CHRONOFY: POSE-AWARE GENERATIVE 4D RECONSTRUCTION FROM VIDEOS

Anonymous authors
Paper under double-blind review

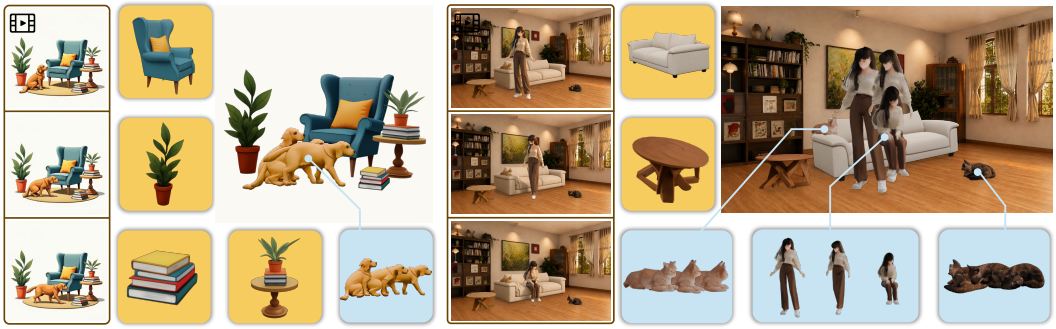


Figure 1: **Chronofy jointly generates per-object dynamic Gaussians and predicts their camera-space poses.** The predicted poses place the generated objects in a shared camera coordinate system, enabling dynamic scene reconstruction. Blue box: spatiotemporally consistent dynamic objects.

ABSTRACT

Recent generative reconstruction methods effectively reconstruct static 3D content but often exhibit pronounced inter-frame jitter when applied to dynamic content. We present **Chronofy**, a pose-aware 4D generative reconstruction framework that jointly predicts complete dynamic objects and unified camera-space poses from single-view videos. Built upon SAM3D, Chronofy transfers its powerful generative reconstruction priors to the 4D domain through three key designs: (i) a *pose-aware 4D representation* that encodes frame-dependent motion in spatiotemporal structured latents and uses a unified pose for sequence-level placement; (ii) a *generative reconstruction model* that utilizes a spatiotemporal mixture of transformers for temporal modeling and frame-wise correspondence between shape and pose; and (iii) a *dynamic Gaussian decoder* that combines temporal latent interactions with time-varying Gaussian attributes to reduce inter-frame flickering. To preserve occlusion robustness with limited 4D training data, we use temporal masking augmentation, while layout test-time optimization refines multi-object spatial alignment. Experiments on Motion-80, Kubric-4D, and our simulated real-world Chrono-64 datasets demonstrate that Chronofy substantially improves spatiotemporal consistency in both 4D object generation and 4D scene reconstruction.

1 INTRODUCTION

Reconstructing dynamic 3D (also called 4D) content from videos is a crucial yet challenging task, with broad applications in virtual reality (VR), augmented reality (AR), and game design. Recent geometry-based approaches extend foundational 3D reconstruction models (Wang et al., 2024; 2025a; Lin et al., 2025) by modeling inter-frame relationships to recover dynamic structure and geometry (Zhang et al., 2025b; Sucar et al., 2025; Zhang et al., 2026). Although these methods produce reconstructions that align well with the input video, they struggle to recover regions that are not observed. Reconstructions from single-view videos are thus often limited to visible 2.5D surfaces, making them less suitable for novel-view synthesis or reuse as complete 4D assets.

Diffusion-based 4D generation offers a complementary strength by completing unobserved regions using learned generative priors. By adapting pretrained 3D generative models (Li et al., 2025a;

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.

Zhao et al., 2025; Xiang et al., 2025) with limited 4D data, recent methods (Yenphraphai et al., 2026; Li et al., 2025b; Sabathier et al., 2026) synthesize complete 4D assets with plausible geometry and appearance beyond the observed surfaces. However, these methods primarily emphasize visual quality without explicitly constraining object poses in the camera coordinate system, leaving spatial alignment with the input video unresolved. Recovering this alignment can require additional pose estimation or rendering-based pose optimization. Such pipelines separate asset generation from pose recovery, rather than learning them jointly. The challenge is thus to generate complete 4D assets from ambiguous observations while maintaining temporal coherence and spatial alignment.

By integrating generative priors with geometric cues, generative reconstruction brings completion and spatial alignment into a unified framework. Existing approaches have explored this direction for static objects and scenes (Chang et al., 2026; Huang et al., 2026; Siddiqui et al., 2026). Among these methods, SAM3D (Chen et al., 2026b) jointly predicts object shape and pose in the camera coordinate system, allowing reconstructed objects to be placed in a shared scene. This provides a useful starting point for 4D generative reconstruction, but applying SAM3D independently to video frames produces temporally inconsistent shapes and poses. Extending this prior to 4D content therefore requires addressing two key challenges. (i) Coordinating shape and pose over time. Frame-to-frame motion can be represented through either object deformation or pose transformations. Without a consistent allocation of motion between these components across frames, their predictions can become temporally inconsistent, causing shape and pose jitter. (ii) Preserving robustness with limited 4D data. SAM3D benefits from extensive post-training on diverse real-world 3D data, whereas adaptation using only synthetic 4D data risks weakening its robustness and generalization.

To address these challenges, we propose Chronofy, a pose-aware 4D generative reconstruction framework that jointly generates complete dynamic objects and camera-space poses from single-view videos. Chronofy comprises three main components. (i) Pose-aware 4D representation. We encode both object deformation and spatial displacement in spatiotemporal structured latents within a sequence-shared normalized space. A unified pose maps the entire sequence to the camera coordinate system, removing the need to synchronize frame-dependent motion across shape and pose branches. (ii) Generative reconstruction model. We extend the spatial Mixture of Transformers (MoT) (Liang et al., 2024) into a spatiotemporal architecture that models temporal dependencies within each branch. A tailored attention mask restricts each pose token to shape tokens from the corresponding frame, preserving their correspondence during joint generation. (iii) Dynamic Gaussian decoder. We combine temporal latent interactions with time-varying Gaussian attributes (Luiten et al., 2024; Wang et al., 2025d) to decode the structured latents into temporally coherent 4D content, reducing jitter during decoding and rendering. To retain SAM3D's occlusion robustness despite limited real-world 4D training data, we introduce temporal masking augmentation (Wu et al., 2025) to simulate occlusions during fine-tuning. For multi-object scenes, the predicted unified poses place individual reconstructions in a shared camera coordinate system. Layout test-time optimization (Labbé et al., 2022) further refines these poses to improve scene alignment with the input video.

Overall, our contributions can be summarized as follows:

- We propose Chronofy, a pose-aware 4D generative reconstruction framework. To our knowledge, it is the first to jointly generate complete 4D objects and camera space poses from single-view videos, enabling both object reconstruction and multi-object scene composition.
- We develop a pose-aware 4D representation, a generative reconstruction model, and a dynamic Gaussian decoder to jointly model object motion and scene placement while maintaining temporal consistency. Temporal masking augmentation and layout test-time optimization further improve occlusion robustness and scene alignment in complex multi-object dynamic scenes.
- We introduce Chrono-64, a 64-video dataset of simulated real-world scenes, and evaluate our framework on it alongside Motion-80 (Chen et al., 2026a) and Kubric-4D (Van Hoorick et al., 2024). Extensive quantitative and qualitative experiments demonstrate competitive performance in both 4D object generation and 4D scene reconstruction with accurate layouts.

2 Related Work

Multi-view 4D Reconstruction. Classical 3D reconstruction builds on Structure-from-Motion (SfM) and Multi-View Stereo (MVS) (Hartley et al., 2003; Tomasi & Kanade, 1992; Goesele et al., 2006; Schönberger et al., 2016), while early 4D systems extend per-frame reconstruction through

temporal registration or joint spatiotemporal optimization (Collet et al., 2015; Goldluecke & Magnor, 2004; Mustafa et al., 2016). Recent feed-forward models, such as DUSt3R (Wang et al., 2024) and VGGT (Wang et al., 2025a), directly infer geometry from image collections and have inspired extensions to dynamic scenes (Zhang et al., 2025b; Wang et al., 2025c; Hu et al., 2026). In particular, DPM (Sucar et al., 2025) predicts dynamic pointmaps in a time-invariant canonical space, while D4RT (Zhang et al., 2026) uses a global 4D representation to query scene geometry at arbitrary locations and times. Although these reconstruction-based methods achieve accurate video alignment, they rely on sufficient multi-view evidence and primarily recover observed scene surfaces. Consequently, they cannot reliably reconstruct complete 4D assets when large regions remain unobserved. In contrast, we target complete and video-aligned 4D reconstruction from a single-view video.

Single-view 4D Generation. Single-view 4D generation uses generative priors to recover geometry and appearance beyond the observed views. Early approaches distill image or video diffusion priors (Poole et al., 2022; Bahmani et al., 2024; Yin et al., 2023; Ren et al., 2023; Zeng et al., 2024), whereas recent methods extend 3D-native generative models (Li et al., 2025a; Zhao et al., 2025; Xiang et al., 2025) with temporal modules to produce dynamic mesh (Chen et al., 2026a; Jiang et al., 2026; Yenphraphai et al., 2026; Gökmen et al., 2026) or 3D Gaussian sequences (Ren et al., 2024; Li et al., 2025b; Zhang et al., 2025a; Kwon et al., 2026). These methods do not directly predict poses, so composition can require additional pose estimation or rendering-based pose optimization. We aim to jointly recover complete 4D objects and explicit camera-space poses from the input video.

Generative 3D Reconstruction. Generative reconstruction combines input alignment with generative completion, achieving alignment implicitly or through explicit pose estimation. Implicit approaches achieve alignment through intermediate views or geometric conditioning. Some synthesize novel views before reconstructing geometry (Voleti et al., 2024; Liu et al., 2023; Xu et al., 2024; Hong et al., 2024; Tang et al., 2024), but view inconsistencies can degrade reconstruction quality. Within this category, geometry-aware methods such as ReconViaGen (Chang et al., 2026) condition TRELLIS (Xiang et al., 2025) on VGGT geometric features (Wang et al., 2025a) to improve input alignment, without jointly generating object shape and camera-space pose. Explicit approaches instead jointly generate complete 3D objects and estimate their camera-space poses (Meng et al., 2026; Huang et al., 2026), enabling direct scene-level composition. Based on the pretrained 3D generative reconstruction prior (Chen et al., 2026b), Lift4D (Litman et al., 2026) keeps it frozen and optimizes deformable Gaussians for each video to improve temporal consistency. In contrast, Chronofy adapts this prior to the 4D domain, learning to jointly generate complete dynamic objects and unified camera-space poses, enabling multi-object dynamic scene reconstruction.

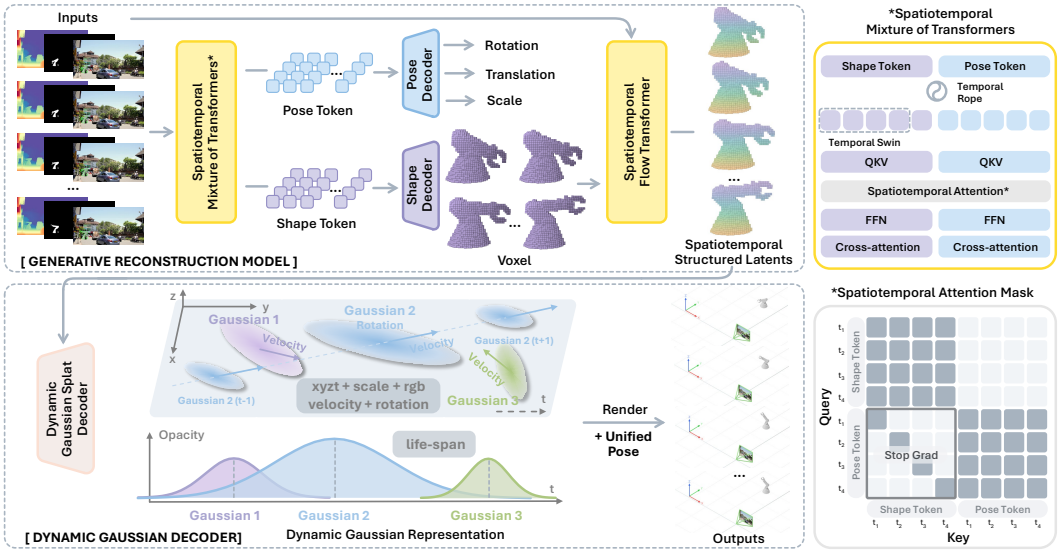


Figure 2: **Chronofy overview.** The whole framework comprises two main modules: (i) the Generative Reconstruction Model, which jointly predicts spatiotemporal structured latents and unified poses from input conditions; and (ii) the Dynamic Gaussian Decoder, which converts these latents into dynamic Gaussians that are mapped to the camera coordinate system using the unified poses.

3 METHOD

We propose **Chronofy**, a pose-aware generative 4D reconstruction framework built on SAM3D (Chen et al., 2026b). Figure 2 provides an overview of our framework. Given input conditions $\mathcal{C} = \{C_t\}_{t=1}^{T}$ with T frames, Chronofy generates dynamic objects as dynamic Gaussians and predicts unified poses that place them in the camera coordinate system. We first introduce a pose-aware 4D representation (Section 3.1) to efficiently represent the relationship between dynamic objects and corresponding poses, and then we present a generative reconstruction model (Section 3.2), based on a spatiotemporal Mixture of Transformers (MoT), that jointly generates the spatiotemporal structured latents $\mathcal{Z}$ and unified poses P. Finally, we design a dynamic Gaussian decoder (Section 3.3) that converts the latents into dynamic Gaussians with time-varying attributes.

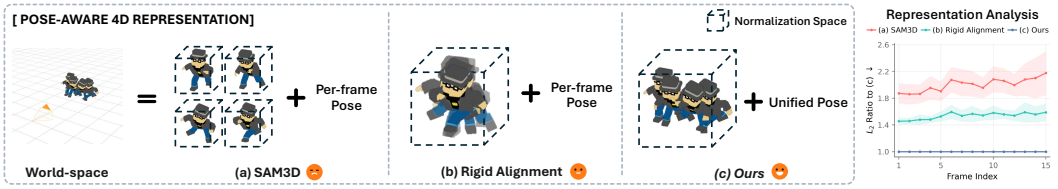


Figure 3: **Comparison of pose-aware 4D representations.** (a)-(c) illustrate three normalization and pose strategies. On the right, our representation with a unified pose shows the lowest L_2 variation between adjacent frames, indicating greater temporal consistency, i.e., less jitter.

3.1 POSE-AWARE 4D REPRESENTATION

Dynamic content often combines intrinsic deformation and extrinsic pose changes, as in walking or dancing. To capture both forms of motion coherently, a suitable 4D representation is needed. Figure 3 compares three alternatives, with our formulation in (c) exhibiting the lowest frame-to-frame variation, potentially facilitating learning of smooth temporal dynamics. Specifically, in (a), each frame is independently normalized and encoded with a per-frame pose, as in SAM3D. In (b), frames are rigidly aligned to the first frame and then jointly normalized to encode deformation, with per-frame poses recovering the removed transformations. Both require temporal synchronization between shape and pose predictions, and rigid alignment in (b) may introduce errors under nonrigid deformation. In (c), we instead propose a pose-aware 4D representation $\mathcal{R}$ comprising spatiotemporal structured latents $\mathcal{Z}$ (Xiang et al., 2025) and a unified spatial pose P. The latents encode deformation and spatial displacement in a sequence-shared normalized space; the unified pose maps the sequence to camera coordinates. Thus, frame-dependent motion resides in $\mathcal{Z}$, eliminating the need for cross-branch temporal synchronization and promoting smoother, more consistent motion.

Spatiotemporal structured latents. We define a dynamic 3D asset as $\mathcal{O} = \{\mathcal{O}_t\}_{t=1}^{T}$, where $\mathcal{O}_t = \langle \mathcal{V}_t, \mathcal{F}_t, \mathcal{T}_t \rangle$ denotes the mesh at frame t, with vertices $\mathcal{V}_t$, faces $\mathcal{F}_t$, and texture $\mathcal{T}_t$. To place all frames in a shared canonical space, we jointly normalize the sequence using a global Axis-Aligned Bounding Box (AABB) shared across the entire sequence of length T:

$$b_{\min}^{(d)} = \min_{1 \le t \le T} \left(\min_{\mathbf{x} \in \mathcal{V}^t} x^{(d)} \right), \quad b_{\max}^{(d)} = \max_{1 \le t \le T} \left(\max_{\mathbf{x} \in \mathcal{V}^t} x^{(d)} \right) \tag{1}$$

where $\mathbf{b}_{\min}, \mathbf{b}_{\max} \in \mathbb{R}^3$ are the sequence-level bounds, and $d \in \{1, 2, 3\}$ indexes the three spatial axes. We then compute the shared center $\mathbf{c} = \frac{1}{2}(\mathbf{b}_{\min} + \mathbf{b}_{\max})$ and isotropic normalization scale $s_{\text{norm}} = 1/\|\mathbf{b}_{\max} - \mathbf{b}_{\min}\|_{\infty}$. Subsequently, we voxelize the jointly normalized asset sequence into O and use SAM3D's shape encoder (Chen et al., 2026b) to obtain coarse shape latents $\mathcal{S} = \{s_t\}_{t=1}^{T}$. We then employ SAM3D Depth-VAE to encode the sequence into spatiotemporal structured latents $\mathcal{Z} = \{z_t\}_{t=1}^{T}$, capturing complete motion for consistent 4D representation learning.

Unified spatial pose. For the motion sequence in the jointly normalized space, we only model a unified pose P shared across frames for the normalized space. Specifically, the unified pose $P = [R, t, s] \in \mathbb{R}^{12}$ comprises the 6D continuous rotation representation $R \in \mathbb{R}^6$ (Zhou et al., 2019), translation $t \in \mathbb{R}^3$, and scale $s \in \mathbb{R}^3$ of the normalized space within the camera coordinate system. To retain compatibility with SAM3D's architecture, we keep per-frame object pose predictions during training but supervise them with the same unified pose. At inference, we aggregate the

predicted poses across frames to obtain the final unified pose P. This design avoids independent frame-wise transformations and reduces pose jitter across the reconstructed sequence.

3.2 Generative Reconstruction Model

Our generative reconstruction model jointly generates spatiotemporal structured latents $\mathcal{Z}$ and a unified pose P conditioned on the single-view video $\mathcal{V} = \{I_t\}_{t=1}^T$, object masks $\mathcal{M} = \{M_t\}_{t=1}^T$ and pointmaps $\mathcal{P} = \{\mathcal{P}_t\}_{t=1}^T$. However, single-view videos provide only partial observations of dynamic objects, leaving occluded and unseen regions underdetermined. Multiple complete 4D reconstructions can therefore be consistent with the same observations. To capture this ambiguity, we model reconstruction as a conditional distribution $p(\mathcal{Z}, P|\mathcal{C})$, where $\mathcal{C} = \{\mathcal{V}, \mathcal{M}, \mathcal{P}\}$. Our generative reconstruction model $q(\mathcal{Z}, P|\mathcal{C})$ aims to approximate p to achieve high-fidelity 4D reconstruction.

Input encoding. Following SAM3D (Chen et al., 2026b), we use DINOv2 (Oquab et al., 2023) to encode video frames $\mathcal{V}$ and masks $\mathcal{M}$, and a pointmap encoder to encode pointmaps $\mathcal{P}$, producing three conditioning token sets. Before encoding, we compute a joint bounding box over the valid foreground regions in all T frames using the masks and consistently crop each frame. This shared cropping preserves cross-frame motion cues aligned with the spatiotemporal structured latents. As the original model's pose prediction relies heavily on input pointmaps (Wang et al., 2025b), we also condition the model on $\mathcal{P}$, which is jointly normalized into the unified canonical space. Unlike SAM3D, we use only the cropped object content, while scene-level spatial relationships are recovered by denormalizing the pointmaps. This design greatly simplifies the construction of 4D training data with complex backgrounds and ground-truth poses, facilitating adaptation from 3D to 4D.

Spatiotemporal mixture of transformers. To model coherent temporal dynamics while preserving the multi-modal modeling capability of SAM3D, we temporally inflate the self-attention layers in its original mixture of transformers (Liang et al., 2024; Deng et al., 2025), as illustrated in Figure 2. Specifically, we separately inflate the self-attention of the pose tokens P and the shape tokens corresponding to coarse shape latents $\mathcal{S}$ encoded from the coarse voxel grid O:

$$\text{infattn}(\mathbf{X}) = \text{reshape}^{-1}(\text{selfattn}(\text{reshape}(\mathbf{X}))), \quad \text{for } \mathbf{X} \in \{\mathcal{S}, P\} \tag{2}$$

where $\mathbf{X} \in \mathbb{R}^{B\times T\times L\times C}$, with batchsize B, T frames, L tokens per frame, and feature dimension C. We merge T and L to enable cross-frame spatiotemporal information exchange, then restore the original tensor shape. We further apply 1D Rotary Positional Embedding (RoPE) (Heo et al., 2024) along the temporal dimension to encode relative temporal positions between tokens. Together, these operations enable each stream to jointly model spatial and temporal dependencies. Additionally, we extend the original spatial multi-modal attention mask across time. Specifically, each pose token P_t attends to pose tokens from all frames but only the corresponding coarse shape latent s_t:

$$P'_t = \text{Attn}(\mathbf{Q} = P_t,\ \mathbf{K} = [\{P_\tau\}_{\tau=1}^T, \text{sg}(s_t)],\ \mathbf{V} = [\{P_\tau\}_{\tau=1}^T, \text{sg}(s_t)]) \tag{3}$$

which enables efficient cross-modal interactions. We retain SAM3D's stop-gradient operator $\text{sg}(\cdot)$ to prevent pose gradients from affecting coarse shape learning, and use temporal sliding windows (Liu et al., 2021; Li et al., 2025b) to improve efficiency for long sequences. Conditioned on the predicted coarse shape O, video frames $\mathcal{V}$, and masks $\mathcal{M}$, the subsequent spatiotemporal flow transformer applies the same temporal self-attention inflation to generate spatiotemporal structured latents $\mathcal{Z}$.

3.3 Dynamic Gaussian Decoder

The generative reconstruction model produces spatiotemporal structured latents $\mathcal{Z}$ and a unified pose P. However, existing structured latent decoders (Xiang et al., 2025; Chen et al., 2026b) are designed to convert these latents into static 3D assets, such as meshes or 3D Gaussians. Applying them independently across frames can introduce temporal jitter and inconsistencies, even with temporally coherent latents. Inspired by 4DGT (Xu et al., 2025) and FreeTimeGS (Wang et al., 2025d), we introduce a dynamic Gaussian decoder that combines temporal feature interactions with time-varying Gaussian attributes, promoting temporal consistency during both decoding and rendering.

Dynamic Gaussian representation. We adopt a variant of 4DGS (Luiten et al., 2024; Yang et al., 2024) to represent dynamic content. Each Gaussian extends the standard 3DGS (Kerbl et al., 2023), denoted by position $\mathbf{x} \in \mathbb{R}^3$, scale $\mathbf{s} \in \mathbb{R}^3$, color $\mathbf{c} \in \mathbb{R}^3$, opacity $o \in \mathbb{R}$, and orientation quaternion $\mathbf{q} \in \mathbb{R}^4$, with lifespan $\ell \in \mathbb{R}$, velocity $\mathbf{v} \in \mathbb{R}^3$, and angular velocity $\boldsymbol{\omega} \in \mathbb{R}^3$ in axis-angle

form. For a sequence of T frames, each Gaussian has a source frame $t \in \{1, \dots, T\}$ that fixes its temporal center at $c_s = (t-1)/(T-1)$. The decoder predicts the log-lifespan parameter l, which is exponentiated and bounded to obtain a positive lifespan ℓ in frame units. At a normalized target timestamp t_s, its effective opacity is

$$\mathbf{o}_{t_s} = \mathbf{o} \cdot \exp\left(-\frac{1}{2}\frac{(t_s - c_s)^2}{\boldsymbol{\sigma}^2}\right), \quad \boldsymbol{\sigma} = \frac{\ell}{T-1}, \tag{4}$$

where σ is the temporal standard deviation in normalized time. During rendering, cross-frame Gaussian contributions are discarded when their temporal multiplier falls below $m_{\text{th}} = 0.05$. To account for motion, the position and orientation evolve according to

$$\mathbf{x}_{t_s} = \mathbf{x} + \mathbf{v} \cdot (t_s - \mathbf{c}_s), \quad \mathbf{q}_{t_s} = \mathbf{q} \otimes \phi(\boldsymbol{\omega} \cdot (t_s - \mathbf{c}_s)), \tag{5}$$

where ϕ converts an axis-angle representation to a quaternion and $\otimes$ denotes quaternion multiplication. We then render each target frame using standard 3DGS rasterization (Ye et al., 2025).

Decoder architecture. To decode this representation, we extend the SAM3D 3DGS decoder (Chen et al., 2026b) with the temporal attention inflation used in our spatiotemporal mixture of transformers, allowing structured latent features to interact across frames during decoding. We also expand the output head from 14 to 21 dimensions to jointly predict $\{\mathbf{x}, \mathbf{s}, \mathbf{c}, o, \mathbf{q}, l, \mathbf{v}, \boldsymbol{\omega}\}$. Together, these changes allow the decoder to use temporal context when predicting both Gaussian appearance and motion. Figure 2 illustrates the overall architecture. During training, we use mesh-guided interpolation (Zhang et al., 2025a) to obtain approximate ground-truth Gaussian velocities for supervision, encouraging realistic motion. Details are provided in Appendix A.3.

4 Multi-Object Dynamic Scene Reconstruction

Our framework naturally extends to compositional reconstruction of multi-object dynamic scenes. We first use SAM3 (Carion et al., 2026) to segment individual objects in the input video. For each object, Chronofy jointly generates dynamic Gaussians and a unified pose conditioned on its video frames, masks, and pointmaps. These poses place the reconstructed objects in a shared camera coordinate system, allowing them to be composed into a dynamic scene, as shown in Figure 1.

Temporal masking augmentation. Videos with multiple dynamic objects inevitably contain mutual occlusions, which are difficult to model using our synthetic data. Inspired by Amodal3R (Wu et al., 2025), we introduce a temporal masking strategy prior to conditional embedding during finetuning. We randomly overlay dynamic foreground regions from other videos onto the conditioning frames before embedding. This strategy is further adapted to our multi-condition encoding to maximally exploit SAM3D's pre-trained occlusion-robust prior. Further details are provided in Appendix A.5. Subsequently, we utilize the occlusion-robust model for scene reconstruction.

Layout test-time optimization. To form scenes, we compose independently reconstructed objects in a shared camera coordinate system. Object-centric crops simplify training data construction but limit scene-level context, potentially affecting object placement. We initialize render-and-compare optimization (Labbé et al., 2022; Wen et al., 2024) with predicted poses. Specifically, we render each object's dynamic Gaussians and refine its pose individually via backpropagation to match the input images and masks. Accurate per-object placement thus supports scene-level alignment, details in Appendix A.6. We retain the optimized layout only if mask IoU improves over the initial prediction.

5 Experiments

5.1 Experimental Setup

Datasets. We evaluate 4D object generation on 64 short sequences from Motion-80 (Chen et al., 2026a) and 4D scene reconstruction on the static-camera test set of Kubric-4D (Van Hoorick et al., 2024). To further assess both generation quality and layout accuracy in simulated real-world scenes, we construct Chrono-64, a 64-video test dataset. We create these scenes by plausibly placing dynamic meshes from Objaverse (Deitke et al., 2023b;a) into backgrounds provided by SA-3DAO (Chen et al., 2026b). For each scene, we render a 16-frame video from a fixed viewpoint using the original scene camera and provide ground-truth object geometry and object poses.

Evaluation Metrics. For 4D object generation, since we represent 4D objects using dynamic Gaussians, we quantitatively evaluate their novel-view renderings against the corresponding ground truth using PSNR, SSIM (Wang et al., 2004), LPIPS (Zhang et al., 2018), and CLIP similarity (CLIP-S) (Radford et al., 2021). Additionally, we employ Fréchet Video Distance (FVD) (Unterthiner et al., 2018) to evaluate the spatiotemporal quality of videos rendered from the generated 4D content. For 4D scene reconstruction, following SAM3D (Chen et al., 2026b), we evaluate reconstruction quality by adopting the following 6D pose estimation metrics: 3D IoU, ICP-Rot, ADD-S, and ADD-S@0.1. For fair comparison, these layout metrics are computed on meshes decoded from Chronofy's predicted structured latents using the SAM3D mesh decoder, with details in Appendix A.4.

Baselines. For 4D object generation, we compare against frame-wise SAM3D (Chen et al., 2026b), Gaussian-based methods L4GM (Ren et al., 2024), GVFDiffusion (Zhang et al., 2025a), SS4D (Li et al., 2025b), and MORPHOS (Kwon et al., 2026), and textured mesh sequence methods Mesh4D (Jiang et al., 2026) and Motion324 (Chen et al., 2026a). These methods are evaluated quantitatively using novel-view renderings. Since ActionMesh (Sabathier et al., 2026) generates untextured mesh sequences, we only compare it qualitatively. For 4D scene reconstruction, we combine ActionMesh with MegaPose (Labbé et al., 2022) or FoundationPose (Wen et al., 2024), and pair Chronofy with FoundationPose to compare joint pose prediction with a separate pose estimation pipeline. We also compare against SAM3D and COM4D (Gökmen et al., 2026), which generates multi-object dynamic scenes from videos. Our comparisons focus on methods that recover complete objects rather than only 2.5D visible surfaces. All baselines use their official implementations.



Figure 4: **Qualitative comparison of 4D object generation on the Chrono-64 dataset.** Red boxes highlight SAM3D's structural jitter; green boxes show Chronofy's temporal consistency.

5.2 Experimental Results

4D Object Generation. The quantitative results in Table 1 show that Chronofy achieves the best results on most metrics, including the lowest FVD on both datasets. Although its SSIM and LPIPS scores are not always the best, its performance on both metrics remains competitive. Its strong PSNR and CLIP-S results indicate substantially improved visual fidelity and semantic alignment, while the

FVD results demonstrate better spatiotemporal consistency, particularly on real-world-style inputs (Chrono-64). Figures 4 and 13 provide qualitative comparisons on Chrono-64 across diverse object motions and challenging occlusion patterns. Although SAM3D (Chen et al., 2026b) generates high-quality objects in individual frames, its frame-wise reconstructions exhibit abrupt structural variations over time, as highlighted in the red boxes. In the corresponding regions, Chronofy maintains more temporally stable object structures and textures across frames while retaining much of SAM3D's occlusion robustness, demonstrating improved temporal consistency. The mesh-based baselines exhibit failures in challenging cases, possibly due to topology constraints.

Table 1: **Quantitative comparison of 4D object generation.** Evaluation metrics are computed by comparing rendered and ground-truth novel-view videos. Best results are in **bold**.

	Chrono-64					Motion-80 Short Sequences				
Model	LPIPS (↓)	CLIP-S (↑)	PSNR (↑)	SSIM (↑)	FVD (↓)	LPIPS (↓)	CLIP-S (↑)	PSNR (↑)	SSIM (↑)	FVD (↓)
SAM3D	0.2075	0.8392	13.1902	**0.7725**	724.04	0.1937	0.8435	14.7362	0.7543	784.31
L4GM	0.1885	0.7158	11.1340	0.6555	345.53	0.1602	0.8370	12.5225	0.7276	275.19
GVFDiffusion	0.2530	0.5928	9.1948	0.6140	858.50	0.2426	0.7263	12.4425	0.6962	436.67
SS4D	0.1932	0.7365	10.0302	0.6688	342.23	**0.1458**	0.8641	14.9383	0.7125	267.74
MORPHOS	0.1758	0.7902	10.6326	0.6592	307.93	0.1761	0.8316	13.6591	0.7051	285.31
Mesh4D	0.3201	0.6155	9.3020	0.6125	584.96	0.2153	0.7819	12.5281	0.6877	563.01
Motion324	0.3184	0.6547	9.1981	0.5986	388.23	0.2231	0.8031	10.2187	0.6894	308.07
Chronofy	**0.1635**	**0.8512**	**13.9488**	0.7647	**263.19**	0.1513	**0.8776**	**15.6836**	**0.7726**	**243.46**

4D Scene Reconstruction. Quantitative results for 4D scene reconstruction are presented in Table 2. On Chrono-64, Chronofy retains SAM3D's spatial alignment capabilities and maintains competitive performance across layout metrics. On Kubric-4D, Chronofy improves ICP-Rot, supporting the effectiveness of our unified pose formulation, but its ADD-S performance is somewhat worse than that of SAM3D. This gap may reflect the motion characteristics of the Kubric-4D dataset, where objects undergo rapid rigid translations and rotations without nonrigid deformation. Since our 4D representation models these transformations as deformations, rapid rigid motion can introduce shape errors that adversely affect ADD-S. Figures 5 and 14 further compare the methods on real-world-style scenes. Frame-wise SAM3D produces independently reconstructed objects and poses, causing temporal shape and pose jitter. COM4D, which targets scene generation rather than video-aligned reconstruction, produces larger reconstruction errors and generalizes less reliably to these inputs. In comparison, Chronofy improves inter-frame consistency across the reconstructed sequence while maintaining single-frame quality comparable to SAM3D. More evaluations in Appendix A.2.

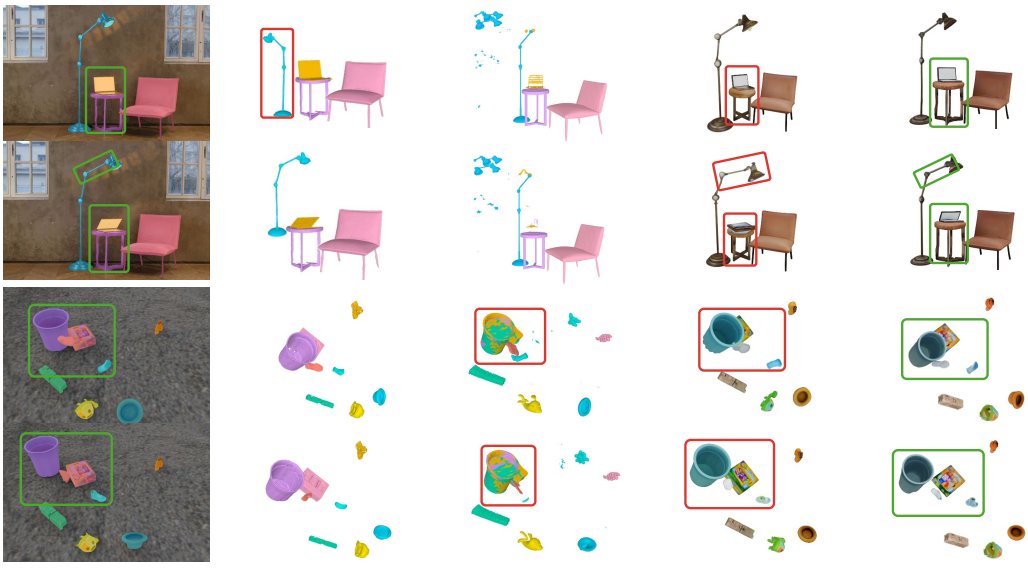


Figure 5: **Qualitative comparison of 4D scene reconstruction.** We present novel-view renderings of complete 4D scenes reconstructed by Chronofy and the competing baselines.

Table 2: **Quantitative comparison of 4D scene reconstruction.** Each Chrono-64 scene contains only one annotated dynamic object, so we do not show COM4D results. Best results are in **bold**.

		Chrono-64				Kubric-4D Test Set			
Generation	Model	3D IoU (↑)	ICP-Rot. (↓)	ADD-S (↓)	ADD-S @ 0.1 (↑)	3D IoU (↑)	ICP-Rot. (↓)	ADD-S (↓)	ADD-S @ 0.1 (↑)
Pipeline	ActionMesh + MegaPose	0.5496	16.2401	0.1132	0.7441	0.4694	32.2834	0.1625	0.5439
Pipeline	ActionMesh + FoundationPose	**0.6150**	14.7853	0.0578	0.8877	0.4713	30.7262	0.1431	0.6107
Pipeline	Chronofy + FoundationPose	0.6014	13.5789	0.0613	0.8496	0.4522	26.3855	0.1374	0.6756
Joint	COM4D	-	-	-	-	0.0612	39.2516	1.4747	0.0245
Joint	SAM3D	0.5862	13.9937	0.0477	0.9414	0.5776	16.5672	**0.0592**	**0.8728**
Joint	**Chronofy**	0.5977	**12.5925**	**0.0461**	**0.9463**	**0.5817**	**15.4926**	0.0771	0.8649

5.3 Ablation Study

We build Chronofy on top of SAM3D by introducing four components: (i) the spatiotemporal mixture of transformers (MoT); (ii) the dynamic Gaussian decoder; (iii) temporal masking augmentation; and (iv) layout test-time optimization (TTO). We evaluate each component's incremental contribution through cumulative ablations on the Chrono-64 dataset, as shown in Figure 6 and Table 3. Adding the spatiotemporal MoT improves temporal consistency but is accompanied by reduced occlusion robustness, possibly due to limitations in 4D data scale. The dynamic Gaussian decoder further lowers FVD and reduces visible inter-frame jitter, particularly in regions with complex textures. Temporal masking augmentation partially restores SAM3D's original occlusion robustness. As shown in Table 3, these three components improve 4D object generation while retaining pose estimation accuracy comparable to SAM3D. Finally, layout TTO refines the predicted poses through render-and-compare, increasing 2D mask IoU while also providing gains in 3D layout metrics.

Table 3: **Ablation study of Chronofy on the Chrono-64 dataset.** Variant (c) does not affect layout accuracy, while (e) does not affect the quality of generated 4D objects. Best results are in **bold**.

		4D Object Quality					Layout Accuracy		
	Model	LPIPS (↓)	CLIP-S (↑)	PSNR (↑)	SSIM (↑)	FVD (↓)	3D IoU (↑)	ICP-Rot. (↓)	2D IoU (↑)
(a)	SAM3D	0.2075	0.8392	13.1902	**0.7725**	724.04	0.5862	13.9937	0.5345
(b)	+ Spatiotemporal MoT	0.1742	0.8493	13.4375	0.7042	403.62	0.5589	13.1760	0.5304
(c)	+ Dynamic Gaussian Decoder	0.1809	0.8446	12.8691	0.6927	289.36	-	-	-
(d)	+ Temporal masking Augmentation	**0.1635**	**0.8512**	**13.9488**	0.7647	**263.19**	0.5841	12.7263	0.5379
(e)	+ Layout TTO	-	-	-	-	-	**0.5977**	**12.5925**	**0.5910**

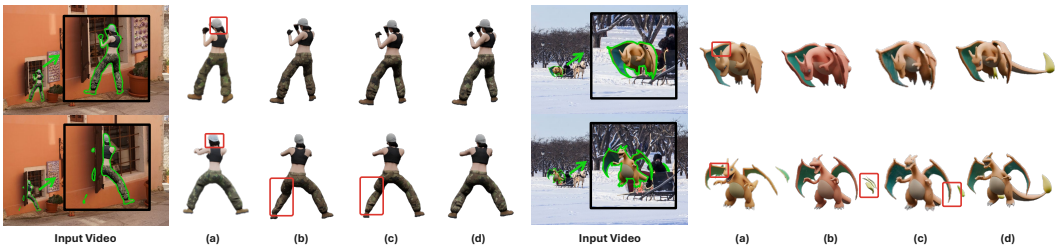


Figure 6: **Qualitative comparison with different model variants on Chrono-64 dataset.**

6 Conclusion

In this paper, we present **Chronofy**, a novel 4D generative reconstruction framework. To the best of our knowledge, Chronofy is the first framework to jointly generate complete 4D objects and explicit camera-space poses from single-view videos, enabling spatially aligned dynamic scene reconstruction and offering a new perspective on 4D generative reconstruction. We first analyze two key challenges in extending static 3D generative reconstruction to 4D: the need to coordinate object deformation and pose transformations over time, and the scarcity of real-world 4D training data. To address these issues, we introduce three novel architectural improvements to realize the generative reconstruction of dynamic objects, and employ two strategies to enhance the robustness and generalization of our method in real-world dynamic scenes. Extensive experiments demonstrate that Chronofy substantially improves temporal consistency in 4D object generation while retaining some occlusion robustness, and achieves competitive performance in 4D scene reconstruction.

AI USE STATEMENT

Large Language Models (LLMs) were used to assist with manuscript writing and language polishing. Specifically, we employed LLMs to refine the wording, improve readability, and enhance the clarity and overall flow of the paper, including tasks such as sentence rephrasing and grammar checking. LLMs also assisted in designing scripts for constructing portions of the training and test datasets and in conducting literature searches to identify a more comprehensive set of baselines for experimental comparison. LLMs were not involved in formulating the research ideas, developing the methodology or designing the core algorithms.

REPRODUCIBILITY STATEMENT

The source assets and datasets used in this work are publicly available. SAM3D, which serves as the foundation of our method, is also publicly accessible through its official GitHub repository. In Appendix A.3, we provide detailed descriptions of the training data processing pipeline and training configurations, while Appendix A.4 specifies the evaluation datasets and metrics. Upon acceptance, we will release our Chrono-64 dataset, training and inference code to further facilitate reproducibility and future research.

REFERENCES

Sherwin Bahmani, Ivan Skorokhodov, Victor Rong, Gordon Wetzstein, Leonidas Guibas, Peter Wonka, Sergey Tulyakov, Jeong Joon Park, Andrea Tagliasacchi, and David B Lindell. 4d-fy: Text-to-4d generation using hybrid score distillation sampling. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7996–8006. IEEE, 2024.

Blender Development Team. Blender, version 4.2.0. Blender Foundation, 2024. URL `https://www.blender.org`.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris Coll-Vinent, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. In *International Conference on Learning Representations*, volume 2026, pp. 138846–138923, 2026.

Jiahao Chang, Chongjie Ye, Yushuang Wu, Yuantao Chen, Yidan Zhang, Zhongjin Luo, Chenghong Li, Yihao Zhi, and Xiaoguang Han. Reconviagen: Towards accurate multi-view 3d object reconstruction via generation. In *International Conference on Learning Representations*, volume 2026, pp. 90387–90401, 2026.

Hongyuan Chen, Xingyu Chen, Zexiang Xu, and Anpei Chen. Motion 3-to-4: 3d motion reconstruction for 4d synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 28947–28958, 2026a.

Xingyu Chen, Fu-Jen Chu, Pierre Gleize, Kevin J Liang, Alexander Sax, Hao Tang, Weiyao Wang, Michelle Guo, Thibaut Hardin, Xiang Li, et al. Sam 3d: 3dfy anything in images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7220–7232, 2026b.

Alvaro Collet, Ming Chuang, Pat Sweeney, Don Gillett, Dennis Evseev, David Calabrese, Hugues Hoppe, Adam Kirk, and Steve Sullivan. High-quality streamable free-viewpoint video. *ACM Transactions on Graphics (ToG)*, 34(4):1–13, 2015.

Matt Deitke, Ruoshi Liu, Matthew Wallingford, Huong Ngo, Oscar Michel, Aditya Kusupati, Alan Fan, Christian Laforte, Vikram Voleti, Samir Yitzhak Gadre, et al. Objaverse-xl: A universe of 10m+ 3d objects. *Advances in Neural Information Processing Systems*, 36:35799–35813, 2023a.

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. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13142–13153, 2023b.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, et al. Emerging properties in unified multimodal pretraining. *arXiv preprint arXiv:2505.14683*, 2025.

Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B. McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 2553–2560. IEEE, 2022.

Michael Goesele, Brian Curless, and Steven M Seitz. Multi-view stereo revisited. In *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'06)*, volume 2, pp. 2402–2409. IEEE, 2006.

Ahmet Berke Gökmen, Ajad Chhatkuli, Luc Van Gool, and Danda Pani Paudel. Inferring compositional 4d scenes without ever seeing one. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 317–329, 2026.

Bastian Goldluecke and Marcus Magnor. Space-time isosurface evolution for temporally coherent 3d reconstruction. In *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004.*, volume 1, pp. I–I. IEEE, 2004.

Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J. Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, Thomas Kipf, Abhijit Kundu, Dmitry Lagun, Issam Laradji, Hsueh-Ti Liu, Henning Meyer, Yishu Miao, Derek Nowrouzezahrai, Cengiz Oztireli, Etienne Pot, Noha Radwan, Daniel Rebain, Sara Sabour, Mehdi S. M. Sajjadi, Matan Sela, Vincent Sitzmann, Austin Stone, Deqing Sun, Suhani Vora, Ziyu Wang, Tianhao Wu, Kwang Moo Yi, Fangcheng Zhong, and Andrea Tagliasacchi. Kubric: A scalable dataset generator. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022.

Richard Hartley, Andrew Zisserman, et al. *Multiple View Geometry in Computer Vision*, volume 2. Cambridge University Press, 2003.

Byeongho Heo, Song Park, Dongyoon Han, and Sangdoo Yun. Rotary position embedding for vision transformer. In *European Conference on Computer Vision*, pp. 289–305. Springer, 2024.

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022.

Yicong Hong, Kai Zhang, Jiuxiang Gu, Sai Bi, Yang Zhou, Difan Liu, Feng Liu, Kalyan Sunkavalli, Trung Bui, and Hao Tan. Lrm: Large reconstruction model for single image to 3d. In *International Conference on Learning Representations*, volume 2024, pp. 50678–50702, 2024.

Yu Hu, Chong Cheng, Sicheng Yu, Xiaoyang Guo, and Hao Wang. Vggt4d: Mining motion cues in visual geometry transformers for 4d scene reconstruction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 414–424, 2026.

Binbin Huang, Haobin Duan, Yiqun Zhao, Zibo Zhao, Yi Ma, and Shenghua Gao. Cupid: Generative 3d reconstruction via joint object and pose modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12741–12752, 2026.

Zeren Jiang, Chuanxia Zheng, Iro Laina, Diane Larlus, and Andrea Vedaldi. Mesh4d: 4d mesh reconstruction and tracking from monocular video. *arXiv preprint arXiv:2601.05251*, 2026.

Wolfgang Kabsch. A solution for the best rotation to relate two sets of vectors. *Foundations of Crystallography*, 32(5):922–923, 1976.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, George Drettakis, et al. 3d gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139–1, 2023.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Minkyung Kwon, Jinhyeok Choi, Youngjin Shin, Jaeyeong Kim, JongMin Lee, and Seungryong Kim. Morphos: Autoregressive 4d generation with temporal structured latents. *arXiv preprint arXiv:2606.02491*, 2026.

Yann Labbé, Lucas Manuelli, Arsalan Mousavian, Stephen Tyree, Stan Birchfield, Jonathan Tremblay, Justin Carpentier, Mathieu Aubry, Dieter Fox, and Josef Sivic. Megapose: 6d pose estimation of novel objects via render & compare. *arXiv preprint arXiv:2212.06870*, 2022.

Yangguang Li, Zi-Xin Zou, Zexiang Liu, Dehu Wang, Yuan Liang, Zhipeng Yu, Xingchao Liu, Yuan-Chen Guo, Ding Liang, Wanli Ouyang, et al. Triposg: High-fidelity 3d shape synthesis using large-scale rectified flow models. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025a.

Zhibing Li, Mengchen Zhang, Tong Wu, Jing Tan, Jiaqi Wang, and Dahua Lin. Ss4d: Native 4d generative model via structured spacetime latents. *ACM Transactions on Graphics (TOG)*, 44(6): 1–12, 2025b.

Weixin Liang, Lili Yu, Liang Luo, Srinivasan Iyer, Ning Dong, Chunting Zhou, Gargi Ghosh, Mike Lewis, Wen-tau Yih, Luke Zettlemoyer, et al. Mixture-of-transformers: A sparse and scalable architecture for multi-modal foundation models. *arXiv preprint arXiv:2411.04996*, 2024.

Chenguo Lin, Yuchen Lin, Panwang Pan, Yifan Yu, Tao Hu, Honglei Yan, Katerina Fragkiadaki, and Yadong Mu. Movies: Motion-aware 4d dynamic view synthesis in one second. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 295–306, 2026.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. *arXiv preprint arXiv:2210.02747*, 2022.

Yehonathan Litman, Xiaoxuan Ma, Manan Shah, Nicolas Ugrinovic, Kris Kitani, Fernando De la Torre, and Shubham Tulsiani. Lift4d: Harmonizing single-view 3d estimation for 4d reconstruction in-the-wild. *arXiv preprint arXiv:2606.23688*, 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.

Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9992–10002. IEEE, 2021.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3d gaussians: Tracking by persistent dynamic view synthesis. In *2024 International Conference on 3D Vision (3DV)*, pp. 800–809. IEEE, 2024.

F. Landis Markley, Yang Cheng, John L. Crassidis, and Yaakov Oshman. Averaging quaternions. *Journal of Guidance, Control, and Dynamics*, 30(4):1193–1197, 2007.

Yanxu Meng, Haoning Wu, Ya Zhang, and Weidi Xie. Scenegen: Single-image 3d scene generation in one feedforward pass. In *2026 International Conference on 3D Vision (3DV)*, pp. 543–553. IEEE, 2026.

Armin Mustafa, Hansung Kim, Jean-Yves Guillemaut, and Adrian Hilton. Temporally coherent 4d reconstruction of complex dynamic scenes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 4660–4669, 2016.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 724–732, 2016.

Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*, 2022.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763. PmLR, 2021.

Jiawei Ren, Liang Pan, Jiaxiang Tang, Chi Zhang, Ang Cao, Gang Zeng, and Ziwei Liu. Dreamgaussian4d: Generative 4d gaussian splatting. *arXiv preprint arXiv:2312.17142*, 2023.

Jiawei Ren, Kevin Xie, Ashkan Mirzaei, Hanxue Liang, Xiaohui Zeng, Karsten Kreis, Ziwei Liu, Antonio Torralba, Sanja Fidler, Seung W Kim, et al. L4gm: Large 4d gaussian reconstruction model. *Advances in Neural Information Processing Systems*, 37:56828–56858, 2024.

Remy Sabathier, David Novotny, Niloy J Mitra, and Tom Monnier. Actionmesh: Animated 3d mesh generation with temporal 3d diffusion. *arXiv preprint arXiv:2601.16148*, 2026.

Johannes L Schönberger, Enliang Zheng, Jan-Michael Frahm, and Marc Pollefeys. Pixelwise view selection for unstructured multi-view stereo. In *European conference on computer vision*, pp. 501–518. Springer, 2016.

Yawar Siddiqui, Duncan Frost, Samir Aroudj, Armen Avetisyan, Henry Howard-Jenkins, Daniel DeTone, Pierre Moulon, Qirui Wu, Zhengqin Li, Julian Straub, et al. Shaper: Robust conditional 3d shape generation from casual captures. *arXiv preprint arXiv:2601.11514*, 2026.

Edgar Sucar, Zihang Lai, Eldar Insafutdinov, and Andrea Vedaldi. Dynamic point maps: A versatile representation for dynamic 3d reconstruction. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7295–7305. IEEE, 2025.

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, 2024.

Carlo Tomasi and Takeo Kanade. Shape and motion from image streams under orthography: a factorization method. *International journal of computer vision*, 9(2):137–154, 1992.

Thomas Unterthiner, Sjoerd Van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges. *arXiv preprint arXiv:1812.01717*, 2018.

Basile Van Hoorick, Rundi Wu, Ege Ozguroglu, Kyle Sargent, Ruoshi Liu, Pavel Tokmakov, Achal Dave, Changxi Zheng, and Carl Vondrick. Generative camera dolly: Extreme monocular dynamic novel view synthesis. In *European Conference on Computer Vision*, pp. 313–331. Springer, 2024.

Vikram Voleti, Chun-Han Yao, Mark Boss, Adam Letts, David Pankratz, Dmitry Tochilkin, Christian Laforte, Robin Rombach, and Varun Jampani. Sv3d: Novel multi-view synthesis and 3d generation from a single image using latent video diffusion. In *European Conference on Computer Vision*, pp. 439–457. Springer, 2024.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5294–5306. IEEE, 2025a.

Ruicheng Wang, Sicheng Xu, Cassie Dai, Jianfeng Xiang, Yu Deng, Xin Tong, and Jiaolong Yang. Moge: Unlocking accurate monocular geometry estimation for open-domain images with optimal training supervision. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5261–5271. IEEE, 2025b.

Shizun Wang, Zhenxiang Jiang, Xingyi Yang, and Xinchao Wang. C4d: 4d made from 3d through dual correspondences. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 7570–7580. IEEE, 2025c.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 20697–20709. IEEE, 2024.

Yifan Wang, Peishan Yang, Zhen Xu, Jiaming Sun, Zhanhua Zhang, Yong Chen, Hujun Bao, Sida Peng, and Xiaowei Zhou. Freetimegs: Free gaussian primitives at anytime anywhere for dynamic scene reconstruction. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21750–21760. IEEE, 2025d.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Bowen Wen, Wei Yang, Jan Kautz, and Stan Birchfield. Foundationpose: Unified 6d pose estimation and tracking of novel objects. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17868–17879. IEEE, 2024.

Tianhao Wu, Chuanxia Zheng, Frank Guan, Andrea Vedaldi, and Tat-Jen Cham. Amodal3r: Amodal 3d reconstruction from occluded 2d images. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 9181–9193. IEEE, 2025.

Jianfeng Xiang, Zelong Lv, Sicheng Xu, Yu Deng, Ruicheng Wang, Bowen Zhang, Dong Chen, Xin Tong, and Jiaolong Yang. Structured 3d latents for scalable and versatile 3d generation. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21469–21480. IEEE, 2025.

Yu Xiang, Tanner Schmidt, Venkatraman Narayanan, and Dieter Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. *arXiv preprint arXiv:1711.00199*, 2017.

Jiale Xu, Weihao Cheng, Yiming Gao, Xintao Wang, Shenghua Gao, and Ying Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024.

Zhen Xu, Zhengqin Li, Zhao Dong, Xiaowei Zhou, Richard Newcombe, and Zhaoyang Lv. 4dgt: Learning a 4d gaussian transformer using real-world monocular videos. *arXiv preprint arXiv:2506.08015*, 2025.

Yuxue Yang, Lue Fan, Ziqi Shi, Junran Peng, Feng Wang, and Zhaoxiang Zhang. Neoverse: Enhancing 4d world model with in-the-wild monocular videos. *arXiv preprint arXiv:2601.00393*, 2026.

Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. In *International Conference on Learning Representations*, volume 2024, pp. 9142–9159, 2024.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *International Conference on Learning Representations*, volume 2025, pp. 83048–83077, 2025.

Vickie Ye, Ruilong Li, Justin Kerr, Matias Turkulainen, Brent Yi, Zhuoyang Pan, Otto Seiskari, Jianbo Ye, Jeffrey Hu, Matthew Tancik, et al. gsplat: An open-source library for gaussian splatting. *Journal of Machine Learning Research*, 26(34):1–17, 2025.

Jiraphon Yenphraphai, Ashkan Mirzaei, Jianqi Chen, Jiaxu Zou, Sergey Tulyakov, Raymond A Yeh, Peter Wonka, and Chaoyang Wang. Shapegen4d: Towards high quality 4d shape generation from videos. In *International Conference on Learning Representations*, volume 2026, pp. 118891–118909, 2026.

Yuyang Yin, Dejia Xu, Zhangyang Wang, Yao Zhao, and Yunchao Wei. 4dgen: Grounded 4d content generation with spatial-temporal consistency. *arXiv preprint arXiv:2312.17225*, 2023.

Yifei Zeng, Yanqin Jiang, Siyu Zhu, Yuanxun Lu, Youtian Lin, Hao Zhu, Weiming Hu, Xun Cao, and Yao Yao. Stag4d: Spatial-temporal anchored generative 4d gaussians. In *European Conference on Computer Vision*, pp. 163–179. Springer, 2024.

Bowen Zhang, Sicheng Xu, Chuxin Wang, Jiaolong Yang, Feng Zhao, Dong Chen, and Baining Guo. Gaussian variation field diffusion for high-fidelity video-to-4d synthesis. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 12502–12513. IEEE, 2025a.

Chuhan Zhang, Guillaume Le Moing, Skanda Koppula, Ignacio Rocco, Liliane Momeni, Junyu Xie, Shuyang Sun, Rahul Sukthankar, Joëlle K Barral, Raia Hadsell, et al. Efficiently reconstructing dynamic scenes one d4rt at a time. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7382–7392, 2026.

Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Forrester Cole, Deqing Sun, Ming-Hsuan Yang, et al. Monst3r: A simple approach for estimating geometry in the presence of motion. In *International Conference on Learning Representations*, volume 2025, pp. 82863–82886, 2025b.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

Zibo Zhao, Zeqiang Lai, Qingxiang Lin, Yunfei Zhao, Haolin Liu, Shuhui Yang, Yifei Feng, Mingxin Yang, Sheng Zhang, Xianghui Yang, et al. Hunyuan3d 2.0: Scaling diffusion models for high resolution textured 3d assets generation. *arXiv preprint arXiv:2501.12202*, 2025.

Yi Zhou, Connelly Barnes, Jingwan Lu, Jimei Yang, and Hao Li. On the continuity of rotation representations in neural networks. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5738–5746. IEEE, 2019.

A APPENDIX

We present additional ablation studies in Appendix A.1 and further quantitative and qualitative evaluations in Appendix A.2. Appendix A.3 describes the implementation details, including the training data, network architecture, and training objectives. The evaluation datasets and metrics are detailed in Appendix A.4. We elaborate on temporal masking augmentation in Appendix A.5 and describe the layout test-time optimization details in Appendix A.6. Finally, Appendix A.7 discusses the limitations of our framework and directions for future work. The supplementary material will include sample videos.

A.1 ADDITIONAL ABLATION STUDIES

Ablations on 4D representations. Following the temporal variation analysis in Figure 3, we conduct a controlled first-stage voxel-and-pose generation ablation to investigate how the 4D representation affects learning. We compare three alternatives: (a) frame-wise normalization with per-frame poses, as in SAM3D (Chen et al., 2026b); (b) rigid alignment (Kabsch, 1976) to the first frame followed by joint normalization, with per-frame poses recovering the alignment transformations; and (c) our sequence-level normalization with a unified pose. All variants use the same sparse-structure encoder and spatiotemporal mixture of transformers, trained on the same 2,000 sequences for 4000 steps with identical initialization and optimization settings. We transform the predicted voxel sequences into a common camera coordinate system using their predicted poses and evaluate them against ground truth using silhouette 2D IoU and voxel IoU, with identical rendering and voxelization settings across variants. As shown in Table 4, our representation achieves the highest scores on both metrics, indicating more accurate camera-space reconstruction. The qualitative comparisons in Figure 7 further show closer silhouette alignment and fewer abrupt changes in the posed geometry across frames, supporting the effectiveness of our representation for learning accurate and temporally coherent coarse 4D geometry.

Table 4: **Ablation study of 4D representations on Chrono-64 dataset.** Voxel IoU measures 3D occupancy overlap, while 2D IoU measures silhouette overlap in the input view.

Type	Voxel IoU (↑)	2D IoU (↑)
(a) SAM3D	0.0532	0.2069
(b) Rigid Alignment	0.0846	0.2972
(c) Ours	**0.1512**	**0.4420**

Ablations on dynamic Gaussian decoder. Our dynamic Gaussian decoder provides a substantial gain in the temporal consistency of generated 4D objects and comprises three key designs: (i) attention inflation; (ii) dynamic Gaussian representation; and (iii) mesh-guided velocity loss. On the Chrono-64 dataset, we use the original 3DGS decoder of SAM3D (Chen et al., 2026b) as a baseline and incrementally add these components to evaluate their incremental contributions using ground-truth structured latents, as reported in Table 5. Adding attention inflation enables information interaction among the structured latents of different frames. This significantly lowers the FVD and serves as the primary source of our dynamic Gaussian decoder's capability. Replacing standard 3DGS (Kerbl et al., 2023) with the dynamic Gaussian representation reduces jitter in regions with complex textures. However, we observe that, as illustrated in Figure 8, without ground-truth velocity supervision, dynamic Gaussians are prone to producing inter-frame blur and artifacts, or degrading into independent per-frame 3D Gaussians. After incorporating the mesh-guided velocity loss, temporal jitter in the generated 4D objects in visually complex regions is notably alleviated.

Ablations on pointmap conditioning. Chronofy relies on pointmaps as conditions for the joint generation of 4D objects and their poses. To assess its robustness to imperfect geometric conditioning, we evaluate both SAM3D and Chronofy on Chrono-64 using either ground-truth pointmaps or pointmaps estimated by Depth Anything 3 (DA3) (Lin et al., 2025). Since estimated pointmaps may differ from ground truth in global scale, we quantitatively evaluate object reconstruction quality and provide qualitative comparisons of camera-space placement in the input view. Table 6 shows some degradation in rendering-based quality metrics when ground-truth pointmaps are replaced with DA3 estimates. This degradation may partly reflect the effect of pointmap errors on predicted poses, as



Figure 7: **Qualitative examples of ablations on 4D representations.** The top row shows selected input frames. Subsequent rows overlay predicted voxel geometry, transformed using the predicted poses, with ground-truth geometry in a shared camera coordinate system.

Table 5: **Ablation for the Dynamic Gaussian Decoder.** We evaluate the decoder on the Chrono-64 dataset using **ground-truth structured latents**. Best results are in **bold**.

	Model	PSNR (↑)	LPIPS (↓)	SSIM (↑)	FVD (↓)
(a)	3DGS Decoder	29.46	0.04207	0.9480	473.82
(b)	+ Attention Inflation	30.77	0.04055	0.9564	194.43
(c)	+ Dynamic Gaussian Rep.	30.81	0.04082	0.9576	183.67
(d)	+ Mesh-guided Loss	**31.20**	**0.03967**	**0.9628**	**157.66**

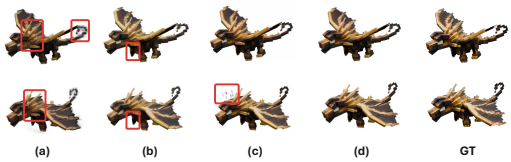


Figure 8: **Qualitative comparison of the Dynamic Gaussian Decoder ablation.**

spatial misalignment also affects comparisons between rendered and reference images. Nevertheless, the qualitative comparisons in Figure 9 show minimal visible differences in the reconstructed objects' shape and appearance, with changes more apparent in their alignment with the input observations. These results highlight Chronofy's robustness to imperfect geometric inputs: it largely preserves object fidelity even when ground-truth pointmaps are unavailable, despite variations in pose alignment and the associated rendering-based metrics. This observation is consistent with the SAM3D study (Chen et al., 2026b), which reports that pointmap conditioning has minimal impact on shape quality. Together, these findings suggest that Chronofy preserves the robustness of SAM3D's generative reconstruction capabilities while extending them to joint 4D object and pose generation.

Table 6: **Ablation study of pointmap conditioning on Chrono-64 dataset.** We compare SAM3D and Chronofy using ground-truth (GT) pointmaps or estimates from Depth Anything 3 (DA3).

Model	LPIPS (↓)	CLIP-S (↑)	PSNR (↑)	SSIM (↑)	FVD (↓)
SAM3D (DA3 pointmaps)	0.2299	0.8067	11.8883	0.7583	783.37
SAM3D (GT pointmaps)	0.2075	0.8392	13.1902	**0.7725**	724.04
Chronofy (DA3 pointmaps)	0.1798	0.8241	12.5063	0.7596	301.68
Chronofy (GT pointmaps)	**0.1635**	**0.8512**	**13.9488**	0.7647	**263.19**

Ablations on layout test-time optimization. Chronofy aims to reduce temporal jitter from independent frame-wise inference while preserving SAM3D's strong spatial alignment capabilities. Since layout test-time optimization (TTO) refines each object's pose independently, we evaluate its effect on per-object camera-space alignment on the Chrono-64 dataset. As shown in Table 7, layout TTO improves pose accuracy for both SAM3D and Chronofy, with particularly notable gains in

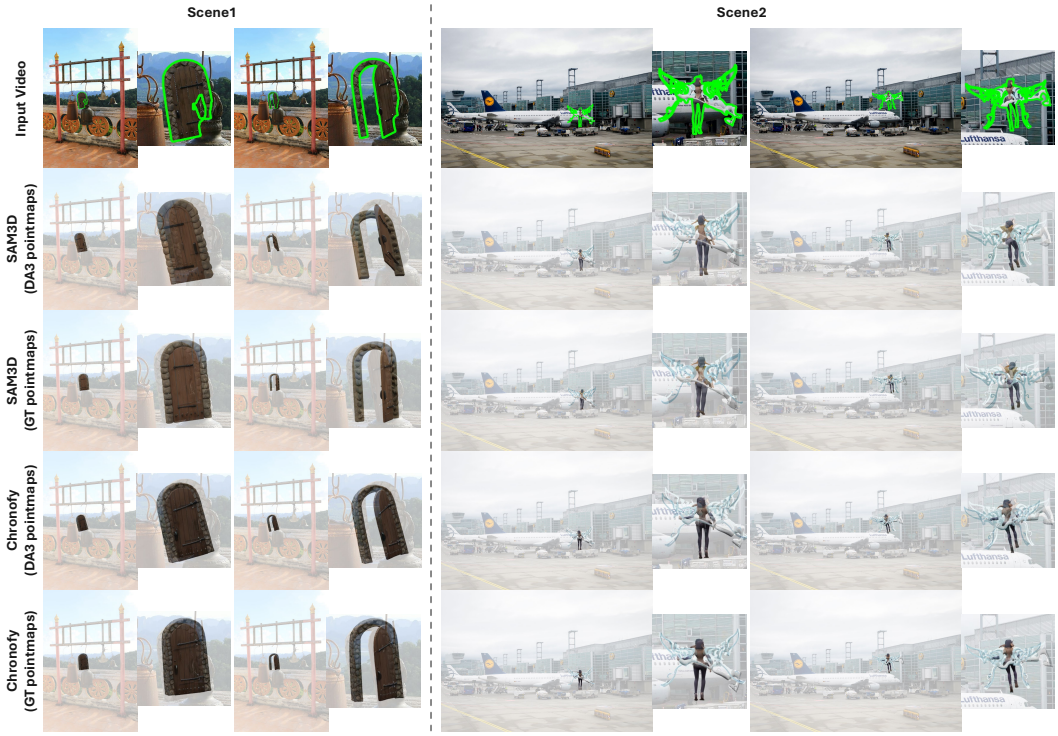


Figure 9: **Qualitative examples of ablations on pointmap conditioning.** Reconstructed objects are overlaid on the corresponding input images using their predicted poses, with both full-frame views and enlarged crops shown.

2D IoU, confirming that this refinement remains effective within our framework. Compared with SAM3D, Chronofy shows smaller gains in 3D IoU, ICP-Rot., and 2D IoU. This may partly reflect its sequence-wide refinement of a shared pose, which must balance alignment across frames and offers less flexibility for individual frames than SAM3D's frame-wise TTO. Although SAM3D (Chen et al., 2026b) performs better on most pose metrics after layout TTO, Chronofy maintains competitive spatial alignment while achieving substantially lower FVD (Table 1), consistent with improved temporal coherence. We hypothesize that the remaining pose accuracy gap partly reflects the limited scale and diversity of the synthetic 4D data used for adaptation, which may make it challenging to fully preserve SAM3D's pretrained alignment capabilities. Together with the qualitative comparisons, these results support Chronofy's ability to reduce temporal jitter while retaining competitive spatial alignment, effectively extending SAM3D to temporally coherent 4D reconstruction.

Table 7: **Ablation study of layout test-time optimization on Chrono-64 dataset.** We compare the layout accuracy of SAM3D and Chronofy with and without layout TTO.

Model	3D IoU (↑)	ICP-Rot. (↓)	ADD-S (↓)	ADD-S @ 0.1 (↑)	2D IoU (↑)
SAM3D (w/o Layout TTO)	0.5862	13.9937	0.0477	0.9414	0.5345
Chronofy (w/o Layout TTO)	0.5841	<u>12.7263</u>	0.0482	0.9395	0.5379
SAM3D (w/ Layout TTO)	**0.6036**	12.8537	<u>0.0470</u>	**0.9502**	**0.5988**
Chronofy (w/ Layout TTO)	<u>0.5977</u>	**12.5925**	**0.0461**	<u>0.9463</u>	<u>0.5910</u>

A.2 Additional Evaluations

User study. We conduct a user study to assess perceptual preferences for 4D object generation and 4D scene reconstruction. Participants compare Chronofy with each baseline in pairwise A/B comparisons and select the preferred result, considering visual quality, geometric completeness, and temporal consistency. Each questionnaire contains 16 examples for 4D object generation, with four examples for each of four baselines, and 15 examples for 4D scene reconstruction, with five

examples for each of three baselines. We collect 10 complete responses in total. Tables 8 and 9 report Chronofy's preference rate against each baseline. The results show that Chronofy is preferred in the majority of pairwise comparisons, consistent with the quantitative evaluation. Notably, although the frame-to-frame jitter of SAM3D (Chen et al., 2026b) is not fully reflected by the aforementioned metrics, it is perceptually noticeable in the user study, leading to a lower preference for SAM3D compared with its quantitative performance.

Table 8: **User study on 4D object generation.**

Model	Chronofy Win Rate (↑)
SS4D	75.00 %
MORPHOS	77.50 %
Motion324	90.00 %
SAM3D	82.50 %

Table 9: **User study on 4D scene reconstruction.**

Model	Chronofy Win Rate (↑)
ActionMesh + FoundationPose	68.00 %
COM4D	94.00 %
SAM3D	80.00 %

Inference time. We compare the average per-object inference times on the 64 objects in the Chrono-64 dataset, with all methods evaluated on an NVIDIA RTX A6000 GPU. Table 10 reports the detailed results. Chronofy takes slightly longer than SAM3D due to cross-frame information exchange. Nevertheless, it achieves better reconstruction performance with faster inference than COM4D (Gökmen et al., 2026) and the pipeline combining ActionMesh (Sabathier et al., 2026) with FoundationPose (Wen et al., 2024).

Table 10: **Quantitative evaluation of inference time for 4D scene reconstruction.** We report the average per-object inference time on the Chrono-64 dataset.

	ActionMesh + FoundationPose	COM4D	SAM3D	**Chronofy**
Inference time / s (↓)	270.12	284.25	**173.81**	198.21

Comparison with Lift4D. Our work focuses on adapting a pretrained 3D generative reconstruction model to jointly generate complete dynamic objects and unified camera-space poses. Specifically, our contribution lies in learning this joint generative capability within the 4D model itself. Lift4D (Litman et al., 2026) follows a different approach, retaining a frozen SAM3D prior and combining causal latent propagation with per-video Gaussian optimization. To complement the main evaluation, we present a dedicated quantitative and qualitative comparison on the Chrono-64 dataset. We report rendering-based metrics, since meshes decoded from Lift4D's initial SAM3D latents would not reflect its final Gaussian optimization. As shown in Table 11, Chronofy outperforms Lift4D across all reported quality metrics, with lower FVD consistent with improved temporal coherence. Chronofy also achieves substantially faster inference than Lift4D. To further assess reconstruction quality on real-world videos, we present qualitative comparisons on DAVIS in Figure 10. Across the illustrated viewpoints, Chronofy produces more complete object geometry and higher visual fidelity, with more consistent object structure across frames. These results support its ability to reconstruct coherent dynamic objects from real-world observations.

Table 11: **Quantitative comparison of 4D object generation with Lift4D on Chrono-64 dataset.**

Model	LPIPS (↓)	CLIP-S (↑)	SSIM (↑)	PSNR (↑)	FVD (↓)	Time (min) (↓)
Lift4D	0.1892	0.7627	0.7261	11.1364	438.54	140.7
Chronofy	**0.1635**	**0.8512**	**0.7647**	**13.9488**	**263.19**	**3.3**

Novel view synthesis. The 4D scene reconstruction metrics in Table 2 are computed on meshes decoded from our predicted structured latents using the SAM3D mesh decoder. These metrics assess geometric and layout accuracy but do not directly reflect the rendering quality of the dynamic Gaussians used in our actual reconstruction pipeline. Therefore, we additionally evaluate novel-view synthesis using our dynamic Gaussian decoder on the Kubric-4D (Van Hoorick et al., 2024) test set. Specifically, we reconstruct each dynamic scene with our method and SAM3D (Chen et al., 2026b) (other baselines do not produce textured results), and render three randomly selected novel views for each scene to evaluate the resulting synthesis quality. The selected viewpoints are kept fixed and

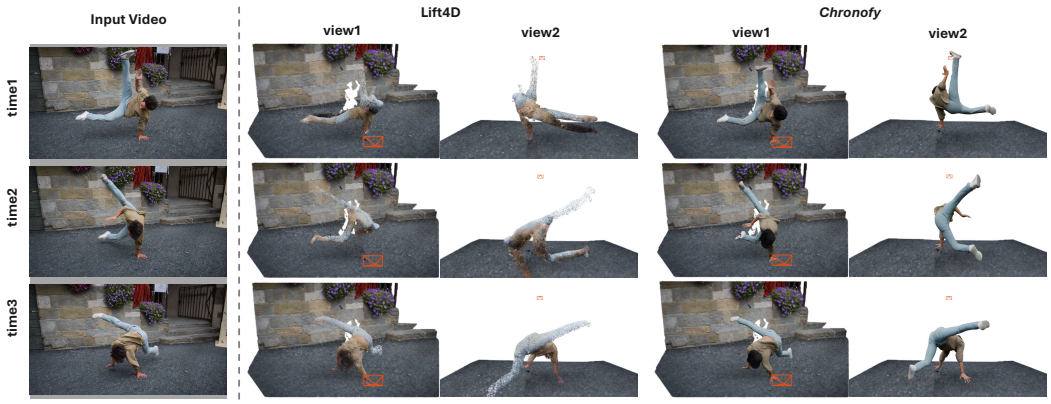


Figure 10: **Qualitative examples of Lift4D and Chronofy on real videos from DAVIS.**

shared across all methods. Since neither SAM3D nor Chronofy reconstructs the background, we evaluate the reconstruction performance by replacing the original background with a white background. For a comprehensive comparison, we additionally evaluate dynamic novel-view synthesis against two world-model baselines, MoVieS (Lin et al., 2026) and NeoVerse (Yang et al., 2026). The quantitative results are reported in Table 12. Despite achieving a substantial improvement in temporal consistency, as measured by video FVD, Chronofy largely maintains single-frame novel-view synthesis quality comparable to that of SAM3D. This result demonstrates the strong potential of our framework for generative reconstruction of realistic dynamic scenes. Qualitative comparisons are presented in Figure 11, where Chronofy consistently achieves the best visual quality. Additional qualitative comparisons are presented in Figure 5 and Figure 14.

Table 12: **Quantitative evaluation of novel-view synthesis for 4D scene reconstruction on the Kubric-4D test set**. We evaluate the visual quality of the complete Chronofy framework using image- and video-perception metrics. Best results are shown in **bold**.

Model	LPIPS (↓)	CLIP-S (↑)	SSIM (↑)	PSNR (↑)	FVD (↓)
MoVieS	0.3318	0.6045	<u>0.7431</u>	11.2189	728.65
NeoVerse	0.3269	0.6394	0.7335	**12.8034**	<u>542.48</u>
SAM3D	<u>0.2879</u>	**0.7836**	0.7355	11.6695	897.65
Chronofy	**0.2587**	<u>0.7337</u>	**0.7597**	<u>12.4570</u>	**285.57**



Figure 11: **Qualitative examples of novel-view synthesis for 4D scene reconstruction.**

Additional qualitative results. Figure 13 presents additional qualitative comparisons with video-to-4D baselines on the Chrono-64 dataset. We further provide scene-level comparisons with 4D reconstruction baselines across a broader range of scenarios in Figure 14. To further assess the applicability of Chronofy to in-the-wild videos, we evaluate it on the DAVIS dataset (Perazzi et al., 2016), as shown in Figure 15. No ground-truth pointmaps are used in this evaluation; instead, both the pointmap conditions and camera poses are estimated directly from the input videos using Depth

Anything 3 (Lin et al., 2025). Since DAVIS videos often contain non-negligible camera motion, we compensate for the estimated camera motion by transforming the per-frame object reconstructions into a shared world coordinate system. This separates viewpoint changes from intrinsic object dynamics and enables physically meaningful reconstruction of object motion. The results remain temporally coherent and geometrically stable, providing qualitative evidence of Chronofy's robustness under realistically obtainable geometric conditions.

A.3 Implementation Details

Training dataset. To train our generative reconstruction framework, we compile 18K dynamic 3D assets and render (Blender Development Team, 2024) 16-frame videos from Objaverse (Deitke et al., 2023b) and ObjaverseXL (Deitke et al., 2023a). For each sample, we compute a unified axis-aligned bounding box (AABB) and perform joint sequence normalization into a canonical $[-0.5, 0.5]^3$ space. For the input conditions, we render 8-view RGBA videos with 512×512 resolution as the random video and mask conditions. Simultaneously, we render the per-frame depth maps of the dynamic assets and back-project them to obtain the pointmap conditions. For the ground truth (GT) targets, we transform the dynamic assets from the object coordinate system to the corresponding view's camera coordinate system to obtain the ground truth pose. Meanwhile, following TRELLIS (Xiang et al., 2025), we voxelize the dynamic sequences in the canonical space and encode them into ground truth spatiotemporal structured latents. For temporal masking, we sample a target middle-frame visible ratio from $\mathcal{U}(0.2, 0.7)$ and use up to eight placement trials, selecting the closest candidate and keeping the selected scale and translation fixed across frames.

Training details. We initialize our models with the pretrained weights of SAM3D (Chen et al., 2026b). During training, we use a conditioning dropout rate of 0.1 for classifier-free guidance (CFG) (Ho & Salimans, 2022). The optimization (Loshchilov & Hutter, 2017) and attention configurations are summarized in Table 13. We fine-tune both generators for 40K steps each, following a two-stage curriculum from 8 to 16 frames. This process takes approximately 10 days on 8 NVIDIA A800 GPUs. The dynamic Gaussian decoder is trained for 2 days on 4 NVIDIA A800 GPUs.

Table 13: **Training configurations of our generative reconstruction framework.** Window shifts alternate between 0 and 4 frames across successive layers. Flow time is sampled independently per video and shared across its frames.

Setting	Spatiotemporal MoT	Spatiotemporal Flow Transformer	Dynamic Gaussian Decoder
Optimizer	AdamW	AdamW	AdamW
Learning rate	1×10^{-4}	1×10^{-4}	1×10^{-4}
Optimizer betas	$(0.9, 0.999)$	$(0.9, 0.999)$	$(0.9, 0.999)$
Weight decay	0.0	0.0	0.0
Batch size	32	32	8
Window size	8	-	8
Window shift (frames)	0 / 4	-	0 / 4
Objective	$\mathcal{L}_{\text{CFM}}$	$\mathcal{L}_{\text{CFM}}$	$\mathcal{L}_{dynamic\ Gaussian}$
EMA decay	0.9999	0.9999	0.9999
Gradient clipping	1.0	1.0	1.0
Frozen modules	Conditioner	Conditioner	None
Flow-time sampling	$\text{LogitNormal}(-1, 1)$	$\text{LogitNormal}(-1, 1)$	-
Flow-time sharing	Per video	Per video	-

Training objective. Both generators are trained using rectified conditional flow matching, with prediction modalities and conditioning inputs tailored to each stage. Each stage minimizes:

$$\mathcal{L}_{\text{CFM}} = \sum_{m \in M} \lambda_m \mathbb{E}_{\tau, x^m_{\text{gt}}} \left[\| v^m - v^m_\theta(x^m_\tau, c, \tau) \|_2^2 \right]. \tag{6}$$

In the first stage, the spatiotemporal mixture of transformers jointly predicts coarse shape latents, rotation, translation, and scale, with $M = \{\mathcal{S}, R, t, s\}$ and video, mask, and pointmap conditions $c = (\mathcal{V}, \mathcal{M}, \mathcal{P})$. The second-stage spatiotemporal flow transformer generates spatiotemporal structured latents, with $M = \{\mathcal{Z}\}$ and conditions $c = (O, \mathcal{V}, \mathcal{M})$ comprising the coarse voxel grid, video, and masks. Here, λ_m weights the loss for modality m, and v^m_θ denotes the corresponding learned velocity field (Lipman et al., 2022). For both stages, we linearly interpolate $x^m_\tau = (1-\tau)\epsilon^m + \tau x^m_{\text{gt}}$

with target velocity $v^m = x^m_{\text{gt}} - \epsilon^m$, where x^m_{gt} is the ground-truth target for modality m and $\epsilon^m \sim \mathcal{N}(0, I)$ is sampled independently across frames. Flow time τ is sampled based on Table 13.

The dynamic Gaussian decoder is trained using image-space reconstruction and mesh-guided motion supervision. We first compute a reconstruction loss between images $\mathcal{I}_t^{\text{render}}$ rendered from the predicted dynamic Gaussians and ground-truth images $\mathcal{I}_t^{\text{gt}}$:

$$\mathcal{L}_{\text{img}} = \mathcal{L}_1 + \lambda_{lpips}\mathcal{L}_{lpips} + \lambda_{ssim}\mathcal{L}_{ssim}, \tag{7}$$

where λ_{lpips} and λ_{ssim} weight the perceptual and SSIM losses, respectively. Inspired by GVFDiffusion (Zhang et al., 2025a), we further introduce mesh-guided velocity supervision to encourage meaningful Gaussian motion. Since mesh vertices correspond across frames, we estimate their velocities using finite differences with a normalized time step of $1/(T-1)$. For each Gaussian i generated from frame t, we identify its K nearest mesh vertices at its source-frame position and interpolate their velocities to obtain the approximate ground-truth velocity $\tilde{\mathbf{v}}_{t,i}$:

$$\mathcal{L}_{mg} = \frac{1}{N}\sum_{t,i} \left\| \mathbf{v}_{t,i} - \tilde{\mathbf{v}}_{t,i} \right\|_1, \tag{8}$$

where N is the total number of Gaussians used for velocity supervision. The complete decoder objective is

$$\mathcal{L}_{dynamic\ Gaussian} = \mathcal{L}_{img} + \lambda_{mg}\mathcal{L}_{mg} + \mathcal{L}_{reg}, \tag{9}$$

where λ_{mg} weights the mesh-guided velocity loss, and $\mathcal{L}_{\text{reg}}$ includes lightweight velocity, angular-velocity, life-span-support, Gaussian-volume, and opacity regularization terms.

Inference details. Both the spatiotemporal SS and SLAT generators use 25 sampling steps and a CFG interval (Ho & Salimans, 2022) of $[0, 500]$ on the 0–1000 time scale, with CFG strengths of 7.0 and 5.0, respectively. To obtain the unified pose, we first decode and denormalize the per-frame pose predictions into input pointmap coordinates. After enforcing isotropic scaling by averaging the three scale components and compensating for sparse-structure downsampling, we aggregate rotations using equal-weight Markley quaternion averaging (Markley et al., 2007), translations using arithmetic averaging, and scales using geometric averaging. This pose is shared across all frames.

A.4 Details on Evaluation

Datasets. We evaluate the quality of 4D object generation and 4D scene reconstruction on the existing Motion-80 (Chen et al., 2026a) and Kubric-4D (Van Hoorick et al., 2024) datasets, respectively. To further simulate realistic 4D scenes, we construct a pseudo-realistic dataset by combining Objaverse (Deitke et al., 2023b;a) with SA-3DAO (Chen et al., 2026b). Specifically:

- **Motion-80:** Motion-80 consists of 64 short motion sequences with 16 frames and 16 long-range motion sequences. Each sequence is accompanied by ground-truth mesh sequences and videos rendered from four viewpoints, $[0°, 90°, 180°, 270°]$, at a resolution of 512×512. These novel-view videos are used to evaluate the quality of 4D object generation. Since our framework is primarily designed for 16-frame 4D object generation, we evaluate only the short sequences in Motion-80. For both Chronofy and SAM3D, we render per-frame depth maps from the ground-truth meshes at the input viewpoint and back-project them using our rendering camera parameters to obtain conditioning pointmaps. This avoids unreliable Depth Anything 3 (Lin et al., 2025) estimates on these white-background sequences.
- **Kubric-4D:** Kubric-4D consists of 4D scenes composed of GSO objects (Downs et al., 2022) generated using the Kubric framework (Greff et al., 2022). Each scene provides multi-view camera parameters, single-view rendered videos, dynamic object pose trajectories, ground-truth pointmaps, and ground-truth meshes. The dataset is suitable for evaluating 6D pose estimation and 4D scene reconstruction. We evaluate our method on 100 test scenes.
- **Chrono-64:** Chrono-64 consists of 64 pseudo-realistic 4D scenes constructed by placing 64 dynamic objects from Objaverse, all held out from our training set, into scenes from SA-3DAO. Specifically, based on the pointmaps of the real static backgrounds in SA-3DAO, we appropriately place the dynamic objects within the background point-map space to establish plausible spatial configurations and occlusion relationships, while obtaining the corresponding ground-truth object poses. We then render videos from the known camera viewpoints of SA-3DAO

and obtain pointmaps that preserve the corresponding realistic occlusion relationships. In addition, we render novel-view videos of the objects by rotating the reference viewpoint clockwise by $[0°, 90°, 180°, 270°]$, which are used to evaluate the quality of 4D object generation.

4D object generation evaluation. For each predicted 4D object, we render the first frame at four candidate azimuths $\Theta = \{0°, 90°, 180°, 270°\}$ and select the reference azimuth that minimizes the ℓ_2 distance to the ground-truth input-view image of the first frame. The selected azimuth defines the aligned $0°$ reference, and this alignment is held fixed across all frames of the sequence. We exclude the input view from scoring and evaluate only the three aligned novel views at relative azimuths $\{90°, 180°, 270°\}$ against their corresponding ground-truth videos. For LPIPS, CLIP-S, PSNR, and SSIM, both predicted and ground-truth images are composited onto a white background, cropped using the same ground-truth alpha bounding box, and resized to 512×512. These image-level metrics are averaged over frames, novel views, and test instances. FVD is computed on full, uncropped 512×512 videos with 16 frames. For methods requiring pointmap conditioning, Motion-80 uses the pointmaps rendered from ground-truth meshes at the input viewpoint, as described above, whereas Chrono-64 uses the ground-truth pointmaps provided by the dataset.

4D scene reconstruction evaluation. Following SAM3D, we evaluate 4D scene reconstruction primarily using 6D pose estimation metrics and adopt the same metric definitions as SAM3D:

- **3D IoU:** 3D Intersection over Union measures the overlap between the predicted and ground-truth axis-aligned 3D bounding boxes based on their intersection and union. Its value ranges from 0, indicating no overlap, to 1, indicating complete overlap.
- **ICP-Rot.:** ICP Rotation Error measures the rotational correction estimated by ICP, reported in degrees. The predicted and ground-truth meshes are first transformed using their respective poses, after which ICP estimates the alignment rotation R_{ICP} from the predicted mesh to the ground-truth mesh. ICP-Rot. is computed as the angle of R_{ICP} relative to the identity.
- **ADD-S:** Following SAM3D, we adopt a symmetric, diameter-normalized variant of ADD-S (Xiang et al., 2017), which measures the bidirectional nearest-neighbor distance between the predicted and ground-truth posed point clouds. Specifically, given the predicted point cloud $\mathcal{M}$, the ground-truth point cloud $\mathcal{M}_{\mathrm{gt}}$, and the object diameter d, ADD-S is defined as:

$$\text{ADD-S} = \frac{\mathrm{ADD}(\mathcal{M}, \mathcal{M}_{\mathrm{gt}}) + \mathrm{ADD}(\mathcal{M}_{\mathrm{gt}}, \mathcal{M})}{2d}, \quad \mathrm{ADD}(\mathcal{A}, \mathcal{B}) = \frac{1}{|\mathcal{A}|} \sum_{\mathbf{x} \in \mathcal{A}} \min_{\mathbf{y} \in \mathcal{B}} \|\mathbf{x} - \mathbf{y}\|_2, \quad (10)$$

  where $d = \max_{\mathbf{x},\mathbf{y} \in \mathcal{M}_{\mathrm{gt}}} \|\mathbf{x} - \mathbf{y}\|_2$ denotes the object diameter. Lower ADD-S values indicate more accurate pose estimation.
- **ADD-S @ 0.1:** For each evaluated object-frame pair, this metric assigns a value of 1 if the diameter-normalized ADD-S is less than 0.1, and 0 otherwise.

Since our evaluation focuses on dynamic scenes containing multiple frames, we compute each metric independently for every object in every frame and report the arithmetic mean over all object–frame pairs across the evaluation scenes as the final performance. Note that the above metrics assess geometric overlap and alignment, whereas the dynamic Gaussians generated by our framework are optimized primarily for visual fidelity and are not explicitly constrained to preserve accurate point-level geometry. For a fair comparison, we therefore **decode the spatiotemporal structured latents generated by Chronofy into mesh sequences** using the SAM3D mesh decoder (Chen et al., 2026b), and compute the aforementioned metrics on the resulting meshes. Additionally, for all reconstruction methods that require pointmaps as conditioning inputs, we use identical pointmaps for each test sample to ensure a fair comparison under consistent geometric conditioning. Specifically, we use the ground-truth pointmaps provided by Kubric-4D and Chrono-64.

A.5 Details on Temporal Masking Augmentation

To improve robustness to dynamic occlusions, we synthesize occluded training clips by randomly sampling another dynamic object sequence as the occluder. The target and occluder are sampled from different objects, and each sequence uses a randomly selected but temporally fixed camera view. We resize and translate the occluder consistently across all frames, while its original motion

provides natural temporal variation. The placement is controlled according to the visible ratio of the target, encouraging the model to handle different occlusion levels.

Let I_t and M_t denote the target image and mask, and let $\widetilde{I}_t$ and $\widetilde{M}_t$ denote the transformed occluder image and mask at frame t. The visible target crop and the occlusion-aware context crop are then constructed through mask-guided alpha compositing as follows:

$$I_t^{\text{vis}} = I_t \odot (1 - \widetilde{M}_t), \quad M_t^{\text{vis}} = M_t \odot (1 - \widetilde{M}_t),$$
$$I_t^{\text{ctx}} = I_t \odot (1 - \widetilde{M}_t) + \widetilde{I}_t \odot \widetilde{M}_t. \tag{11}$$

Inspired by Amodal3R (Wu et al., 2025), we find that retaining evidence related to the occluded area in the condition is important for recovering the complete object. However, instead of introducing an additional occlusion-mask branch (e.g., M_{occ}), we reuse SAM3D's (Chen et al., 2026b) pretrained full-image conditioning pathway to encode I_t^{ctx}. In our framework, this condition is also a target-centric crop rather than the complete input frame. It contains the visible target together with the inserted occluder, allowing the model to extract occluder appearance and boundary cues.In parallel, the original crop condition contains only I_t^{vis}, providing complementary target-only cues.

The target mask and pointmap are updated to retain only visible target regions, with occluded pointmap locations marked as invalid, while the complete target shape remains unchanged as supervision. This design preserves SAM3D's original conditioning interface without adding a separate occlusion-mask encoder or modifying the network architecture, thereby providing effective occlusion robustness with minimal changes to the pretrained model.

A.6 Details on Layout Test-Time Optimization

We extend SAM3D's pose refinement (Chen et al., 2026b) to sequences by optimizing a single residual similarity transform per object, with its rotation, three-dimensional translation, and isotropic scale shared across all frames. The underlying generated 4D geometry, motion, and appearance parameters remain fixed during refinement. The objective averages foreground RGB and full-image mask MSE losses over frames, with a weight of 10 for each term. We regularize the normalized residual quaternion toward the identity quaternion, the translation toward zero, and the isotropic scale toward one, with weights of 0.1, 0.05, and 0.05, respectively. These regularizers softly anchor refinement to the pointmap-conditioned initialization, discouraging scale–depth drift without resolving the underlying ambiguity. Starting from an identity residual transform, we use Adam (Kingma & Ba, 2014) to optimize translation and scale for up to 5 iterations at a learning rate of 10^{-2}, followed by joint refinement of all three components for up to 25 iterations at 5×10^{-3}. Each stage stops early if the absolute change in the total objective falls below 10^{-6}. The refined pose remains shared across the sequence and is retained only if the mean per-frame mask IoU improves relative to initialization; otherwise, the initial pose is restored.

A.7 Failure Cases and Future Work

Failure cases. Figure 12 presents representative failure cases, including (a) failed shape completion and (b) implausible geometry generation. Recovering complete 4D objects and their scene layouts from a single-view video is inherently ill-posed: multiple combinations of geometry, motion, and pose can explain the same observations, making it difficult to learn reliable priors that disambiguate these factors. This challenge is further exacerbated by the scarcity of real-world training data with complete 4D shapes and accurate poses. Consequently, under severe occlusion or rapid pose transformation, Chronofy may generate incomplete or implausible shapes. Because shape and pose are jointly inferred, such shape errors may also degrade pose estimation and cause spatial misalignment.

Future work. Building on Chronofy's current capabilities, the following extensions could improve its robustness and broaden its applicability:

- **Richer 4D training data.** High-quality real-world 4D data are crucial for improving robustness and generalization, yet obtaining complete dynamic geometry together with accurate camera-space poses remains challenging and costly. Expanding training data with more diverse motions

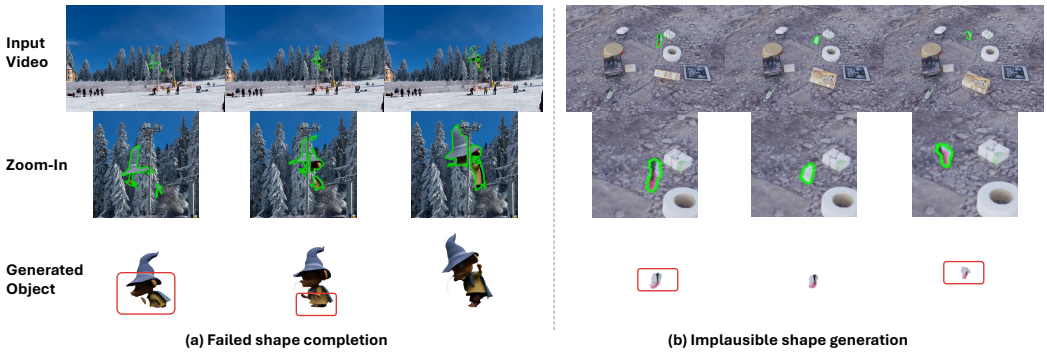


Figure 12: **Failure cases.** (a) Severe occlusion without complementary cues from surrounding frames makes complete shape recovery difficult, even with generative priors. (b) When objects occupy small image regions and rotate rapidly, Chronofy may generate implausible shapes.

and occlusions, while combining large-scale synthetic data with limited real-world 4D observations, could further strengthen Chronofy's generative priors.

- **Long-horizon generation.** Chronofy is trained on 16-frame sequences and can generate up to 32 frames, but substantially longer sequences increase inference cost and degrade generation quality and temporal consistency. Autoregressive generation conditioned on previous 4D states (Sabathier et al., 2026) could extend the supported duration, although errors may accumulate across segments. Inspired by causal video VAEs (Wan et al., 2025; Yang et al., 2025), a promising direction is a causally structured 4D VAE that performs temporal compression and models dependencies between successive 3D states to better preserve information over long horizons.



Figure 13: **Additional qualitative 4D object generation results of our framework on the Chrono-64 dataset.** For 4D generative methods that produce mesh sequences, we present both the original meshes and textured meshes, with textures generated using the official tools.



Figure 14: **Additional qualitative examples for dynamic scene reconstruction.** We include examples of Chronofy and alternative dynamic scene reconstruction methods with pointmaps predicted by the geometric model.



Figure 15: **Additional qualitative results on real videos from DAVIS.** For each example, the top row shows the input video, the middle row shows the generated 4D object placed in the background point cloud, and the bottom row shows the generated object alone. Conditioning pointmaps are predicted using Depth Anything 3.