, $|a| \leq \log 3$, $b \in [-0.1, \log 30]$, and opacity at least 0.5. Center offsets are limited to 15 mm along the viewing ray and 5 mm along each transverse axis. Offsets affect rendering only.

We supervise 1024×1024 renderings using four ground-truth renders and photographs from the same twelve rig cameras used for depth supervision.

The image objective is

$$\mathcal{L}_{\text{img}} = \lambda_{\text{color}}\mathcal{L}_{\text{color}} + \lambda_{\text{spill}}\mathcal{L}_{\text{spill}} + \lambda_{\text{LPIPS}}\mathcal{L}_{\text{LPIPS}}. \tag{27}$$

The terms measure foreground color ℓ_1 error, opacity outside the silhouette, and perceptual error. They are normalized per image and use the view weights in Table 5. LPIPS uses 512×512 images with gray outside the target mask.

Three regularizers constrain Gaussian shape and displacement. The shape loss is forcing the gaussians to take a spherical shape and calculated as the two logarithmic shape ratios each splat is parameterised by, its in-plane stretch and its flatness, both zero for a perfect sphere, and average their squares over all points. The offset loss is the mean squared displacement in ray coordinates after dividing each component by its allowed maximum (15, 5, and 5 mm). The shifted-center loss is coverage-weighted Chamfer between $\boldsymbol{\mu}$ and ground-truth centers. Their weights appear in Table 4. Shape regularization updates only the appearance head; the two offset losses update only the offset layer through detached features and point positions. The full objective adds $\mathcal{L}_{\text{img}}$ and these three weighted regularizers to $\mathcal{L}_{\text{recon}}$.

Training configuration. We use NVIDIA RTX 6000 Ada GPUs and AdamW with weight decay 10^{-5}. Training uses BF16 mixed precision, with FP32 depth outputs to preserve small metric differences and FP32 query features supplied to the point decoder.

The observation heads train for 20,000 steps on 400 recorded performance sequences, with 28 frame pairs per step. A 200-step warm-up precedes cosine learning-rate decay from 5×10^{-5} to 5×10^{-6}; skeleton and camera multipliers are 10 and 0.5. Depth, camera, and field heads start from an earlier 3,000-step checkpoint; joint and scale heads start from scratch.

Reconstruction and appearance train separately, one after the other, on the Dynamic 3dgs DNA dataset wihout the 3dgs augmentations, for 35,000 steps. We sample consecutive-frame pairs from 16-frame windows; observation pretraining uses gaps of 1, 2, 4, or 8 frames. Reconstruction warms up to 10^{-4} over 1,000 steps, follows cosine decay toward 10^{-5}, and uses a fixed 5×10^{-5} from step 25,000. Appearance trains afterwards, on top of the reconstruction network frozen at its 35,000-step checkpoint: the point decoder is kept in evaluation mode and only the appearance head is optimised, so no image gradient ever reaches the geometry. It sees single frames - 395 takes, 84,975 frames, 48 frames per step over seven GPUs in two stages: 5,000 steps against 1,024-pixel renders, and 3,000 final steps in which the per-point center offsets are enabled (at most 15 mm along the viewing ray and 5 mm across it, penalized with weight 0.01). No stage uses a warm-up; each follows a cosine decay over its own length, from 10^{-4} in the first two stages and 5×10^{-5} in the last, down to 0.5 % of that peak. Table 5 records the settings along this training sequence.

Loss weights increase linearly during warm-up. Front-depth and joint-reprojection terms ramp over the first 200 observation-training steps; joint-surface coupling starts at step 500. During reconstruction training, part and flow losses ramp over steps 200-1,000, depth and voxel terms over 1,000-2,000. The appearance stages carry their own ramps: each is supervised by sixteen target views, the four cardinal ones weighted twice and twelve drawn at random; the image loss enters at one tenth of its weight and reaches full strength over 1,000 steps, the perceptual term is held back for those same 1,000 steps, and the point-set term grows from 0.05 to 0.25 over 500 steps.

Validation and additional results. The reconstruction ablations in Table 8 use eight consecutive frames from each of 16 sequences belonging to subjects excluded from reconstruction training, yielding 128 scored frames. This set informed training decisions and is therefore a validation set.

Table 6 supplements the reconstruction comparisons in Section 4.3 with separate input-, back-, and side-view appearance scores, together with results on held-out takes and in-the-wild inputs. In-the-wild inputs lack reference geometry and novel-view ground truth, so their quantitative evaluation is restricted to input-view appearance against the observed photograph.

Coarse appearance comparison. Table 7 compares coarse renderings before diffusion refinement, averaging appearance metrics over the input, opposite, and two side views. Our predictor achieves higher masked PSNR and SSIM than MonoPort and LHM++ on all three datasets. Compared with MonoPort, LPIPS is slightly lower on 4D-DRESS but higher on ActorsHQ and DNA-

Table 6: Reconstruction and novel-view quality on the three test sources, held-out takes, and in-the-wild clips. Geometry is evaluated using symmetric Chamfer distance (CD) and F1 at 1 cm / 3 cm against ground-truth surface points. Appearance is evaluated using PSNR(dB), SSIM, and LPIPS (VGG), averaged from all rendered images, all at 2.825 m with the same field of view. Values are medians across sequences of per-sequence medians. In-the-wild clips have no ground truth, the front-view reference is the input photograph.

	Geometry			Images, avg.		
Source	CD (mm)↓	F1@1↑	F1@3↑	PSNR↑	SSIM↑	LPIPS↓
4DAnyone	24.84	0.270	0.745	21.36	0.887	0.105
ActorsHQ	18.65	0.365	0.813	23.35	0.921	0.095
DNA-Rendering	19.77	0.338	0.808	22.64	0.905	0.105
Held-out takes	17.88	0.319	0.852	22.56	0.893	0.099

Table 7: Coarse appearance against ground-truth images. PSNR (dB), SSIM, and LPIPS (VGG) are averaged over front, back, and two side views at a distance of 2.825 m with the same field of view as the input. SSIM and LPIPS use ground-truth silhouette crops. Values are medians across sequences.

	4DAnyone			ActorsHQ			DNA-Rendering		
Method	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓	PSNR↑	SSIM↑	LPIPS↓
MonoPort	20.58	0.881	0.106	21.32	**0.913**	0.079	20.59	0.892	0.104
LHM++	20.82	0.883	**0.102**	22.21	0.918	0.074	21.89	0.898	0.093
Human3Diffusion	21.40	**0.889**	0.098	23.2	0.922	0.069	21.83	0.899	**0.091**
IDOL	20.58	0.876	0.111	22.15	0.914	0.083	21.68	0.894	0.101
PERSONA	20.07	0.878	0.152	21.29	0.910	0.110	20.93	0.896	0.133
Disco4D	19.91	0.858	0.131	20.01	0.887	0.111	20.02	0.868	0.113
Ours	**21.36**	0.887	0.105	**23.35**	0.921	**0.095**	**22.64**	**0.905**	0.105

Rendering. LHM++ achieves lower LPIPS on all three datasets despite its larger geometric errors. These results distinguish coarse geometric accuracy from rendered perceptual quality.

Ablation details. Table 8 presents a cumulative ablation of the reconstruction architecture and losses. Each configuration is trained from scratch for 5,000 steps with a single seed using 16-frame windows, with loss ramps rescaled to this budget. Consecutive rows add one component to the preceding configuration.

We evaluate on 128 frames from 16 validation subjects excluded from reconstruction training, as described in Appendix A.3. This validation set informed training decisions. Ground-truth front surfaces isolate reconstruction from upstream depth errors. Alongside geometric accuracy, we report temporal churn, defined in Eq. equation 28.

Adding skeleton conditioning reduces Chamfer distance from 18.30 to 17.36 mm. Temporal supervision reduces churn from 25.57 to 13.84 mm, approximately 46%, while Chamfer distance changes from 16.09 to 16.18 mm. Hierarchical decoding subsequently reduces Chamfer distance to 15.14 mm and increases F1@1 cm from 0.332 to 0.380. These comparisons characterize the incremental effects of the components under accurate partial observations.

We quantify agreement with tracked surface motion using temporal churn:

$$\mathrm{churn}_t^{\mathrm{gt}} = \frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \left\| \left(p_i^t - p_i^{t-1}\right) - \left(g_{j(i)}^t - g_{j(i)}^{t-1}\right) \right\|_2, \qquad j(i) = \arg\min_j \left\| p_i^{t-1} - g_j^{t-1} \right\|_2, \quad (28)$$

here, p_i^t is the position predicted by slot i at time t, and g_j^t is a temporally tracked ground-truth point. The set $\mathcal{M}$ contains slots whose nearest ground-truth match at time $t-1$ lies within 5 cm. Churn measures the residual between predicted slot displacement and matched ground-truth displacement.

Table 8: Cumulative reconstruction ablation with ground-truth front surfaces. All configurations use a 5,000-step training budget and a single seed. CD and churn are reported in mm; F1 thresholds are 1 and 3 cm.

	Geometry			Temporal
Component	CD↓	F1@1↑	F1@3↑	Churn↓
Transformer	33.38	0.168	0.621	14.36
+ Condition tokens	19.62	0.287	0.838	16.25
+ Image adaLN	18.30	0.301	0.852	26.46
+ Skeleton adaLn	17.36	0.309	0.875	25.30
+ Huber Chamfer	17.32	0.318	0.876	24.49
+ Adaptive balancing	17.13	0.307	0.878	23.06
+ Region Chamfer	16.09	0.341	0.900	25.57
+ Temporal loss	16.18	0.332	0.904	13.84
+ Hierarchical decoder (*full*)	**15.14**	**0.380**	**0.920**	**13.67**

Lower values indicate closer agreement with the tracked motion among valid matches; unmatched slots do not contribute.

A.4 Diffusion Re-renderer Module

Backbone and LoRA configuration. We use Wan2.2-I2V-A14B, comprising two 14B-parameter experts. Routing uses the diffusion timestep t: the high-noise expert is active for $t \geq 900$, and the low-noise expert otherwise. Rank-64 LoRA updates are applied to the attention query, key, value, and output projections, the gate-compression projections, and both feed-forward linear layers in both experts.

Input construction. The native patch embedding receives 36 channels: 16 noisy target-latent channels, four temporally packed mask channels, and 16 identity-conditioning latent channels. We construct the identity-conditioning video from the initial reference image followed by zero-valued frames in normalized RGB space, then encode it with the VAE. The corresponding mask marks the initial frame as observed and subsequent frames as unobserved.

Conditioning adapter architecture. The projection W_c is implemented as a 3D convolution whose kernel and stride match those of the pretrained patch embedding.

We initialize W_c to zero, so the added conditioning pathways initially contribute zero residuals to the pretrained input embedding.

Training configuration. The initial video-to-video adaptation uses complete 149-frame clips, trained on 16 NVIDIA GH200 GPUs with an effective batch size of 16. We use the same fixed text prompt for all clips.

A.5 Autoregressive Adaptation

Temporal layout and frame sampling. The 149-frame RGB training clips are encoded into $L = 38$ latent frames. The VAE has a temporal compression factor of four, with special handling of the initial frame. We index the identity anchor by 0 and sample the supervised target uniformly:

$$k \sim \mathrm{Uniform}\left(\{1, \ldots, L-1\}\right). \tag{29}$$

Only frame k contributes to the denoising loss. Preceding ground-truth latents, including the identity anchor, are supplied at diffusion timestep $t = 0$.

Noise levels and expert routing. The training target noise level is sampled uniformly from the eight levels of the inference schedule, $\mathcal{S}_\sigma = \{\sigma_e\}_{e=0}^{7}$. Three levels route to the high-noise expert

and five to the low-noise expert, giving sampling proportions of $3/8$ and $5/8$, respectively. At inference we use four steps. Routing follows the timestep boundary specified in Appendix A.4.

Identity anchor and cache updates. The identity anchor attends only to itself and is excluded from the denoising loss. When the cache reaches capacity, we evict the oldest non-anchor frame.

During denoising, the current frame's cache entries are refreshed at each sampling step. After generation, its clean latent is processed through a separate forward pass at $t = 0$ to construct the history entry used by subsequent frames.

Rotary-position implementation. Keys are cached before applying the temporal rotary transformation. At each attention call, we apply this transformation using the frames' current window-relative positions.

Let n denote the number of frames participating in the call, including the identity anchor, retained history, and current frame. Their temporal positions are $0, \ldots, n-1$: the anchor occupies position 0, and the current frame's queries and keys both use position $n-1$. Spatial rotary positions remain unchanged.

A.6 DSR Implementation Details

Algorithms 1 and 2 describe the rollout and replay procedures.

Rollout configuration. The student is initialized from the teacher-forced causal model. We use a rolling cache of capacity $C = 16$ committed history frames, including one persistent identity sink. Rollout lengths are sampled as

$$R \sim \mathrm{Uniform}\left(\{8, \ldots, 24\}\right). \tag{30}$$

For each frame k, the exit index is sampled uniformly from the eight-step denoising schedule:

$$e_k \sim \mathrm{Uniform}\left(\{0, \ldots, 7\}\right), \qquad \sigma_k = \sigma_{e_k}. \tag{31}$$

This sampling exposes training to different denoising stages and allows predictions of varying quality to enter subsequent histories. The rollout-length range also includes sequences long enough to exercise cache eviction. History entries are committed using the zero-timestep cache update described in Appendix A.5.

Replay records and frame selection. For each generated frame, we store

$$\mathcal{B}_k = \left(\hat{x}_{0,k}, u_k, \sigma_k\right), \tag{32}$$

where $\hat{x}_{0,k}$ is the rollout's clean estimate. We select up to $K = 12$ frame indices uniformly without replacement from $\{1, \ldots, R\}$, giving

$$K_{\mathrm{eff}} = \min(K, R), \qquad |\mathcal{S}| = K_{\mathrm{eff}}. \tag{33}$$

The rollout length, exit-step assignments, and replay indices are synchronized across ranks that must execute the same expert. Synchronizing exit steps ensures consistent timestep-dependent expert routing.

History reconstruction and perturbation. For each selected frame k, we reconstruct $\mathcal{H}_k$ from the identity sink and the $\min(k-1, C-1)$ most recent recorded clean estimates preceding k. We perturb this history with additive Gaussian noise:

$$\begin{aligned} \tilde{\mathcal{H}}_k &= \mathrm{sg}(\mathcal{H}_k) + \sigma_k^{\mathrm{ctx}} \eta_k, \\ \eta_k &\sim \mathcal{N}(0, I), \qquad \sigma_k^{\mathrm{ctx}} \sim \mathrm{Uniform}[0, 0.1]. \end{aligned} \tag{34}$$

Here, sg denotes stop-gradient. The perturbed history is supplied at diffusion timestep $t = 0$, while the current frame retains its recorded input u_k and noise level σ_k. The ground-truth target remains unperturbed.

Replay execution and optimization. Each selected frame is processed through one dense block-causal forward pass containing its history window and saved current-frame input. Under the cache convention above, this requires up to C history frames plus the current frame.

We backpropagate each selected frame's loss separately, scaled by $1/K_{\text{eff}}$, and update the model after accumulating all selected-frame gradients. This avoids retaining activation graphs for multiple replay windows simultaneously.

Both experts' LoRA and conditioning-adapter parameters are trained in FP32 with a learning rate of 10^{-5}.

A.7 Surface-Aligned Reference Implementation

Inputs and coordinate convention. Training examples contain target frames x_t at 512×512, coarse renderings c_t, reference images r_t, and temporally corresponding Gaussian positions $\{\boldsymbol{\mu}_t^i\}$ (Kerbl et al., 2023). During training, reference images are rendered from the clean Gaussians using a fixed frontal camera on the trajectory orbit. At deployment, the reference is the live input-camera stream. For the equations below, $\mathbf{E}_t$ and $\mathbf{E}_r$ denote world-to-camera transformations for the target and reference cameras, respectively; $\mathbf{K}_t$ and $\mathbf{K}_r$ are their intrinsics. All geometry and observations used to construct frame t are available at or before t.

Temporal correspondence. For the training data, we fit Dynamic 3D Gaussians to synchronized multiview captures (Luiten et al., 2024). Gaussian indices persist across frames, while centers and orientations are optimized using multiview photometric supervision and local-rigidity, rotation-similarity, and long-term local-isometry regularization. These fitted associations are obtained without flow supervision. At deployment, we disable the memory module. Temporal flow supervision encourages the indices to follow tracked surface motion during training, but independent per-frame prediction does not enforce persistent material correspondence. These learned associations provide approximate temporal transport for Eq. equation **??**. Tracking or deformation errors can produce incorrect texture transfers that still pass the visibility tests, particularly under garment sliding, hair motion, or self-contact. The trust mask records observation coverage and source type, but does not explicitly model correspondence uncertainty.

Depth and coverage proxy. We rasterize the coarse Gaussians to obtain coverage and depth in the target and reference views. Let $w_{t,i}(p)$ denote the compositing weight of Gaussian i at target pixel p. Coverage and normalized rasterization of per-Gaussian attributes ϕ^i are defined as

$$\begin{aligned} \alpha_t(p) &= \sum_i w_{t,i}(p), \\ \mathcal{R}_t[\phi](p) &= \frac{\sum_i w_{t,i}(p)\phi^i}{\alpha_t(p)}. \end{aligned} \tag{35}$$

These expressions are evaluated on covered pixels with $\alpha_t(p) > 0$. Using each Gaussian's camera-axis depth $z_{t,i}$ as the attribute gives $d_t(p) = \mathcal{R}_t[z](p)$. The same construction in the reference view yields $\alpha_{r,t}$ and $d_{r,t}$, which are used in the visibility tests below.

Direct reference warping. Following geometry-based view warping (Riegler & Koltun, 2020; 2021), we map a target pixel p into the reference camera:

$$\begin{aligned} \mathbf{X}_t(p) &= \mathbf{E}_t^{-1}\pi_{\mathbf{K}_t}^{-1}(p, d_t(p))\,, \\ \mathbf{X}_t^r(p) &= \mathbf{E}_r\mathbf{X}_t(p), \\ q_t(p) &= \pi_{\mathbf{K}_r}(\mathbf{X}_t^r(p))\,, \\ \zeta_t(p) &= [\mathbf{X}_t^r(p)]_z\,. \end{aligned} \tag{36}$$

Algorithm 1 Rollout without gradients: generating training history

Require: Model parameters θ, cache capacity $C = 16$, identity-anchor latent z_{sink}, conditioning inputs $\{\text{cond}_k\}$, noise schedule $\{\sigma_e\}_{e=0}^{7}$
Ensure: Replay records $\mathcal{B}$

Disable gradient recording
Initialize an empty rolling cache $\mathcal{C}$ and replay records $\mathcal{B}$
Write z_{sink} into $\mathcal{C}$ using a $t = 0$ forward pass
if designated rank **then**
    Sample $R \sim \text{Uniform}(\{8, \dots, 24\})$
    Sample $e_k \sim \text{Uniform}(\{0, \dots, 7\})$ for $k = 1, \dots, R$
end if
Broadcast R and $\{e_k\}_{k=1}^{R}$ to ranks requiring synchronized expert routing
for $k = 1, \dots, R$ **do**
    Sample $\epsilon_k \sim \mathcal{N}(0, I)$
    $u_k \leftarrow \epsilon_k$
    if $e_k > 0$ **then**
        Advance u_k through denoising steps $0, \dots, e_k - 1$, using $\mathcal{C}$ and cond_k
    end if
    $\sigma_k \leftarrow \sigma_{e_k}$ ▷ u_k is the input to the exit step
    $\hat{x}_{0,k} \leftarrow u_k - \sigma_k \, v_\theta(u_k, \sigma_k; \mathcal{C}, \text{cond}_k)$
    Commit $\hat{x}_{0,k}$ to $\mathcal{C}$ using a $t = 0$ forward pass, evicting the oldest non-sink entry if necessary
    Append $(\hat{x}_{0,k}, u_k, \sigma_k)$ to $\mathcal{B}$
end for
return $\mathcal{B}$

Algorithm 2 Replay with gradients: ground-truth supervision

Require: Model parameters θ, replay records $\mathcal{B}$ from Algorithm 1, identity-anchor latent z_{sink}, conditioning inputs $\{\text{cond}_k\}$, ground-truth latents $\{z_k\}$
Require: History capacity $C = 16$, maximum supervised frames $K = 12$, context-noise limit $\sigma_{\max} = 0.1$
Ensure: Updated parameters θ

Enable gradient recording
Clear accumulated parameter gradients
$R \leftarrow |\mathcal{B}|$; $K_{\text{eff}} \leftarrow \min(K, R)$
if designated rank **then**
    Sample K_{eff} indices $\mathcal{S}$ uniformly without replacement from $\{1, \dots, R\}$
end if
Broadcast $\mathcal{S}$ to ranks requiring synchronized expert routing
for each $k \in \mathcal{S}$ **do**
    Retrieve $(\hat{x}_{0,k}, u_k, \sigma_k) \leftarrow \mathcal{B}_k$
    $u_k \leftarrow \text{sg}(u_k)$
    $\mathcal{H}_k \leftarrow z_{\text{sink}}$ and the $\min(k-1, C-1)$ most recent recorded predictions preceding k
    Sample $\sigma_k^{\text{ctx}} \sim \text{Uniform}[0, \sigma_{\max}]$ and $\eta_k \sim \mathcal{N}(0, I)$
    $\tilde{\mathcal{H}}_k \leftarrow \text{sg}(\mathcal{H}_k) + \sigma_k^{\text{ctx}} \eta_k$
    Supply $\tilde{\mathcal{H}}_k$ at $t = 0$ and the current input u_k at noise level σ_k
    Compute with one dense block-causal forward:
    $\tilde{x}_{0,k} \leftarrow u_k - \sigma_k \, v_\theta(u_k, \sigma_k; \tilde{\mathcal{H}}_k, \text{cond}_k)$
    Backpropagate $\frac{1}{K_{\text{eff}}} \|\tilde{x}_{0,k} - z_k\|_2^2$ and accumulate parameter gradients ▷ One replay window at a time
end for
Update θ using the accumulated gradients

Here, $\pi_{\mathbf{K}}$ denotes image projection, $\pi_{\mathbf{K}}^{-1}$ denotes back-projection, and $\zeta_t(p)$ is the projected point's camera-axis depth. A correspondence is accepted when the projected point lies in front of the reference camera and inside its image, both views have sufficient coverage, and their depths agree:

$$\begin{aligned} \alpha_t(p) &> \alpha_{\min}, \\ \alpha_{r,t}(q_t(p)) &> \alpha_{\min}, \\ |\zeta_t(p) - d_{r,t}(q_t(p))| &< \tau. \end{aligned} \tag{37}$$

We use $\alpha_{\min} = 0.3$ and $\tau = 0.06\,\mathrm{m}$. The accepted set is eroded by one pixel to reduce silhouette contamination. Valid pixels sample r_t bilinearly at $q_t(p)$. The depth proxy is derived from corrupted geometry, so accepted correspondences can still contain alignment errors, particularly at oblique views. The visibility test rejects depth-inconsistent matches but does not guarantee exact texture alignment.

Surface-memory storage and retrieval. We store every eighth reference frame in a bank of at most 64 keyframes. Each entry contains the reference image, its reference-view depth and coverage, and the Gaussian positions at that time. For keyframe $s < t$, equation **??** transports the current target surface using rasterized Gaussian displacements. We then project the transported points into the reference camera and apply equation 37 using the stored depth and coverage at time s. Fusion first accepts valid texture from the current reference. Remaining pixels are filled from valid keyframes in reverse chronological order. Retrieval stops when all covered target pixels are filled or the available keyframes are exhausted.

Pixel-level trust and compositing. We assign a source label to each target pixel:

$$\ell_t(p) = \begin{cases} \mathrm{direct}, & \text{current reference accepted}, \\ \mathrm{memory}, & \text{past reference accepted}, \\ \mathrm{absent}, & \text{no reference accepted}. \end{cases} \tag{38}$$

In the rendered training data, we additionally reject retrieved pixels with luminance at most $8/255$. This suppresses black-background samples introduced by the enlarged splats near silhouettes and between limbs. It is a background-rejection heuristic and can also reject genuinely dark foreground texture. Let $a_t(p)$ denote the accepted reference color. The composite supplied to the VAE is

$$\tilde{r}_t(p) = \begin{cases} a_t(p), & \ell_t(p) \neq \mathrm{absent}, \\ c_t(p), & \text{otherwise}. \end{cases} \tag{39}$$

We encode the composite using the VAE posterior mean to obtain $z^w \in \mathbb{R}^{16\times L\times 64\times 64}$.

Pooling reference coverage. For pooling, RGB frames are indexed from zero. The first latent frame corresponds to $G_0 = \{0\}$, and subsequent temporal groups are $G_k = \{4k-3, \ldots, 4k\}$. Each spatial latent cell u corresponds to an 8×8 pixel block. We compute direct and memory coverage as fractions:

$$\begin{aligned} M_k^{\mathrm{d}}(u) &= \frac{\sum_{t\in G_k} |\{p \in u : \ell_t(p) = \mathrm{direct}\}|}{64|G_k|}, \\ M_k^{\mathrm{m}}(u) &= \frac{\sum_{t\in G_k} |\{p \in u : \ell_t(p) = \mathrm{memory}\}|}{64|G_k|}. \end{aligned} \tag{40}$$

Thus, partial coverage remains fractional rather than being converted to a binary decision. We use $\lambda_{\mathrm{m}} = 1/2$ in Eq. equation 7.

Table 9: Complete refinement ablation across training stages, rollout settings, and reference conditioning. DSR uses at most $K = 12$ supervised replay frames unless specified otherwise. "No history noise" removes the additional history perturbation while retaining self-generated histories and diffusion noise. "Detached history K/V" blocks gradients through historical key/value features during replay. "Memory free" means that the "Full" model is inferenced with the memory disabled to show that during in-the-wild deployment the lack of memory does not significantly harm the performance. Bold indicates the best result among causal variants.

Model	PSNR↑	SSIM↑	LPIPS↓
Bidirectional (offline)	27.30	0.9522	0.0434
Causal (teacher-forced)	22.98	0.9234	0.0914
Reference warm-up	22.50	0.9207	0.0951
DMD	21.92	0.9192	0.0923
DSR ($K = 12$)	25.51	0.9390	0.0784
DSR (detached history K/V)	25.45	0.9349	0.0789
DSR ($K = 20$)	25.70	0.9385	0.0788
DSR (no history noise)	25.69	0.9381	0.0814
DSR + reference warp	25.13	0.9347	0.0881
VideoPort Full (Memory-Free)	26.87	0.9450	0.0778
VideoPort Full	**26.93**	**0.9456**	**0.0769**

Conditioning-channel construction. The native input comprises 16 noisy-latent channels, four mask channels, and 16 image-conditioning latent channels. For $k \geq 1$, we construct y_k using Eq. equation 8 and repeat the pooled trust mask across the four mask channels:

$$\mathbf{m}_k = \mathrm{Repeat}_4(\bar{m}_k). \tag{41}$$

The initial slot retains the original identity conditioning with $\mathbf{m}_0 = 1$. Unsupported cells use the encoded black-video latent $z_k^{\varnothing}$, rather than the composite latent. This avoids supplying coarse render content as reference texture in regions without an accepted observation. The fractional masks and reference content in later frames extend the original image-conditioning usage and are learned during adaptation. The coarse-render remains as defined in Eq. equation 4; reference texture does not pass through an additional adapter branch.

Warm-up and DSR training. We first warm up the extended conditioning interface using teacher forcing on the full training data, starting from initial causal forced model. We then apply the DSR objective of Section 3.5, including the reference latents and masks in both the no-gradient rollout and differentiable replay. With probability 0.1 per training sample, we remove reference texture from all non-initial slots by setting $\bar{m}_k = 0$ and $y_k = z_k^{\varnothing}$ for $k \geq 1$. The identity conditioning and geometric adapters remain active. This trains the model to operate when reference retrieval provides no usable appearance evidence. The trainable parameters remain the LoRA updates on both experts and the existing conditioning adapters; reference warping and memory introduce no additional trainable parameters.

A.8 Additional Refinement Ablations

Table 9 provides the complete refinement comparison, including the training stages and sensitivity variants omitted from the compact main-paper table. The coarse predictor is fixed throughout these comparisons.

Training stages. The bidirectional model provides an offline reference. Teacher-forced adaptation initializes the causal model. Reference warm-up further adapts the causal model to the extended reference-conditioning interface before rollout training. The DMD-DSR objective comparison uses matching initialization, conditioning, rollout sampling, history perturbation, gradient boundaries, and student training updates.

Gradients through history encoding. We ablate the history-encoding gradient by detaching historical K/V features during replay. Recorded history latents remain detached in both configurations; the ablation additionally blocks the current prediction loss from updating their K/V encoding. With $K = 12$, this variant achieves 25.45 dB PSNR, 0.9349 SSIM, and 0.0789 LPIPS, compared with 25.51 dB, 0.9390, and 0.0784 for standard DSR. Differentiable history encoding therefore provides a modest improvement across all three metrics. The detached variant still substantially outperforms DMD, showing that the reconstruction advantage of paired rollout supervision is largely retained without this gradient path.

Replay supervision density. Increasing the maximum number of supervised replay frames from $K = 12$ to $K = 20$ raises PSNR from 25.51 to 25.70 dB, while SSIM decreases from 0.9390 to 0.9385 and LPIPS increases from 0.0784 to 0.0788. These results indicate a small trade-off across the reported metrics. As described in Appendix A.6, K is a cap: the effective number of supervised frames is $\min(K, R)$ for rollout length R.

History perturbation. Removing the additional Gaussian history perturbation raises PSNR from 25.51 to 25.69 dB, while SSIM decreases from 0.9390 to 0.9381 and LPIPS increases from 0.0784 to 0.0814. Thus, the perturbation improves SSIM and LPIPS under this evaluation, with a small reduction in PSNR. This variant retains self-generated histories and diffusion noise; only the additional history perturbation is removed. We retain history perturbation in the final model because it improves SSIM and LPIPS in this evaluation, despite a small reduction in PSNR. The final two rows add direct reference warping and then surface memory to DSR, evaluating appearance retrieved from current and past input observations, respectively.

A.9 Additional Qualitative Results and Visual Comparisons

In this section, we provide extended visual comparisons and temporal evaluations across the 4DAnyone, ActorsHQ, and DNA-Rendering benchmarks. As in the main paper, all zoomed insets represent **exact, identical pixel-space crops** without manual re-centering or realignment. This directly exposes camera trajectory stability, non-rigid deformation modeling (e.g., loose clothing and dynamic hair), and subject identity preservation across long streaming rollouts.



Figure 5: **Extended qualitative novel-view synthesis comparisons on challenging motion sequences.** We compare VideoPort against state-of-the-art baselines under diverse actions, severe self-occlusions, and loose apparel. Zoomed insets display **exact, unadjusted image-coordinate crops** at identical camera viewpoints: baseline methods frequently suffer from trajectory drift, over-smoothed kinematic skinning artifacts, and identity degeneration, whereas VideoPort maintains precise camera extrinsic adherence, faithful geometry, and high-frequency appearance details.



Figure 6: **Additional qualitative novel-view comparison**. Synthesized views at matched target viewpoints. Competing methods exhibit camera-pose drift and identity degradation, whereas VideoPort strictly follows camera conditioning while preserving fine dynamic garment details.



Figure 7: **Quality of novel-view synthesis.**



Figure 8: **Quality of novel-view synthesis.**

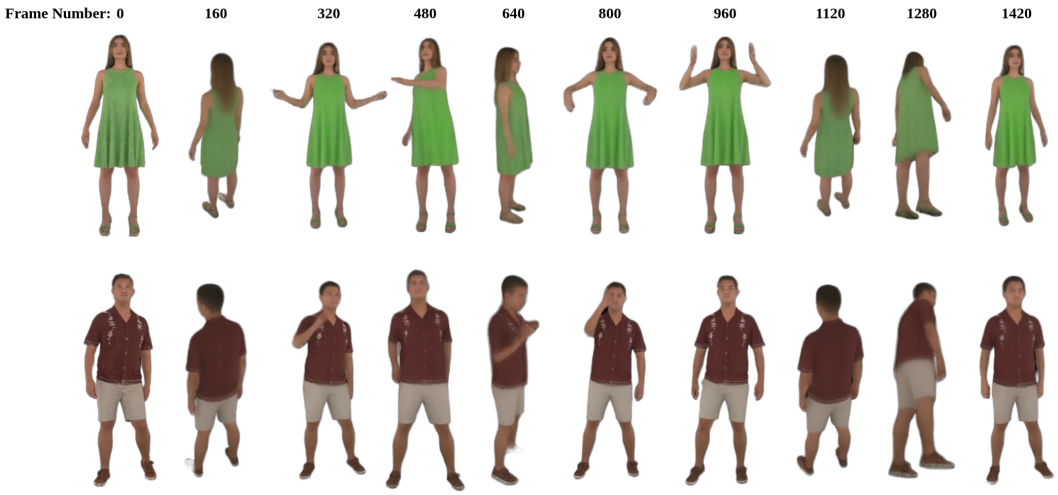


Figure 9: **Visual quality of long-horizon performance (1-3 minutes) using VideoPort Full.**



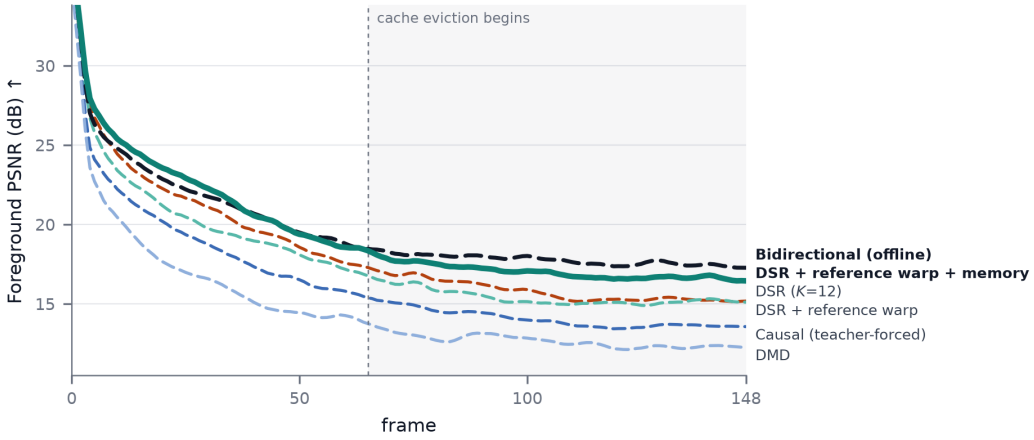
Figure 10: **In-the-wild examples.** VideoPort consistently reconstructs long horizon rollouts, preserving identity and temporal consistency even during complex motions.

Long-horizon behavior. Figure 11 reports quality throughout the 149-frame rollout. All models lose approximately 15 dB of foreground PSNR over the first 40 frames as the camera moves away from the identity view. Because viewpoint and rollout position change together, this decline alone does not distinguish increasing view difficulty from accumulated generation error.

The full model maintains higher fidelity late in the rollout, including after cache eviction begins at frame 65. Over the final 20 frames, it achieves 16.6 dB PSNR, compared with 13.6 dB for the teacher-forced causal baseline, 12.3 dB for DMD, and 15.2 dB for DSR with reference warping alone. Its PSNR is within 0.8 dB of the offline bidirectional reference, which achieves 17.4 dB.

The LPIPS results show both the benefit of surface memory and the remaining perceptual gap. Over the final 20 frames, the full model achieves 0.112, compared with 0.125 for reference warping alone and 0.135 for both the causal baseline and DMD. Thus, adding memory improves both late-rollout metrics relative to direct warping alone. However, the offline reference achieves substantially lower LPIPS at 0.061, indicating that the smaller PSNR gap does not imply equivalent perceptual quality.

(a) Foreground PSNR↑.



(b) LPIPS↓.

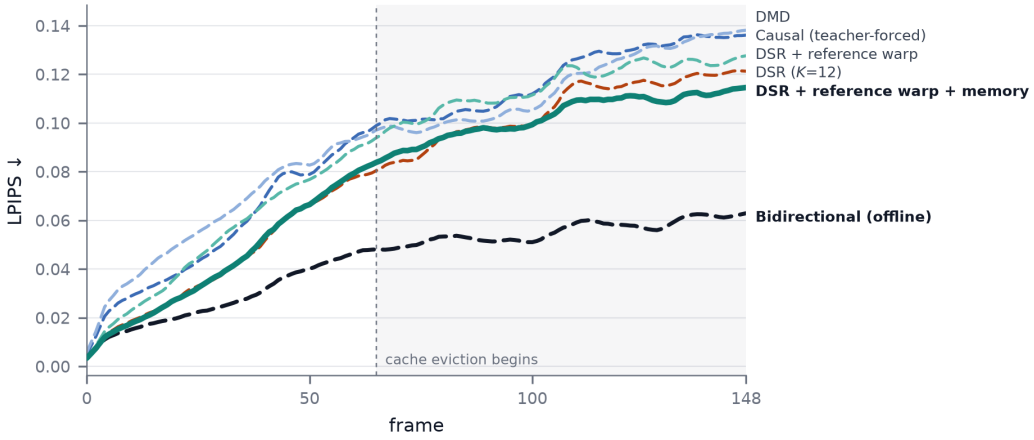


Figure 11: **Ablation Quality over 149-frame rollouts.** An ablation on per-frame metrics averaged over 16 held-out identities from the augmneted dataset with a five-frame moving average. All models, including the offline bidirectional reference, use self-generated rollouts without ground-truth target history. The **VideoPort Full** model is shown solid and other variants dashed. Shading marks frames from the onset of rolling KV-cache eviction at frame 65.

A.10 USER STUDY

Protocol. To evaluate perceptual quality, we conducted a randomized, forced-choice user study on 20 video sequences (12 from mixed evaluation datasets and 8 in-the-wild captures). 23 participants viewed synchronized, side-by-side renderings of our method and competing baselines and provided a total of 460 responses - 23 participants responded to 20 questions featuring 8 videos from the different models each. All methods were rendered along identical novel-view camera trajectories at the same resolution and frame rate. To prevent bias, method placements were completely anonymized and randomly shuffled per sequence. The order, in which sequences were shown to users, was randomized as well. Using the original monocular video as a labeled reference in the top-left corner, participants evaluated the reconstructions across seven criteria: (1) overall preference, (2) frame-to-frame temporal stability, (3) multi-view consistency during camera motion, (4) motion smoothness, (5) naturalness of dynamics, (6) identity preservation relative to the input, and (7) plausibility of clothing motion.

Results. As reported in Table 10, VideoPort outperforms all baselines across every perceptual metric. Our method is preferred overall in 60.6% of the comparisons, comfortably defeating the strongest baseline, LHM++, which secured only 21.2%. This decisive margin remains consistent across the isolated criteria, peaking at 62.1% for temporal consistency. The baselines close the gap slightly only on identity preservation (where our share is 54.2%); this aligns with our quantitative findings that while per-frame baselines can accurately reproduce the conditioning input view, they suffer severe geometric and texture degradation during novel-view spatial sweeps and temporal rollouts.

Limitations. While forced-choice perceptual studies effectively capture holistic human judgment, they remain inherently subjective. Viewers may penalize distinct generative artifacts (e.g., temporal jitter vs. texture blur) inconsistently. Furthermore, while our sequence selection includes both studio and in-the-wild captures, the perceptual evaluation is bounded by the diversity of these 20 examples, which may not fully expose failure modes associated with extreme topological changes or severe lighting variations.

Table 10: User study vote shares in %. Each column indicates a question from Sec. A.10. Higher is better.

Method	Overall	Temporal	Multi-view	Smoothness	Dynamics	Identity	Clothing
MonoPort	2.5	2.5	2.0	2.8	2.9	4.6	3.4
Human3Diffusion	2.0	2.0	2.0	1.9	2.4	2.1	3.4
DISCO-4D	1.5	1.5	1.5	1.4	1.4	1.3	1.4
IDOL	6.1	6.9	7.1	6.6	6.8	7.1	7.2
LHM++	21.2	20.2	20.9	20.4	20.8	23.1	20.7
PERSONA (ICP)	6.1	4.9	5.6	5.2	5.3	7.6	5.8
Ours	**60.6**	**62.0**	**60.9**	**61.7**	**60.4**	**54.2**	**58.1**

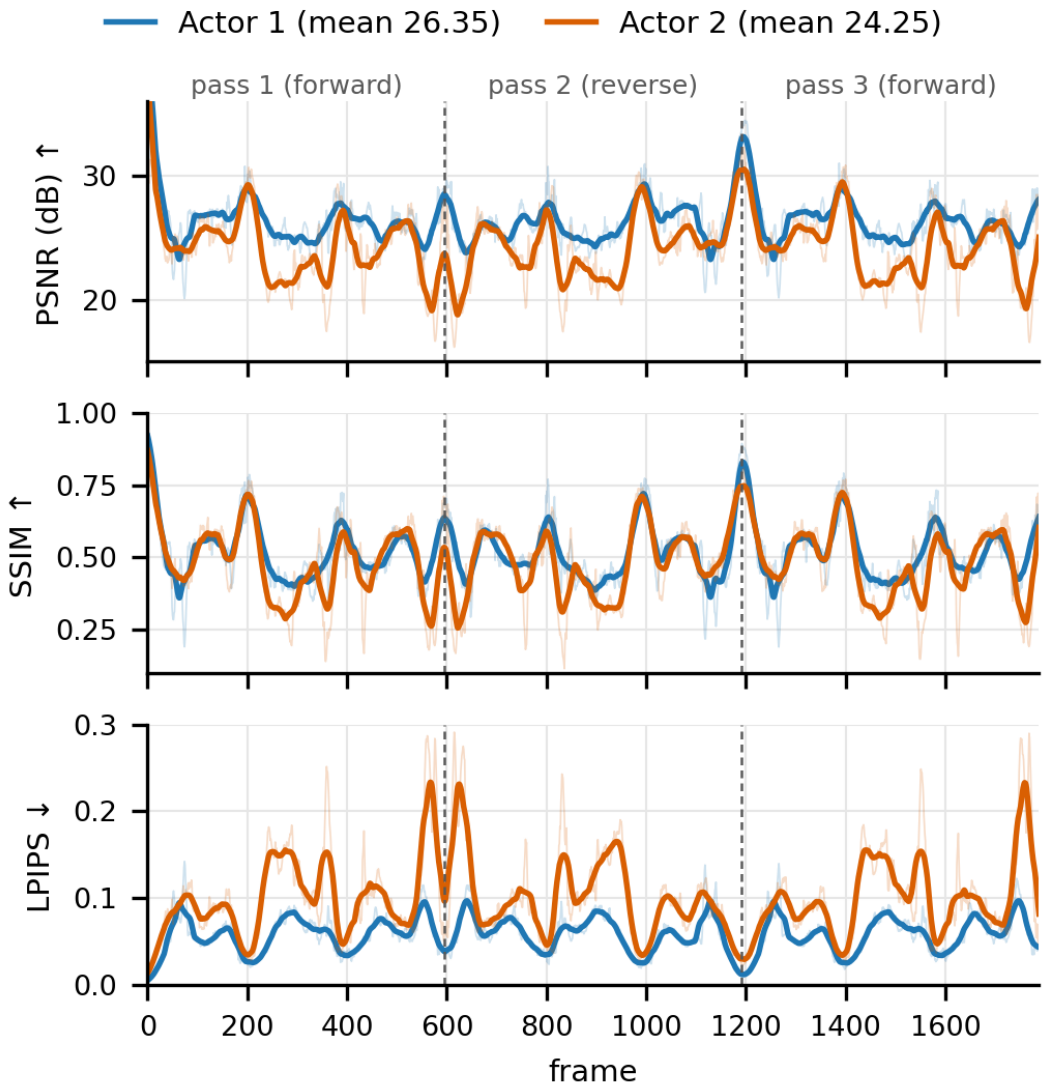


Figure 12: Two-minute novel-view synthesis using **VideoPort Full**. Per-frame PSNR, SSIM, and LPIPS for two actors across forward, reverse, and repeated forward camera passes. Mean PSNR is 26.35 and 24.25 dB, respectively.