ENVISION4D: ENVISIONING VISUAL FUTURES VIA FEED-FORWARD 4D GAUSSIAN SPLATTING FOR AUTONOMOUS DRIVING

Anonymous authors
Paper under double-blind review

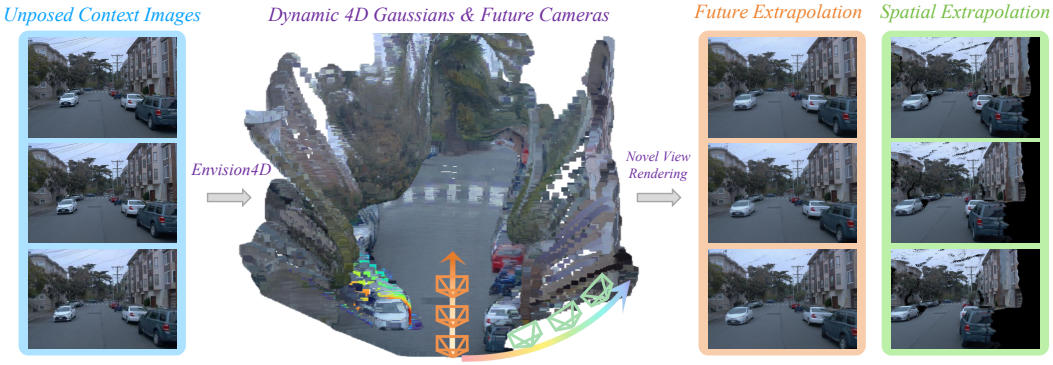


Figure 1: **Illustration of Envision4D.** Envision4D jointly predicts 4D Gaussians and future camera poses in a single forward pass, enabling efficient dynamic scene extrapolation without explicit motion supervision.

ABSTRACT

Forecasting the future evolution of dynamic scenes is crucial in autonomous driving. However, existing feed-forward Gaussian Splatting methods are primarily designed for scene interpolation. When extended to future extrapolation, they suffer from ghosting artifacts under large displacements and are constrained by simplified motion assumptions or strict future priors. To overcome these challenges, we propose Envision4D, a feed-forward framework for pose-free future extrapolation with self-supervised motion learning. Specifically, we introduce a Future Pose Prediction module that infers future camera parameters via an iterative refinement process. Furthermore, to capture non-linear dynamics, we propose In-layer Temporal Attention and employ Conditioned Motion Lifting, which transforms the highly uncertain extrapolation process into robust relational mappings. Finally, a Progressive Training Strategy is utilized to stabilize unsupervised motion learning against error accumulation. Extensive experiments demonstrate that Envision4D achieves state-of-the-art performance, significantly outperforming existing methods in future view synthesis.

1 INTRODUCTION

Modeling dynamic scenes is fundamental to autonomous driving (Yan et al., 2025; Song et al., 2024; Chen et al., 2024b; 2026), where understanding future scene evolution is important for further planning (Wang et al., 2024; Fu et al., 2025; Zhou et al., 2026). Despite this imperative, current feed-forward paradigms (Hur et al., 2026; Sucar et al., 2026; Hu et al., 2025; He et al., 2026; Yang et al., 2026) are primarily tailored for interpolating observed dynamics. When applied directly to future extrapolation, these interpolation-centric models struggle with increasing motion uncertainty, leading to severe error accumulation.

The challenges lie in two main aspects: (1) **Ineffective unsupervised motion modeling.** To fully unleash the potential of data-driven paradigms, methods like STORM (Yang et al., 2024a) and

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.

Flux4D (Wang et al., 2025b) explore unsupervised motion learning, bypassing the reliance on expensive explicit guidance such as optical flow (Lin et al., 2025; Yang et al., 2026; He et al., 2026), off-the-shelf trackers (Fei et al., 2024; Miao et al., 2026; Chen et al., 2025), or dynamic masks (Wu et al., 2025c; Yu et al., 2026). Unfortunately, without such strong priors, the unsupervised models struggle to handle large displacements, yielding severe ghosting artifacts around dynamic objects. (2) **Inflexible extrapolation mechanisms.** Existing feed-forward reconstruction methods often rely on simplified motion assumptions or strict future priors. For example, STORM (Yang et al., 2024a) and ReconDrive (Yu et al., 2026) propagate dynamic Gaussians using constant velocities and necessitate provided target camera poses for view synthesis. DGGT (Chen et al., 2025) instead estimates target camera poses under a linear camera-motion assumption. Extending these simplified object- and camera-motion models beyond the observed interval can amplify object displacement errors and ego-pose drift, degrading the spatial consistency of future reconstructions.

To address these issues, we propose Envision4D, a pose-free framework for dynamic scene extrapolation without explicit motion supervision. As illustrated in Fig. 1, Envision4D achieves robust future and spatial extrapolation in dynamic scenarios, successfully breaking free from the constraints of extra explicit guidance and restrictive future priors.

Specifically, to eliminate the reliance on given ego-trajectories, we introduce a *Future Pose Prediction* module that employs iterative refinement to adaptively infer future cameras. Built upon this, we rethink the velocity formulation by proposing time-conditioned motion, which is crucial for capturing real-world non-linear dynamics. Within this representation, *In-layer Temporal Attention* is first employed to enhance the network's sensitivity to dynamic cues. Furthermore, our *Conditioned Motion Lifting* mitigates extrapolation uncertainties by modeling source-to-target velocities conditioned on the current state, ego-motion, and temporal priors. Finally, a *Progressive Training Strategy* stabilizes the unsupervised motion learning, allowing the model to gradually refine motion-aware features and prevent the severe error accumulation typical of unconstrained extrapolation.

Our main contributions are summarized as follows:

- We propose Envision4D, a novel 4DGS model capable of dynamic scene extrapolation in a future pose-free manner, without requiring any explicit motion guidance.
- We address the reliance on future ego-trajectories and linear motion assumptions by jointly inferring future camera poses and time-conditioned, non-linear motions. A tailored training strategy is further introduced to stabilize unsupervised motion learning.
- Extensive experiments show that Envision4D achieves state-of-the-art performance in future extrapolation and exhibits promising generalization capabilities in open-world driving scenarios.

2 Related Work

Feed-forward Gaussian Splatting. Recent years have witnessed a rapid transition of Gaussian models from 3D (Yu et al., 2024; Szymanowicz et al., 2024; Jiang et al., 2025; Xu et al., 2025a) to 4D (Wu et al., 2024; Chen et al., 2026; Li et al., 2024a; Lin et al., 2025) scene reconstruction. For feed-forward 3D reconstruction, MVSplat (Chen et al., 2024a) leverages costvolume representations to enhance depth estimation, while ADGaussian (Song et al., 2025) integrates multi-modal cues for robust geometry and visual modeling. To improve practical flexibility, pose-free frameworks like GGRt (Li et al., 2024b) and NoPoSplat (Ye et al., 2024) enable efficient static reconstruction directly from unposed images. Building upon these 3D foundations, recent 4D Gaussian models extend the paradigm by incorporating motion modeling. Specifically, DynamicVGGT (He et al., 2026) and NeoVerse (Yang et al., 2026) introduce motion attention modules on the top of VGGT (Wang et al., 2025a) backbone to capture inter-frame motions. ReconDrive (Yu et al., 2026) utilizes SAM2 (Ravi et al., 2024) for instance-level static-dynamic decomposition and computes object displacements via coordinate transformations, while DGGT (Chen et al., 2025) employs external trackers for motion interpolation. Despite these advances, most feed-forward methods primarily focus on scene interpolation within observed frames, with limited exploration of extrapolation scenarios crucial for practical use. Moreover, they typically rely on pre-given future camera poses to perform extrapolation, failing to generalize to unseen open-world scenarios.

Unsupervised Dynamic Reconstruction. Reconstructing dynamic scenes without extra supervision, such as dynamic masks (Wang et al., 2025c; Yu et al., 2026), optical flow (Lin et al., 2025;

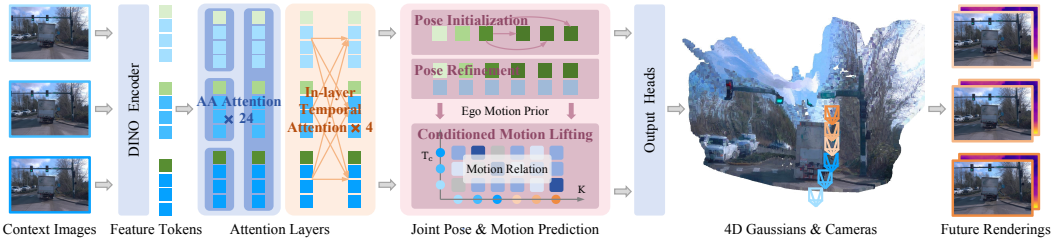


Figure 2: **Framework of Envision4D.** Given a sequence of context images, Envision4D predicts 4D Gaussians and all target camera poses. The motion awareness of feature tokens is first enhanced via *In-layer Temporal Attention*. Subsequently, *Joint Pose-Motion Prediction* is applied to enable future pose estimation through iterative refinement, alongside non-linear motion generation via conditioned motion lifting. The generated tokens are then decoded to render novel future views.

Yang et al., 2026; He et al., 2026), or pre-trained trackers (Fei et al., 2024; Chen et al., 2025), remains a highly challenging task. Recent works have explored self-supervised, scene-optimized strategies to decompose dynamic objects. Specifically, methods like S^3Gaussian (Huang et al., 2024) and EvoGS (Asiimwe & Vondrick, 2025) capture scene dynamics by learning a spatial-temporal hexplane representation. Despite high fidelity, these methods require time-consuming per-scene optimization, limiting their scalability. To address this, another group of approaches turns to exploring the feed-forward unsupervised dynamic reconstruction. Some methods, such as V-DPM (Sucar et al., 2026) and BTimer (Liang et al., 2024), implicitly learn dynamic reconstruction at given target timestamps utilizing time-conditioned transformer blocks. In contrast, other feed-forward methods, e.g., STORM (Yang et al., 2024a) and 4DGT (Xu et al., 2025b) explicitly predict motion parameters for each 3D Gaussian. However, these approaches only estimate instantaneous motion vectors at the current timestamp and move Gaussians with a simplified linear motion assumption, which struggles with accumulated trajectory drift over extended temporal horizons.

Future Scene Prediction. Future scene prediction has been extensively explored in video generation models (Ho et al., 2022; Hong et al., 2022; Blattmann et al., 2023; Gao et al., 2024; Wu et al., 2025b; Zhang et al., 2025). Representative foundation models, such as Sora (Liu et al., 2024), CogVideoX (Yang et al., 2024b), Cosmos (Ali et al., 2025), and Wan (Wan et al., 2025), have demonstrated remarkable visual synthesis capabilities. Beyond pixel-space generation, DINO-World (Baldassarre et al., 2025), DINO-Foresight (Karypidis et al., 2026), and VGGT-World (Sun et al., 2026) model future evolution in the feature space of frozen visual or geometry foundation models. Recent works such as Gen3R (Huang et al., 2026), FantasyWorld (Dai et al., 2025), and Geometry Forcing (Wu et al., 2025a) further incorporate geometric priors from 3D reconstruction to improve structural and view consistency. Despite incorporating geometric clues, these methods do not yield a unified renderable 4D scene representation. In contrast, Envision4D extrapolates future scenes directly in 4D Gaussian space, jointly representing geometry, appearance, and temporal motion.

3 METHOD

The methodology is organized as follows. We first present the task formulation (Sec. 3.1). We then introduce Envision4D, a feed-forward 4D Gaussian Splatting framework for dynamic scene extrapolation. Envision4D consists of three key components: a frozen VGGT encoder with in-layer temporal attention to formulate a motion-biased feature space (Sec. 3.2), a joint pose-motion prediction module to facilitate future scene extrapolation in a pose-free manner (Sec. 3.3), and a progressive training strategy to optimize the unsupervised motion learning process (Sec. 3.4).

3.1 TASK FORMULATION

As shown in Fig. 2, we formulate future scene reconstruction as a pose-free and self-supervised motion learning task. Given a sequence of T_c context images, Envision4D aims to reconstruct the dynamic scene and predict its evolution over a future horizon of T_f frames. For each observed frame $i \in \{1, \ldots, T_c\}$, the model predicts a depth map D_i and a set of Gaussian attributes G_i.

Simultaneously, for all timestamps $j \in \{1, \ldots, T_c+T_f\}$, the model estimates the camera parameters P_j and the time-conditioned velocity $V_{i,j}$ from source frame i to target frame j (where $j \neq i$).

Future Pose-free Extrapolation. To predict the scene state at target timestamp j, we warp the Gaussian centers from a source timestamp i using a time-conditioned velocity, rather than constant linear motion across different target frames:

$$\mu_{i \to j} = \mu_i + \mathbf{v}_{i,j} \cdot (j - i), \quad i \in \{1, \ldots, T_c\}, j \in \{1, \ldots, T_c + T_f\} \tag{1}$$

where $\mu \in \mathbb{R}^3$ denotes the 3D position, $j-i$ acts as a scaling factor to normalize the velocity learning across varying time gaps. By rendering the aggregated Gaussians via the estimated poses P_j from Sec. 3.3, Envision4D enables high-quality future view synthesis in a truly pose-free manner.

3.2 Motion Awareness via In-Layer Temporal Attention

Our model is built upon the VGGT backbone (Wang et al., 2025a), which processes a sequence of images $\mathbf{I} \in \mathbb{R}^{T_c \times 3 \times H \times W}$ into frame-wise tokens via DINOv2 (Oquab et al., 2023) and 24 layers of Alternating-Attention (AA). While concurrent works (He et al., 2026; Yang et al., 2026) typically append auxiliary motion modules on the top of the output frozen tokens, such post-refinement confines motion modeling to the added shallow blocks rather than intermediate backbone representations.

In contrast, we propose an *In-Layer Temporal Attention*. By embedding efficient temporal attention blocks directly into the intermediate stages of the frozen VGGT encoder, we empower the subsequent frozen AA layers to progressively propagate and reinforce the learned motion cues throughout the feature extraction process. Let $\mathbf{F}_l$ be the output of the l-th AA layer. Formally, we have:

$$\mathbf{F}_{l+1} = \mathrm{AA}_{l+1}(\mathrm{TAttn}(\mathbf{F}_l)), \quad l \in \{4, 11, 17, 23\} \tag{2}$$

where TAttn performs attention across the temporal dimension. The tokens from global attention and our temporal attention are then concatenated as motion tokens $\mathbf{F}^M$ for subsequent processing.

However, injecting newly initialized modules into the intermediate stages of a frozen network inherently alters the feature distribution for subsequent layers. This risks severe degradation of the original feature space, which may explain why existing approaches typically settle for post-layer refinements. To resolve this dilemma and stabilize the intermediate representations, we introduce a self-distillation supervision strategy (detailed in the training loss part of Sec. 3.4).

3.3 Future Extrapolation via Joint Pose-Motion Prediction

Future Pose Prediction. The first step for pose-free future extrapolation is to generate the subsequent camera poses. Building on both geometric priors from the observed context frames and target temporal embeddings, we iteratively refine future pose tokens while producing target-specific conditioning features for Gaussian motion prediction.

Given the aggregated camera pose tokens across all context frames, let $\mathbf{z}_{T_c} \in \mathbb{R}^C$ denote the camera token of the last observed frame. We initialize the predictions for the subsequent T_f unknown frames by adding a learnable offset $\boldsymbol{\delta} \in \mathbb{R}^C$ to $\mathbf{z}_{T_c}$:

$$\mathbf{z}_j = \mathbf{z}_{T_c} + \boldsymbol{\delta}, \quad j \in \{T_c + 1, \ldots, T_c + T_f\} \tag{3}$$

where $\boldsymbol{\delta}$ serves as an initialization seed for the extrapolated poses.

The known pose tokens, together with the initialized future tokens, form the full sequence representation $\mathbf{F}^{\text{cam}} \in \mathbb{R}^{(T_c+T_f) \times C}$. Additionally, to endow each token with positional awareness along the temporal axis, we compute a 1D sinusoidal time embedding for each frame index:

$$\mathbf{e}_j = \mathrm{Linear}(\mathrm{SinEmb}(j)) \in \mathbb{R}^C, \quad j \in \{1, \ldots, T_c + T_f\} \tag{4}$$

The sequence tokens are then concatenated with corresponding time embeddings and passed through a stack of self-attention blocks. This mechanism allows the model to iteratively refine the initial future predictions by globally conditioning on the observed history and their time embeddings:

$$\mathbf{Z}^{(l+1)} = \mathrm{SelfAttn}(\mathbf{Z}^{(l)}), \quad \text{where } \mathbf{Z}^{(0)} = [\mathbf{F}^{\text{cam}}, \mathbf{E}] \tag{5}$$

where $\mathbf{E} = [\mathbf{e}_1, \ldots, \mathbf{e}_{T_c+T_f}] \in \mathbb{R}^{(T_c+T_f) \times C}$ represents the time embedding matrix, and $\mathbf{Z}^{(l)}$ denotes the updated token sequence at the l-th attention layer. After iterative refinement, the camera tokens

are decoded into camera parameters P by the frozen camera head. The updated time embeddings $\widetilde{\mathbf{E}}$ also incorporate temporal and camera-motion cues through attention with camera tokens and serve as motion priors for subsequent velocity prediction.

Conditioned Motion Lifting. Unlike linear motion in STORM (Yang et al., 2024a), we model time-conditioned velocities by formulating motion prediction as a dynamic relational mapping between observations and other target states. This design enables the network to adaptively capture complex dynamics, mitigating motion errors caused by conventional constant-velocity extrapolation.

Specifically, given the learned motion tokens $\mathbf{F}^M \in \mathbb{R}^{T_c \times N \times C}$, let $f_{i,n} \in \mathbb{R}^C$ denote the token at context frame i and spatial location n. For each context frame i, we consider every target timestamp $j \neq i$ across all context and future frames, yielding $K = T_c + T_f - 1$ target timestamps. The corresponding motion prior $\widetilde{\mathbf{e}}_j$, the j-th row of $\widetilde{\mathbf{E}}$, encodes temporal and ego-motion cues for target timestamp j. We then modulate $f_{i,n}$ with this prior to obtain the target-specific motion feature $m_{i,j,n} \in \mathbb{R}^C$:

$$m_{i,j,n} = \widetilde{\mathbf{e}}_j \odot f_{i,n} \tag{6}$$

Applying this modulation across all valid source-target pairs lifts the context features into a motion volume $\mathbf{M} \in \mathbb{R}^{T_c \times K \times N \times C}$. After that, we flatten the temporal dimensions and pass it through a DPT head to obtain the final time-conditioned velocity predictions $V \in \mathbb{R}^{(T_c \times K) \times N \times 3}$.

3.4 Progressive Training Strategy

Future extrapolation inherently suffers from much more severe error accumulation than interpolation, with errors growing sharply as the number of extrapolated frames increases. This makes unsupervised velocity learning highly susceptible to model collapse. To stabilize the learning process, we introduce a progressive training strategy that regularizes the network across both the rendering mechanisms and the extrapolation optimization length.

Geometric Warm-up and Self-Exclusive Motion Learning. In the early training stages, we mainly focus on static geometry and pose optimization by enforcing only reconstruction on the context frames. Once a stable geometric foundation is established, we facilitate unsupervised motion learning using a self-exclusive rendering strategy. Specifically, during the rendering phase, we remove the target frame's own Gaussians. This strategy is crucial to prevent model degeneration, where the network might otherwise bypass actual motion learning by exploiting a trivial identity mapping of the target frame's appearance. By amplifying motion-induced misalignments, it forces the network to strictly distinguish between static and dynamic elements, ensuring robust velocity estimation.

Progressive Extrapolation Weighting. To further stabilize highly uncertain extrapolation process, we employ an exponentially decaying weighting scheme that down-weights future frames relative to context frames. For a frame at time index t, its unnormalized loss weight is computed as:

$$w_t = \exp(-\alpha \cdot \max(0, t - T_c)), \quad t \in \{1, \ldots, T_c + T_f\} \tag{7}$$

where the decay rate α linearly anneals from 1 to 0 over S_{max} steps. Then, w_t is normalized to maintain a mean of 1 across the sequence, ensuring stable gradient scale throughout annealing.

Training Loss. The primary training signal is driven by the image rendering loss, where we combine Mean Squared Error (MSE) and Learned Perceptual Image Patch Similarity (LPIPS) losses:

$$\mathcal{L}_{\text{rgb}} = \mathcal{L}_{\text{MSE}} + \lambda_{\text{lpips}} \mathcal{L}_{\text{LPIPS}} \tag{8}$$

Furthermore, we propose a self-distillation loss to regularize scene geometry. It functions as a soft regularizer that provides geometric supervision, and crucially, aligns the updated feature space with the pre-trained one to prevent feature degradation. Specifically, given all $T_c + T_f$ frames during training, a frozen VGGT predicts camera parameters P_{vggt} and depth maps D_{vggt} as pseudo-supervision, formulated as:

$$\mathcal{L}_{\text{cam}} = \|P - P_{\text{vggt}}\|_1, \qquad \mathcal{L}_{\text{depth}} = \|D - D_{\text{vggt}}\|_1, \tag{9}$$

where D comprises directly predicted context depth maps and future depth maps rendered from the extrapolated Gaussians.

Table 1: **Comparison to state-of-the-art methods on the Waymo dataset.** STORM and DGGT are retrained under our extrapolation protocol, using two context frames as in our method. Cosmos3 is included as a generative reference with one (†) or five (‡) context frames, both settings are evaluated at the same two future timestamps. Extrapolation runtimes are measured on a single A100 GPU.

Methods	Render Quality			Inference Speed	Capability	
	PSNR ↑	SSIM ↑	D-RMSE ↓	Time ↓	Pose-free	Unsup. Dynamic
Interpolation References						
PVG (Chen et al., 2026)	22.38	0.661	13.01	27 min	×	✓
DeformableGS (Yang et al., 2024c)	25.29	0.761	14.79	29 min	×	✓
DepthSplat (Xu et al., 2025a)	23.26	0.696	10.05	0.11 s	×	×
NoPoSplat (Ye et al., 2024)	24.31	0.751	9.08	23.22 s	✓	×
STORM (Yang et al., 2024a)	26.38	0.794	5.48	0.18 s	×	✓
DGGT (Chen et al., 2025)	27.41	0.846	3.47	0.39 s	✓	×
Extrapolation (Generative References)						
Cosmos3† (Agarwal et al., 2026)	23.62	0.680	–	1.40 s	✓	N/A
Cosmos3‡ (Agarwal et al., 2026)	27.04	0.774	–	1.59 s	✓	N/A
Extrapolation (Gaussian-based Methods)						
STORM (Yang et al., 2024a)	26.19	0.798	6.13	**0.12 s**	×	✓
DGGT (Chen et al., 2025)	24.38	0.756	7.67	0.56 s	✓	×
Ours	**27.81**	**0.816**	**3.98**	0.37 s	✓	✓



Figure 3: **Qualitative comparison on Waymo dataset.** Even without explicit motion guidance and known future poses, our model can well handle large movements under various conditions.

4 Experimental Results

4.1 Experimental Setup

Setup. We evaluate our method on the official validation splits of Waymo (Sun et al., 2020) and nuScenes (Caesar et al., 2020) datasets. For each validation clip, we condition on T_c frames to generate a full sequence of $T_c + T_f$ frames (where $T_c = T_f = 2$ by default), challenging the model to deduce future motions from sparse observations. Following prior feed-forward 4D reconstruction methods (Yang et al., 2024a; Chen et al., 2025), we sample all sequences at 10 Hz and evaluate extrapolation horizons from 2 frames (0.2s) to 6 frames (0.6s). The 0.6s setting also matches the longest forecasting horizon adopted by the generative VGGT-World (Sun et al., 2026). *Unless otherwise specified, all metrics are computed exclusively on the extrapolated future frames.*

Baselines. We compare against three feed-forward Gaussian Splatting methods for autonomous driving: STORM (Yang et al., 2024a), DGGT (Chen et al., 2025), and ReconDrive (Yu et al., 2026). STORM is our primary baseline, as it also learns scene motion without explicit motion supervision. We additionally include Cosmos3 (Agarwal et al., 2026) and VGGT-World (Sun et al., 2026) as generative reference baselines. Flux4D (Wang et al., 2025b) is excluded because it requires LiDAR

Table 2: **Comparison to state-of-the-art methods on the nuScenes dataset.** Cosmos3 is evaluated with five context frames.

Method	PSNR ↑	SSIM ↑	LPIPS ↓
Cosmos3‡ (Agarwal et al., 2026)	25.41	0.693	0.228
STORM (Yang et al., 2024a)	24.69	0.767	0.309
DGGT (Chen et al., 2025)	23.92	0.758	0.338
ReconDrive (Yu et al., 2026)	23.15	0.718	0.297
Ours	**26.86**	**0.815**	**0.164**

Table 3: **Camera pose estimation on Waymo and nuScenes.** Cosmos3 uses five context frames and is included for reference.

Method	Waymo AUC@30 ↑	nuScenes AUC@30 ↑
Cosmos3‡ (Agarwal et al., 2026)	80.10	80.14
VGGT (Wang et al., 2025a)	78.76	77.47
VGGT-World (Sun et al., 2026)	78.99	77.54
Ours	**80.33**	**78.26**

Table 4: **Quantitative comparison under varying context (T_c) and future (T_f) frames.** Notably, our method at $T_f = 6$ even achieves competitive performance compared to STORM at $T_f = 2$.

T_c	T_f	PSNR ↑		SSIM ↑		LPIPS ↓	
		STORM (Yang et al., 2024a)	Ours	STORM (Yang et al., 2024a)	Ours	STORM (Yang et al., 2024a)	Ours
2	2	26.19	**27.81**	0.798	**0.816**	0.242	**0.151**
2	4	25.55	**26.87**	0.773	**0.790**	0.263	**0.170**
2	6	24.41	**26.21**	0.753	**0.771**	0.346	**0.192**
3	6	24.62	**26.46**	0.752	**0.782**	0.302	**0.187**

inputs, unlike our unposed RGB-only setting. Except for Cosmos3, evaluated with its released checkpoint, all baselines are reproduced and retrained under our experimental setup.

Further details regarding the model architecture and implementation are provided in Appendix A.1.

4.2 Comparison Study

Rendering Quality. Tab. 1 compares methods on the Waymo dataset, with published interpolation results included for reference. For extrapolation, STORM and DGGT are retrained using the same two context frames as Envision4D. Cosmos3 is additionally evaluated as a generative reference with one or five context frames, with both configurations evaluated at the same two future timestamps. Envision4D achieves the best results across all three rendering metrics among Gaussian-based extrapolation methods, while also outperforming Cosmos3 in PSNR and SSIM even with its five-frame context. It is also the only compared Gaussian-based method that jointly supports future pose-free reconstruction and unsupervised motion learning. These gains are also evident in the qualitative comparisons in Fig. 3. DGGT exhibits object misalignment and distorted scene structures, while STORM produces ghosting and blurring around moving vehicles. In contrast, Envision4D better preserves object positions and scene details, yielding more accurate future reconstructions.

We also evaluate all methods on nuScenes in Tab. 2, where Envision4D again achieves superior rendering quality across datasets. More visualization can be seen in Appendix A.3.

Camera Pose Estimation. We evaluate future pose accuracy to assess camera extrapolation. Following VGGT, we compute AUC@30 from pairwise relative rotation and translation angular errors, but excluding pairs where both frames belong to the observed context.

As shown in Tab. 3, our method consistently outperforms VGGT and VGGT-World on both datasets. Notably, it surpasses VGGT using only two context frames, while VGGT receives the full sequence, including future frames. The improvement over VGGT-World under the same future-prediction setting indicates that additional image-space reconstruction constraints from explicit 4DGS rendering benefit joint geometry and pose estimation. This also suggests that our framework can refine VGGT-derived pose estimates without requiring camera calibration.

Context and Extrapolation Lengths. We evaluate performance under different context and extrapolation lengths in Tab. 4. Our method maintains stable rendering quality as the extrapolation horizon increases. Notably, at $T_f = 6$, it still achieves better PSNR and LPIPS than STORM with only $T_f = 2$, indicating stronger robustness to temporal error accumulation. Additionally, extending context frames further enhances extrapolation capability by providing richer dynamic cues.

We present qualitative results for 6-frame extrapolation. As shown in Fig. 4, errors accumulate with the extrapolation horizon. STORM exhibits unstable object motion and severe ghosting at distant frames, largely due to its linear velocity modeling. Despite using a pre-trained tracker, DGGT shows



Figure 4: **Qualitative comparisons of 6 frame extrapolation.** The blue and orange dots denote reconstructed context frames and extrapolated future frames, respectively. Our model achieves significantly more stable extrapolation, even without extra motion guidance and GT future poses.

Table 5: **Ablation on proposed modules.** We use linear bidirectional motion when without *CML*.

CML	*In-layer TAttn*	*Post-layer TAttn*	PSNR↑	SSIM↑	LPIPS↓
			25.12	0.760	0.207
✓			25.88	0.772	0.182
✓		✓	26.98	0.788	0.170
✓	✓		**27.81**	**0.816**	**0.151**

Table 6: **Ablation on Future Pose (*FP*) Prediction module.** *Linear* replaces *FP* with linear pose extrapolation.

Strategy	AUC@30 ↑	PSNR ↑	SSIM ↑	LPIPS↓
Linear	75.55	26.74	0.798	0.186
FP (Ours)	80.33	27.81	0.816	0.151

growing trajectory errors on the moving vehicle. Its linear extrapolation of camera poses introduces ego-motion drift, reflected by static-region inconsistencies such as tree artifacts in the upper-right corner. In contrast, Envision4D maintains stable object motion and camera trajectories, preserving high-fidelity reconstruction over the extended prediction horizon.

4.3 Ablation Study

We ablate motion modeling and temporal attention in Tab. 5, with *Future Pose (FP) Prediction* enabled throughout. The first two rows compare a bidirectional constant-velocity baseline following STORM (Yang et al., 2024a) with *Conditioned Motion Lifting (CML)*. Replacing velocities shared across target timestamps with time-conditioned motion predictions increases PSNR from 25.12 to 25.88, validating the importance of modeling non-linear motion for future extrapolation. The last two rows compare our *In-layer TAttn*, which inserts temporal attention within the frozen backbone, with a *Post-layer TAttn* variant that appends it after the backbone. Both variants use the same number of temporal attention blocks and identical training settings. Our design increases PSNR from 26.98 to 27.81, with corresponding improvements in SSIM and LPIPS, showing that temporal aggregation within the backbone is more effective than aggregation after the backbone. Tab. 6 further compares *FP* against linear pose extrapolation. *FP* improves both future pose accuracy and rendering quality, demonstrating that learned ego-motion prediction benefits subsequent scene reconstruction.

We then ablate the *Progressive Training Strategy* in Tab. 7. Geometric warm-up improves rendering quality by establishing reliable geometry and texture before future extrapolation, while progressive extrapolation loss weighting further stabilizes unsupervised motion learning. Their combination achieves the best performance, demonstrating the complementary benefits of the two strategies.

4.4 Dynamic Scene Evaluation and Generalization

Dynamic Scene Evaluation. We evaluate dynamic-region rendering on future frames and scene flow from each context frame to the next timestamp ($1 \to 2$ and $2 \to 3$, with frames 1 and 2 as context). Scene flow is evaluated against the ground-truth 3D scene flow provided by the Waymo Open Dataset. As shown in Tab. 8, Envision4D outperforms STORM in both evaluations, benefiting from its time-conditioned motion modeling and enhanced temporal motion awareness. The qualitative

Table 7: **Ablation study on *Progressive Training Strategy*.** Both strategies bring consistent gains on novel view synthesis.

Warm-up	*Prog. Weighting*	PSNR ↑	SSIM ↑	LPIPS ↓
		27.05	0.796	0.172
✓		27.36	0.805	0.160
	✓	27.49	0.812	0.156
✓	✓	**27.81**	**0.816**	**0.151**

Table 8: **Quantitative comparison of dynamic-region rendering and flow estimation on Waymo.** Our method consistently outperforms STORM across all motion-related metrics.

Method	Dyn. PSNR ↑	EPE3D (m) ↓	Acc_5 (%) ↑	Acc_{10} (%) ↑
STORM	21.23	0.206	83.12	85.77
Ours	**24.46**	**0.161**	**85.13**	**91.45**



Figure 5: **Visualization of dynamic masks and predicted velocities.** Left: dynamic mask. Right: scene flow. We overlay the original image on each example to enhance clarity.

results in Fig. 5 further show that our model accurately separates dynamic elements and recovers reliable 3D motion. Notably, without explicit motion guidance, Envision4D can distinguish a moving vehicle from a nearby stationary one, producing accurate dynamic masks and motion flows.

In-the-wild Scene Reconstruction. To demonstrate the generalization capability, we extend evaluations to challenging in-the-wild scenarios with entirely unknown camera parameters. As shown in Fig. 6, despite the high uncertainty of unconstrained camera and object movements, Envision4D achieves robust future forecasting with superior rendering fidelity and geometric alignment. More visualization can be seen in Appendix A.3.

5 Limitations and Conclusion

Limitations. While our work exhibits stronger self-supervised motion learning quality and relieves the model from inflexible extrapolation constraints, it presents certain limitations. First, motion estimation for faraway, rapidly approaching objects is constrained by the extreme sparsity of input visual cues. Second, inherent to our reconstruction-based nature, the model struggles to hallucinate entirely unseen regions, meaning it cannot forecast as far into the future as generative models. In future work, we aim to incorporate generative priors to enable longer-horizon extrapolation.

Conclusion. In this paper, we introduce Envision4D, a novel 4DGS model for future scene extrapolation without explicit motion guidance or predefined future cameras. We propose *Joint Pose-Motion Prediction* to infer all target camera poses and corresponding non-linear Gaussian velocities, utilizing *In-layer Temporal Attention* to strengthen temporal perception and facilitate motion learning. Additionally, a *Progressive Training Strategy* stabilizes the unsupervised learning process and mitigates error accumulation. Extensive experiments demonstrate state-of-the-art performance in dynamic scene reconstruction and promising generalization to in-the-wild driving videos.



Figure 6: **Qualitative results on in-the-wild data.** Envision4D is capable to generate reliable future extrapolation directly from uncalibrated open-world context images.

AI USE STATEMENT

We used generative AI tools to improve the grammar, clarity, and readability of the manuscript. All AI-assisted revisions were reviewed by the authors to ensure accuracy and preserve the intended scientific meaning. We take responsibility for the final content of this work, including text, claims or artifacts produced with the aid of generative AI.

REFERENCES

Niket Agarwal, Arslan Ali, Jon Allen, Martin Antolini, Adeline Aubame, Alisson Azzolini, Junjie Bai, Maciej Bala, Yogesh Balaji, Josh Bapst, et al. Cosmos 3: Omnimodal world models for physical ai. *arXiv preprint arXiv:2606.02800*, 2026.

Arslan Ali, Junjie Bai, Maciej Bala, Yogesh Balaji, Aaron Blakeman, Tiffany Cai, Jiaxin Cao, Tianshi Cao, Elizabeth Cha, Yu-Wei Chao, et al. World simulation with video foundation models for physical ai. *arXiv preprint arXiv:2511.00062*, 2025.

Arnold Caleb Asiimwe and Carl Vondrick. 4d gaussian splatting as a learned dynamical system. *arXiv preprint arXiv:2512.19648*, 2025.

Federico Baldassarre, Marc Szafraniec, Basile Terver, Vasil Khalidov, Francisco Massa, Yann LeCun, Patrick Labatut, Maximilian Seitzer, and Piotr Bojanowski. Back to the features: Dino as a foundation for video world models. *arXiv preprint arXiv:2507.19468*, 2025.

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorenz, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023.

Holger Caesar, Varun Bankiti, Alex H Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11621–11631, 2020.

Xiaoxue Chen, Ziyi Xiong, Yuantao Chen, Gen Li, Nan Wang, Hongcheng Luo, Long Chen, Haiyang Sun, Bing Wang, Guang Chen, et al. Dggt: Feedforward 4d reconstruction of dynamic driving scenes using unposed images. *arXiv preprint arXiv:2512.03004*, 2025.

Yuedong Chen, Haofei Xu, Chuanxia Zheng, Bohan Zhuang, Marc Pollefeys, Andreas Geiger, Tat-Jen Cham, and Jianfei Cai. Mvsplat: Efficient 3d gaussian splatting from sparse multi-view images. In *European conference on computer vision*, pp. 370–386. Springer, 2024a.

Yurui Chen, Chun Gu, Junzhe Jiang, Xiatian Zhu, and Li Zhang. Periodic vibration gaussian: Dynamic urban scene reconstruction and real-time rendering. *International Journal of Computer Vision*, 134(3):83, 2026.

Ziyu Chen, Jiawei Yang, Jiahui Huang, Riccardo De Lutio, Janick Martinez Esturo, Boris Ivanovic, Or Litany, Zan Gojcic, Sanja Fidler, Marco Pavone, et al. Omnire: Omni urban scene reconstruction. *arXiv preprint arXiv:2408.16760*, 2024b.

Yixiang Dai, Fan Jiang, Chiyu Wang, Mu Xu, and Yonggang Qi. Fantasyworld: Geometry-consistent world modeling via unified video and 3d prediction. *arXiv preprint arXiv:2509.21657*, 2025.

Xin Fei, Wenzhao Zheng, Yueqi Duan, Wei Zhan, Masayoshi Tomizuka, Kurt Keutzer, and Jiwen Lu. Driv3r: Learning dense 4d reconstruction for autonomous driving. *arXiv preprint arXiv:2412.06777*, 2024.

Haoyu Fu, Diankun Zhang, Zongchuang Zhao, Jianfeng Cui, Dingkang Liang, Chong Zhang, Dingyuan Zhang, Hongwei Xie, Bing Wang, and Xiang Bai. Orion: A holistic end-to-end autonomous driving framework by vision-language instructed action generation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 24823–24834, 2025.

Shenyuan Gao, Jiazhi Yang, Li Chen, Kashyap Chitta, Yihang Qiu, Andreas Geiger, Jun Zhang, and Hongyang Li. Vista: A generalizable driving world model with high fidelity and versatile controllability. *Advances in Neural Information Processing Systems*, 37:91560–91596, 2024.

Zhuolin He, Jing Li, Guanghao Li, Xiaolei Chen, Jiacheng Tang, Siyang Zhang, Zhounan Jin, Feipeng Cai, Bin Li, Jian Pu, et al. Dynamicvggt: Learning dynamic point maps for 4d scene reconstruction in autonomous driving. *arXiv preprint arXiv:2603.08254*, 2026.

Jonathan Ho, Tim Salimans, Alexey Gritsenko, William Chan, Mohammad Norouzi, and David J Fleet. Video diffusion models. *Advances in neural information processing systems*, 35:8633–8646, 2022.

Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *arXiv preprint arXiv:2205.15868*, 2022.

Yu Hu, Chong Cheng, Sicheng Yu, Xiaoyang Guo, and Hao Wang. Vggt4d: Mining motion cues in visual geometry transformers for 4d scene reconstruction. *arXiv preprint arXiv:2511.19971*, 2025.

Jiaxin Huang, Yuanbo Yang, Bangbang Yang, Lin Ma, Yuewen Ma, and Yiyi Liao. Gen3r: 3d scene generation meets feed-forward reconstruction. *arXiv preprint arXiv:2601.04090*, 2026.

Nan Huang, Xiaobao Wei, Wenzhao Zheng, Pengju An, Ming Lu, Wei Zhan, Masayoshi Tomizuka, Kurt Keutzer, and Shanghang Zhang. S3gaussian: Self-supervised street gaussians for autonomous driving. *arXiv preprint arXiv:2405.20323*, 2024.

Junhwa Hur, Charles Herrmann, Songyou Peng, Philipp Henzler, Zeyu Ma, Todd Zickler, and Deqing Sun. Ufo-4d: Unposed feedforward 4d reconstruction from two images. *arXiv preprint arXiv:2602.24290*, 2026.

Lihan Jiang, Yucheng Mao, Linning Xu, Tao Lu, Kerui Ren, Yichen Jin, Xudong Xu, Mulin Yu, Jiangmiao Pang, Feng Zhao, et al. Anysplat: Feed-forward 3d gaussian splatting from unconstrained views. *ACM Transactions on Graphics (TOG)*, 44(6):1–16, 2025.

Efstathios Karypidis, Ioannis Kakogeorgiou, Spyridon Gidaris, and Nikos Komodakis. Dinoforesight: Looking into the future with dino. *Advances in Neural Information Processing Systems*, 38:163779–163811, 2026.

Deqi Li, Shi-Sheng Huang, Zhiyuan Lu, Xinran Duan, and Hua Huang. St-4dgs: Spatial-temporally consistent 4d gaussian splatting for efficient dynamic scene rendering. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–11, 2024a.

Hao Li, Yuanyuan Gao, Chenming Wu, Dingwen Zhang, Yalun Dai, Chen Zhao, Haocheng Feng, Errui Ding, Jingdong Wang, and Junwei Han. Ggrt: Towards pose-free generalizable 3d gaussian splatting in real-time. In *European Conference on Computer Vision*, pp. 325–341. Springer, 2024b.

Hanxue Liang, Jiawei Ren, Ashkan Mirzaei, Antonio Torralba, Ziwei Liu, Igor Gilitschenski, Sanja Fidler, Cengiz Oztireli, Huan Ling, Zan Gojcic, et al. Feed-forward bullet-time reconstruction of dynamic scenes from monocular videos. *arXiv preprint arXiv:2412.03526*, 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. *arXiv preprint arXiv:2507.10065*, 2025.

Yixin Liu, Kai Zhang, Yuan Li, Zhiling Yan, Chujie Gao, Ruoxi Chen, Zhengqing Yuan, Yue Huang, Hanchi Sun, Jianfeng Gao, et al. Sora: A review on background, technology, limitations, and opportunities of large vision models. *arXiv preprint arXiv:2402.17177*, 2024.

Sheng Miao, Sijin Li, Pan Wang, Dongfeng Bai, Bingbing Liu, Yue Wang, Andreas Geiger, and Yiyi Liao. Evolsplat4d: Efficient volume-based gaussian splatting for 4d urban scene synthesis. *arXiv preprint arXiv:2601.15951*, 2026.

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.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. Sam 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

Qi Song, Qingyong Hu, Chi Zhang, Yongquan Chen, and Rui Huang. Divide and conquer: Improving multi-camera 3d perception with 2d semantic-depth priors and input-dependent queries. *IEEE Transactions on Image Processing*, 33:897–909, 2024.

Qi Song, Chenghong Li, Haotong Lin, Sida Peng, and Rui Huang. Adgaussian: Generalizable gaussian splatting for autonomous driving with multi-modal inputs. *arXiv preprint arXiv:2504.00437*, 2025.

Edgar Sucar, Eldar Insafutdinov, Zihang Lai, and Andrea Vedaldi. V-dpm: 4d video reconstruction with dynamic point maps. *arXiv preprint arXiv:2601.09499*, 2026.

Pei Sun, Henrik Kretzschmar, Xerxes Dotiwalla, Aurelien Chouard, Vijaysai Patnaik, Paul Tsui, James Guo, Yin Zhou, Yuning Chai, Benjamin Caine, et al. Scalability in perception for autonomous driving: Waymo open dataset. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2446–2454, 2020.

Xiangyu Sun, Shijie Wang, Fengyi Zhang, Lin Liu, Caiyan Jia, Ziying Song, Zi Huang, and Yadan Luo. Vggt-world: Transforming vggt into an autoregressive geometry world model. *arXiv preprint arXiv:2603.12655*, 2026.

Stanislaw Szymanowicz, Chrisitian Rupprecht, and Andrea Vedaldi. Splatter image: Ultra-fast single-view 3d reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10208–10217, 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 *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5294–5306, 2025a.

Jingkang Wang, Henry Che, Yun Chen, Ze Yang, Lily Goli, Sivabalan Manivasagam, and Raquel Urtasun. Flux4d: Flow-based unsupervised 4d reconstruction. *arXiv preprint arXiv:2512.03210*, 2025b.

Qianqian Wang, Vickie Ye, Hang Gao, Weijia Zeng, Jake Austin, Zhengqi Li, and Angjoo Kanazawa. Shape of motion: 4d reconstruction from a single video. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9660–9672, 2025c.

Shihao Wang, Zhiding Yu, Xiaohui Jiang, Shiyi Lan, Min Shi, Nadine Chang, Jan Kautz, Ying Li, and Jose M Alvarez. Omnidrive: A holistic llm-agent framework for autonomous driving with 3d perception, reasoning and planning. *arXiv preprint arXiv:2405.01533*, 1(2):3, 2024.

Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4d gaussian splatting for real-time dynamic scene rendering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20310–20320, 2024.

Haoyu Wu, Diankun Wu, Tianyu He, Junliang Guo, Yang Ye, Yueqi Duan, and Jiang Bian. Geometry forcing: Marrying video diffusion and 3d representation for consistent world modeling. *arXiv preprint arXiv:2507.07982*, 2025a.

Tong Wu, Shuai Yang, Ryan Po, Yinghao Xu, Ziwei Liu, Dahua Lin, and Gordon Wetzstein. Video world models with long-term spatial memory. *arXiv preprint arXiv:2506.05284*, 2025b.

Zike Wu, Qi Yan, Xuanyu Yi, Lele Wang, and Renjie Liao. Streamsplat: Towards online dynamic 3d reconstruction from uncalibrated video streams. *arXiv preprint arXiv:2506.08862*, 2025c.

Haofei Xu, Songyou Peng, Fangjinhua Wang, Hermann Blum, Daniel Barath, Andreas Geiger, and Marc Pollefeys. Depthsplat: Connecting gaussian splatting and depth. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 16453–16463, 2025a.

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*, 2025b.

Yunzhi Yan, Zhen Xu, Haotong Lin, Haian Jin, Haoyu Guo, Yida Wang, Kun Zhan, Xianpeng Lang, Hujun Bao, Xiaowei Zhou, et al. Streetcrafter: Street view synthesis with controllable video diffusion models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 822–832, 2025.

Jiawei Yang, Jiahui Huang, Yuxiao Chen, Yan Wang, Boyi Li, Yurong You, Apoorva Sharma, Maximilian Igl, Peter Karkus, Danfei Xu, et al. Storm: Spatio-temporal reconstruction model for large-scale outdoor scenes. *arXiv preprint arXiv:2501.00602*, 2024a.

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.

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. *arXiv preprint arXiv:2408.06072*, 2024b.

Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3d gaussians for high-fidelity monocular dynamic scene reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 20331–20341, 2024c.

Botao Ye, Sifei Liu, Haofei Xu, Xueting Li, Marc Pollefeys, Ming-Hsuan Yang, and Songyou Peng. No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images. *arXiv preprint arXiv:2410.24207*, 2024.

Vickie Ye, Ruilong Li, Justin Kerr, Matias Turkulainen, Brent Yi, Zhuoyang Pan, Otto Seiskari, Jianbo Ye, Jeffrey Hu, Matthew Tancik, and Angjoo Kanazawa. gsplat: An open-source library for gaussian splatting. *Journal of Machine Learning Research*, 2025.

Haibao Yu, Kuntao Xiao, Jiahang Wang, Ruiyang Hao, Yuxin Huang, Guoran Hu, Haifang Qin, Bowen Jing, Yuntian Bo, and Ping Luo. Recondrive: Fast feed-forward 4d gaussian splatting for autonomous driving scene reconstruction. *arXiv preprint arXiv:2603.07552*, 2026.

Zehao Yu, Anpei Chen, Binbin Huang, Torsten Sattler, and Andreas Geiger. Mip-splatting: Alias-free 3d gaussian splatting. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19447–19456, 2024.

Kaiwen Zhang, Zhenyu Tang, Xiaotao Hu, Xingang Pan, Xiaoyang Guo, Yuan Liu, Jingwei Huang, Li Yuan, Qian Zhang, Xiao-Xiao Long, et al. Epona: Autoregressive diffusion world model for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27220–27230, 2025.

Xingcheng Zhou, Xuyuan Han, Feng Yang, Yunpu Ma, Volker Tresp, and Alois Knoll. Opendrivevla: Towards end-to-end autonomous driving with large vision language action model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 13782–13790, 2026.

A Appendix

A.1 Implementation Details

Datasets and Evaluation Protocol. The Waymo Open Dataset (Sun et al., 2020) contains 798 training and 202 validation sequences, while the nuScenes dataset (Caesar et al., 2020) comprises 700 training and 150 validation scenes. For evaluation, we sample 10 non-overlapping video clips from each validation sequence or scene, yielding 2,020 clips for Waymo and 1,500 for nuScenes.

Model Architecture. The *In-layer Temporal Attention* and the self-attention module in *Future Pose Prediction* both follow the VGGT attention-block architecture. Each temporal attention module consists of a single block, while the pose self-attention stacks 2 blocks and is iterated for 4 times.

Training and Optimization. We resize all input frames to 350×518 and use GSplat (Ye et al., 2025) for Gaussian Splatting rendering. We train the model for 100K iterations with a batch size of 2, using the Adam optimizer and a cosine learning rate scheduler with an initial learning rate of 1×10^{-4}. The loss weights are set to $\lambda_{\text{LPIPS}} = 0.05$ and $\lambda_{\text{cam}} = 5.0$, with all other loss weights set to 1. For the *Progressive Training Strategy*, the geometric warm-up stage spans 2.5K iterations, while the progressive extrapolation weighting stage spans 50K iterations. All experiments are conducted on a single A100 GPU.

A.2 Additional Results

Cross-frame Scene Reconstruction. Different from previous works that evaluate reconstruction performance using the input frames themselves, which may easily lead to overfitting static geometry rather than assessing motion, we introduce a more strict cross-frame rendering evaluation to thoroughly examine the model's capability in estimating velocities between context frames. To ensure fairness, the baseline model is also enhanced with our self-exclusive motion learning. Meanwhile, since the backward rendering inevitably introduces invisible regions near the image boundaries, we apply a boundary cropping during evaluation. As shown in Tab. A.1, our model significantly outperforms STORM in both rendering directions. These results indicate that the benefits of our motion modeling extend beyond future extrapolation to reconstruction between adjacent observed frames.

Table A.1: **Cross-frame scene reconstruction performance.** The notations $2 \rightarrow 1$ and $1 \rightarrow 2$ denote the backward (warping frame 2 to render frame 1) and forward (warping frame 1 to render frame 2) synthesis, respectively.

Method	2→1			1→2		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
STORM (Yang et al., 2024a)	27.62	0.836	0.237	27.97	0.838	0.218
Ours	30.36	0.886	0.122	30.71	0.891	0.113

A.3 Visualization Results

We provide additional qualitative results on nuScenes, 6-frame extrapolation, and in-the-wild videos. Fig. A.1 shows predictions on nuScenes under diverse lighting and weather conditions, including nighttime and rainy scenes. Our method preserves the overall scene structure and motion consistency across diverse scenarios. Fig. A.2 compares 6-frame extrapolation with STORM and DGGT using two context frames. As the prediction horizon increases, STORM exhibits pronounced ghosting and blurred boundaries around moving vehicles, while DGGT produces visible distortions and trajectory deviations. In particular, the vehicle highlighted in the context frames increasingly deviates from its ground-truth position in the baseline predictions. Our method better maintains its appearance and motion trajectory across future frames. Fig. A.3 presents additional in-the-wild results involving a vehicle crossing an intersection. The predictions capture the vehicle's displacement while preserving the surrounding road layout and background structure, further supporting the generalization of our method to unseen real-world videos.



Figure A.1: **Qualitative results on the nuScenes dataset.** We present qualitative examples under daytime, rainy, and nighttime conditions. Each example shows consecutive timestamps from left to right, illustrating reconstruction quality under varying illumination and weather conditions.

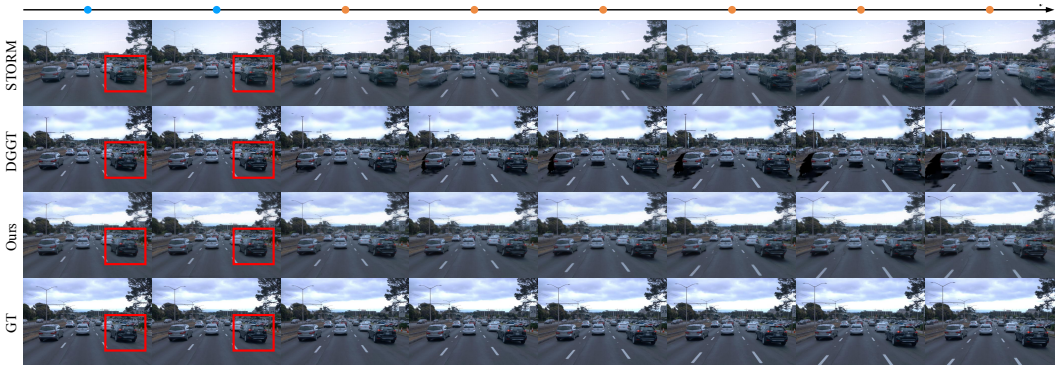


Figure A.2: **Additional comparisons of 6 frame extrapolation.** The first two columns correspond to context timestamps (blue), followed by six future timestamps (orange). Red boxes highlight a moving vehicle in the context frames. Compared with STORM and DGGT, our method better preserves vehicle structure and reduces motion-related artifacts over the extended prediction horizon.



Figure A.3: **Additional in-the-wild results.** Our method accurately predicts both ego-motion and object motion across diverse driving environments.