SeeU: Seeing the Unseen World via 4D Dynamics-aware Generation Supplementary Materials

Yu Yuan[1], Tharindu Wickremasinghe[1], Zeeshan Nadir[2], Xijun Wang[1], Yiheng Chi[1], Stanley H. Chan[1]
[1]Purdue University [2]Samsung Research America

A. Introduction

This supplementary material provides additional discussions and details on the SeeU45 data (Section B), Continuous 4D Dynamics Model (C4DD) design and ablation study (Section C), spatial-temporal in-context generation (Section D), the 3D geometric metrics used in the experiments (Section E), the limitations (Section F), robustness and scalability analysis (Section G), more comparison results (Section H), and more visual results (Section I).

To more clearly demonstrate SeeU's temporal and spatial generation abilities, we recommend that readers refer to the **videos** included in the supplementary materials.

B. More Details of SeeU45 Data

The SeeU45 dataset consists of 45 dynamic scenes, including 10 scenes that we manually captured and 35 scenes collected from public video datasets [1–3, 7, 8]. Each scene is provided in two forms: a ground-truth (GT) sequence and a training subset. The GT split contains the full dynamic sequence for each scene, while the training split is constructed by taking either the middle segment of the GT sequence or a temporally sampled version of that middle segment.

SeeU45 covers a diverse set of conditions in terms of scene type, foreground subjects, camera regimes, and motion types. A summary of the dataset statistics is provided in Table 4.

Composition	
Scenes (total)	45
Our captured	10
From public datasets	35
Scene Types	
Indoor scenes	6
Outdoor scenes	39
Foreground Subjects (some scenes contain both)	
Humans	8
Animals	13
Robots	3
Vehicles	18
Everyday objects	12
Camera Regimes	
Static	10
Handheld	28
Drone	7
Motion Types	
Rigid motion	21
Non-rigid motion	24
Frame Statistics	
GT frames / scene (min / max / avg)	9 / 521 / 80.24
Train frames / scene (min / max / avg)	7 / 47 / 15.96

Table 4. Statistics of the SeeU45 dataset.

C. More Details of Continuous 4D Dynamics Model

C.1. Architecture.

In Algorithm 1 we present the detailed architecture of the Continuous 4D Dynamics Model (C4DD). During training, C4DD learns continuous and smooth motion/camera bases by optimizing B-spline control points, and enforces physically consistent and smooth extrapolation through physics-aware constraints.

C.2. More Details About Ablation Study on C4DD Architecture.

To assess the necessity of the proposed C4DD architecture, we replace the B-spline–based design with a pure MLP layer and train it thoroughly. As shown in Fig. 7, although the MLP-based variant can roughly fit the overall trend of the motion bases, the resulting continuous trajectories are highly noisy and lack smoothness. This significantly degrades the spatio-temporal quality of the model's predictions (see Fig. 8).

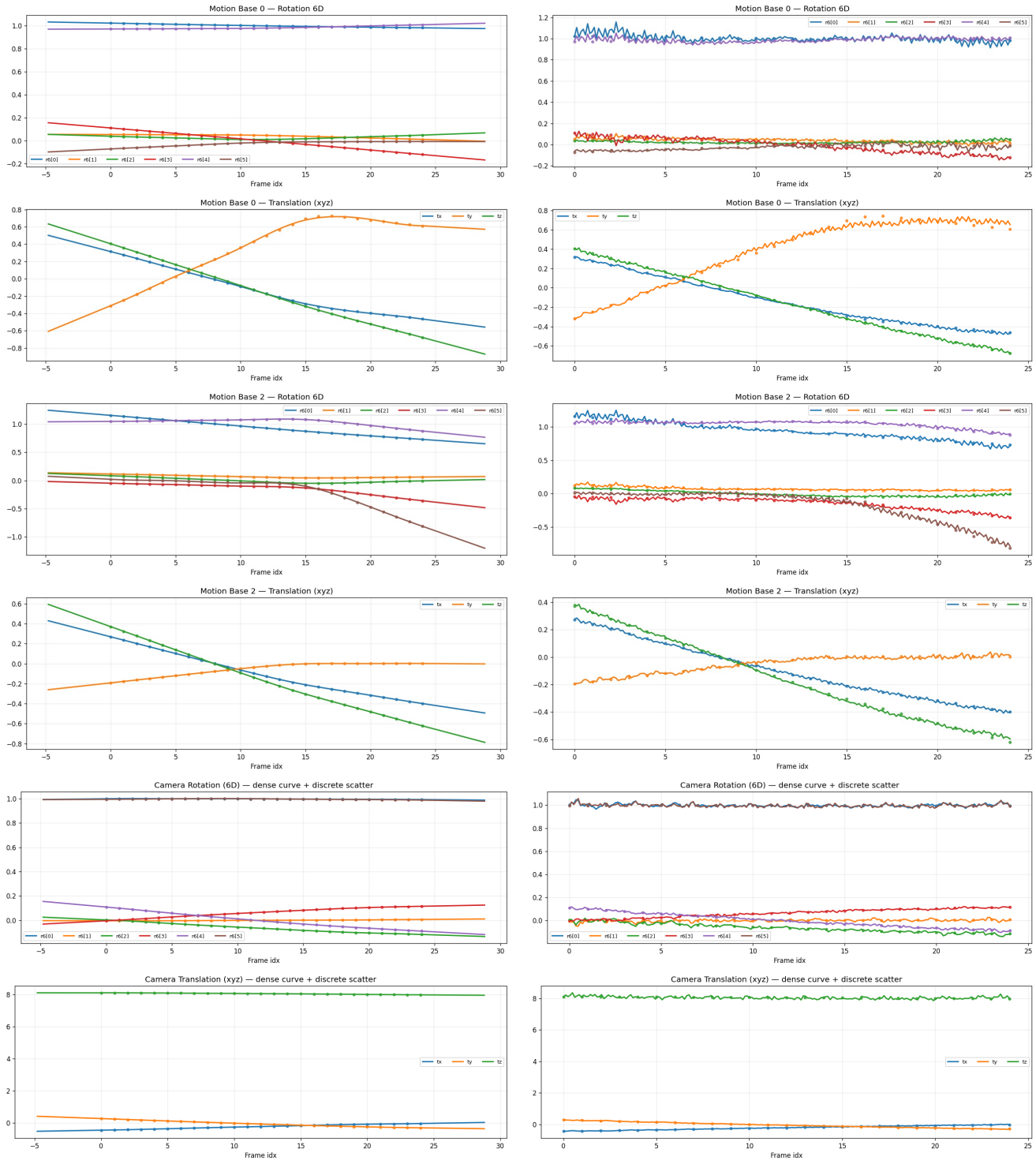


Figure 7. The C4DD with MLP variant (b) predicts motion bases with reduced temporal smoothness and physical consistency, confirming the advantageous inductive bias of spline priors (a) for continuous dynamics.

Algorithm 1 Continuous 4D Dynamics Model Architecture

Require: Number of motion bases K, number of control points M, degree p.

Initialization:
Build an open-uniform knot vector $\{u_i\}_{i=0}^{M+p}$ on $[0,1]$.
Initialize motion control points $\mathbf{C}^{\text{mot}} \in \mathbb{R}^{K\times 9\times M}$.
Initialize camera control points $\mathbf{C}^{\text{cam}} \in \mathbb{R}^{1\times 9\times M}$.

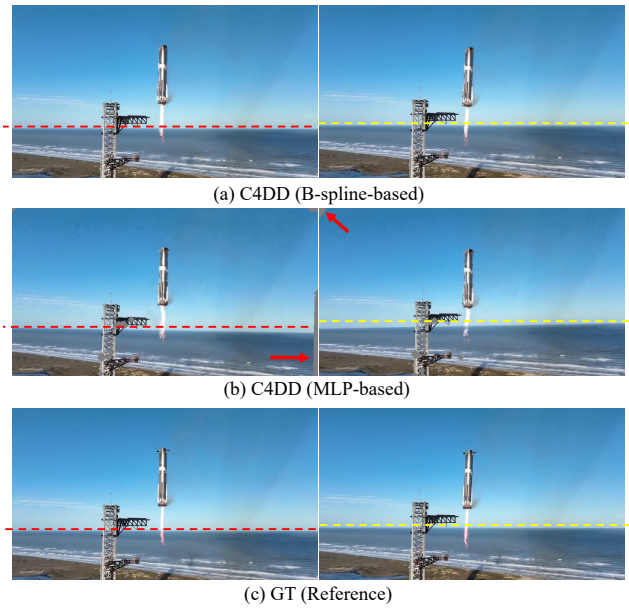
function BSPLINEBASIS($\mathbf{t}_{01}$) ▷ $\mathbf{t}_{01} \in (0,1)^{B\times 1}$
    Compute B-spline basis $\mathbf{B} \in \mathbb{R}^{B\times M}$ using Cox–de Boor recursion on knots $\{u_i\}$.
    return $\mathbf{B}$
end function

function FORWARD($\mathbf{t}_{\text{norm}}$) ▷ $\mathbf{t}_{\text{norm}} \in [-1,1]^{B\times 1}$
    Map time to $(0,1)$: $\mathbf{t}_{01} \leftarrow \text{clip}(0.5\,\mathbf{t}_{\text{norm}} + 0.5,\, 10^{-6},\, 1-10^{-6})$.
    $\mathbf{B} \leftarrow$ BSPLINEBASIS($\mathbf{t}_{01}$) ▷ $\mathbf{B} \in \mathbb{R}^{B\times M}$
    Motion bases: $\mathbf{Y}^{\text{mot}} \leftarrow \mathbf{C}^{\text{mot}}\mathbf{B}^{\top} \in \mathbb{R}^{K\times 9\times B}$.
    Camera pose: $\mathbf{Y}^{\text{cam}} \leftarrow \mathbf{C}^{\text{cam}}\mathbf{B}^{\top} \in \mathbb{R}^{1\times 9\times B}$.
    Reshape to $\mathbf{Y}^{\text{mot}} \in \mathbb{R}^{K\times B\times 9}$, $\mathbf{Y}^{\text{cam}} \in \mathbb{R}^{B\times 9}$.
    return $\mathbf{Y}^{\text{mot}}$, $\mathbf{Y}^{\text{cam}}$
end function

function FORWARDEXTRAP($\mathbf{t}_{\text{norm}}$) ▷ linear extrapolation outside $[-1,1]$
    $\mathbf{t}_{\text{clamp}} \leftarrow \text{clip}(\mathbf{t}_{\text{norm}}, -1, 1)$
    $\mathbf{Y}_0^{\text{mot}}, \mathbf{Y}_0^{\text{cam}} \leftarrow$ FORWARD($\mathbf{t}_{\text{clamp}}$)
    Estimate endpoint slopes $\mathbf{s}_L, \mathbf{s}_R$ at $t=-1$ and $t=1$ using finite differences (step size $\Delta t \approx 2/(T-1)$).
    Initialize $\mathbf{Y}^{\text{mot}} \leftarrow \mathbf{Y}_0^{\text{mot}}$, $\mathbf{Y}^{\text{cam}} \leftarrow \mathbf{Y}_0^{\text{cam}}$.
    for each time index b **do**
        if $t_{\text{norm}}[b] < -1$ **then**
            $\Delta t \leftarrow t_{\text{norm}}[b] + 1$
            Extrapolate $\mathbf{Y}^{\text{mot}}[:,b,:]$ and $\mathbf{Y}^{\text{cam}}[b,:]$ using left endpoint and slope $\mathbf{s}_L$.
        else if $t_{\text{norm}}[b] > 1$ **then**
            $\Delta t \leftarrow t_{\text{norm}}[b] - 1$
            Extrapolate $\mathbf{Y}^{\text{mot}}[:,b,:]$ and $\mathbf{Y}^{\text{cam}}[b,:]$ using right endpoint and slope $\mathbf{s}_R$.
        end if
    end for
    return $\mathbf{Y}^{\text{mot}}$, $\mathbf{Y}^{\text{cam}}$
end function

D. More Details of Spatial-Temporal In-Context Generation

The third stage, context-aware video filling, is mainly adapted from VACE [6]. During inference, We provide three types of contextual information:



(a) C4DD (B-spline-based)

(b) C4DD (MLP-based)

(c) GT (Reference)

Figure 8. Visual comparison on C4DD Architectures. The C4DD with spline constrains (a) has better smoothness and physical consistency (both camera pose and foreground dynamics).

1. **Text prompt.** The text prompt describes the global spatio-temporal semantics, adds extra guidance for the regions to be filled, and emphasizes physical consistency in the scene. For example: *"A camel is walking slowly in his enclosure. The enclosure has sand on the floor, surrounded by a wooden fence and planks. The background has trees. Restore the masked regions of the video with the background of the enclosure. Make the colors and background behind the camel realistic and continuous."*
2. **Projected frames.** These frames come from our 4D $\rightarrow$ 2D rendering. Pixels in the inpainting masks (defined below) are set to a constant gray value (127), so that the projected frames act as a structural scaffold for the video while clearly indicating where content to be synthesized.
3. **Inpainting masks.** The masks specify the unseen regions that need to be filled. They are constructed from three types of areas: (1) regions that are never observed (novel viewpoints or previously occluded areas), which are naturally identified through the inverse-projection process; (2) locations where the projected Gaussians have low confidence, detected via a threshold on the opacity values; and (3) thin structures and sharp depth discontinuities that may cause projection artifacts (e.g., along object boundaries and occlusion edges), detected by checking whether the relative depth difference exceeds a predefined threshold.

E. More Details of Proposed Metrics

To evaluate the spatial consistency of generated videos in unseen viewpoints, we adopt two standard two-view geometric metrics: *Epipolar Error (EE)* and *Epipolar Inlier Ratio (EIR)*. These metrics quantify how well the generated frames obey the underlying epipolar geometry defined by a fundamental matrix estimated from visual correspondences.

Setup. Given a reference frame I_1 and a generated frame I_2, we extract putative feature correspondences $\{(x_1^{(i)}, x_2^{(i)})\}_{i=1}^{N}$ using SIFT with cross-check and ratio test. The fundamental matrix F is then estimated via RANSAC:

$$F = \arg\min_{F'} \sum_{i \in \mathcal{I}(F')} \mathrm{EE}(x_1^{(i)}, x_2^{(i)}, F'), \tag{1}$$

where $\mathcal{I}(F')$ denotes the RANSAC inlier set.

Epipolar Error (EE). For a correspondence (x_1, x_2) with homogeneous coordinates $\tilde{x}_1 = (x_1^\top, 1)^\top$ and $\tilde{x}_2 = (x_2^\top, 1)^\top$, the epipolar constraint states:

$$\tilde{x}_2^\top F \tilde{x}_1 = 0. \tag{2}$$

Deviations from this constraint reflect geometric inconsistency. We adopt the *Sampson approximation* of the reprojection error:

$$\mathrm{EE}(x_1, x_2; F) = \sqrt{\frac{(\tilde{x}_2^\top F \tilde{x}_1)^2}{(F\tilde{x}_1)_0^2 + (F\tilde{x}_1)_1^2 + (F^\top \tilde{x}_2)_0^2 + (F^\top \tilde{x}_2)_1^2}}. \tag{3}$$

This metric has several desirable properties:

- It is expressed in pixel units and directly interpretable.
- It approximates the *geometric reprojection error* without requiring camera intrinsics.
- It is robust to scale ambiguity inherent to fundamental matrices.

In practice, we report:

$$\mathrm{EE}_{\mathrm{median}} = \mathrm{median}_{i \in \mathcal{I}(F)}\ \mathrm{EE}(x_1^{(i)}, x_2^{(i)}; F),$$

where a lower value indicates better geometric alignment between the generated view and the reference view.

Epipolar Inlier Ratio (EIR). While EE measures the *accuracy* of the geometric alignment, we also evaluate the *stability* of the geometry via an inlier ratio:

$$\mathrm{EIR} = \frac{|\mathcal{I}(F)|}{N}, \tag{4}$$

where N is the total number of matched correspondences before RANSAC. A higher EIR indicates that a larger portion of correspondences can be explained by a valid two-view geometry, suggesting more realistic spatial structure in the generated frame.

EIR is particularly informative in generative settings because:

- generated videos may contain distortions that destroy epipolar geometry,
- RANSAC tends to reject correspondences that violate 3D plausibility,
- a low EIR implies strong geometric hallucination or temporal drift.

Together, EE and EIR evaluate the spatial fidelity of generated videos:

- **EE** reflects how close the video is to a physically plausible two-view geometry.
- **EIR** reflects how consistently the generator maintains global 3D structure.

Both metrics do not require camera intrinsics and thus can be applied to arbitrary videos, making them suitable for evaluating geometry realism in this work.

F. Limitations

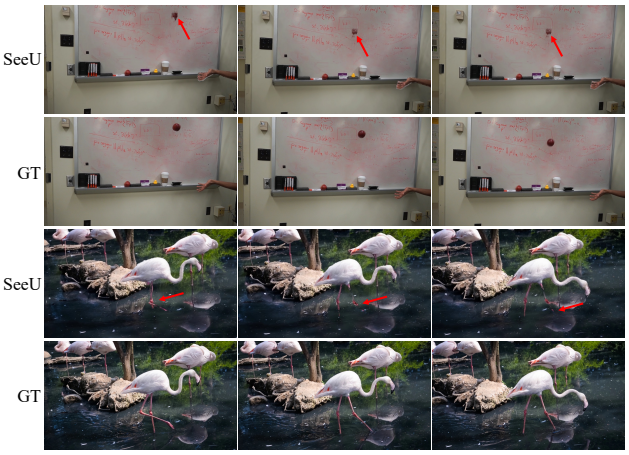


Figure 9. SeeU's performance degrades on inputs containing thin structures or lacking texture, reflecting the inherent limitations of existing base models.

In the first-stage 2D-4D lifting, the performance of SeeU is strongly constrained by the quality of the upstream geometry modules, including camera pose estimation, tracking, and depth prediction. As a result, SeeU requires input videos with salient foreground objects and sufficient spatial details.

As shown in Fig. 9, when the foreground is extremely small or lacks rich texture, these modules become unreliable and the final outputs degrade accordingly. We illustrate such failure cases on small or low-texture foregrounds in the examples below.

G. Robustness and Scalability Analysis

In Fig. 10, we show additional experiments under more challenging cases. These extreme cases involve combinations of adverse factors, including fast-moving cameras or

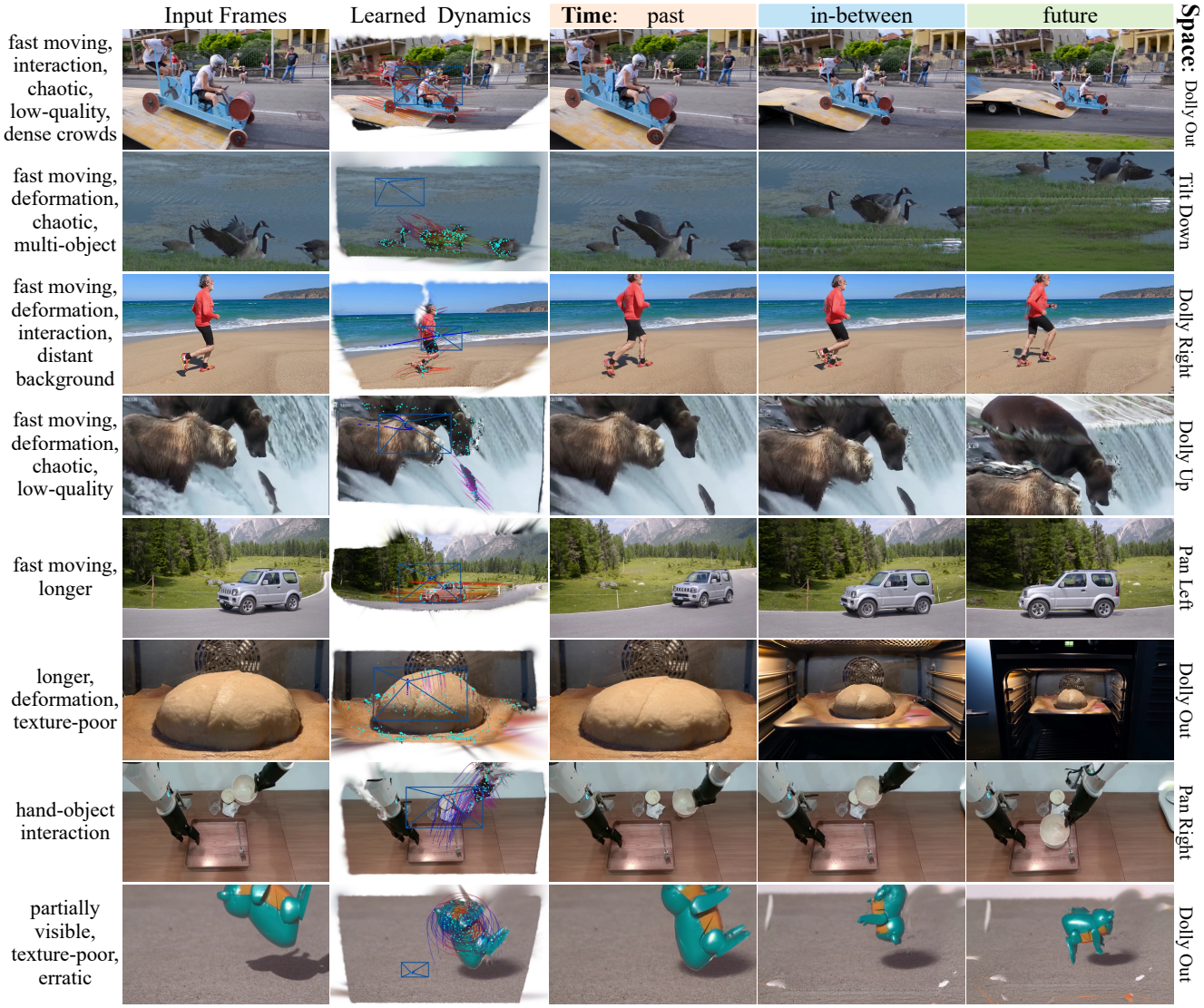


Figure 10. Robustness evaluations (please zoom in).

objects, interactions, chaotic dynamics, low-quality inputs, dense crowds, deformations, multiple objects, longer time span, distant backgrounds, and texture-poor scenes (first column). The results show that SeeU can **effectively learn continuous 4D dynamics (third column)** and generate coherent outputs in unseen time and space, although some artifacts may remain. Additionally, we believe that longer and more complex videos can be decomposed into temporally stable chunks and processed sequentially by SeeU.

H. More Comparison Results

We compare our method with two methods that follow similar 3D/4D memory-based designs, namely DaS [4] and HunyuanWorld-Voyager [5]. As shown in Table 5, our method achieves better performance in both geometric consistency and semantic quality. While all these models inject geometric priors, SeeU **additionally models temporal dynamics**. We will discuss and compare with these approaches in the revised manuscript.

Method	SeeU	DaS	HunyuanWorld-Voyager
EIR↑/CLIP-V↑	**0.8024/0.9588**	0.7624/0.9253	0.7298/0.8911

Table 5. Comparison between 4D-aware models.

I. More Visual Results

We encourage readers to directly watch the **videos** provided in the supplementary materials, as they best demonstrate the

Figure 11. Unseen Temporal World Generated by SeeU.

temporal and spatial behaviors of our method. For completeness, we also include a few representative visual examples below (Fig. 11 to Fig. 13) as a preview.

References

[1] Qingwen Bu, Jisong Cai, Li Chen, Xiuqi Cui, Yan Ding, Siyuan Feng, Xindong He, Xu Huang, et al. Agibot world colosseo: A large-scale manipulation platform for scalable and intelligent embodied systems. In *International Conference on Intelligent Robots and Systems*, 2025. 1

[2] Prateek Chennuri, Yiheng Chi, Enze Jiang, GM Dilshan Godaliyadda, Abhiram Gnanasambandam, Hamid R Sheikh, Istvan Gyongy, and Stanley H Chan. Quanta video restoration. In *European Conference on Computer Vision*, 2024.

[3] Carl Doersch, Ankush Gupta, Larisa Markeeva, Adria Recasens, Lucas Smaira, Yusuf Aytar, Joao Carreira, Andrew Zisserman, and Yi Yang. TAP-Vid: A benchmark for tracking any point in a video. *Advances in Neural Information Processing Systems*, 2022. 1

[4] Zekai Gu, Rui Yan, Jiahao Lu, Peng Li, Zhiyang Dou, Chenyang Si, Zhen Dong, Qifeng Liu, Cheng Lin, Ziwei Liu, Wenping Wang, and Yuan Liu. Diffusion as Shader: 3d-aware video diffusion for versatile video generation control. In *SIGGRAPH*, 2025. 5

[5] Tianyu Huang, Wangguandong Zheng, Tengfei Wang, Yuhao Liu, Zhenwei Wang, Junta Wu, Jie Jiang, Hui Li, Rynson WH Lau, Wangmeng Zuo, and Chunchao Guo. Voyager: Long-

Figure 12. Unseen Spatial World Generated by SeeU.

range and world-consistent video diffusion for explorable 3d scene generation. *arXiv preprint arXiv:2506.04225*, 2025. 5

[6] Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. VACE: All-in-one video creation and editing. In *International Conference on Computer Vision*, 2025. 3

[7] Xun Long Ng, Kian Eng Ong, Qichen Zheng, Yun Ni, Si Yong Yeo, and Jun Liu. Animal kingdom: A large and diverse dataset for animal behavior understanding. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022. 1

[8] F. Perazzi, J. Pont-Tuset, B. McWilliams, L. Van Gool, M. Gross, and A. Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2016. 1

Input Frames

Object Removal by SeeU

Input Frames

Object Replacement by SeeU

Input Frames

Time Lapse by SeeU

Figure 13. Videos edited by SeeU.