Supplementary Materials: Video World Models with Long-term Spatial Memory

Tong Wu*[1], Shuai Yang*[2,4], Ryan Po[1], Yinghao Xu[1],
Ziwei Liu[5], Dahua Lin[3,4], Gordon Wetzstein[1]
[1] Stanford University [2] Shanghai Jiao Tong University [3] The Chinese University of Hong Kong
[4] Shanghai Artificial Intelligence Laboratory [5] S-Lab, Nanyang Technological University

https://spmem.github.io/

In the supplementary material, we provide additional implementation details in Section A, extended experimental results in Section B, and further discussions on related approaches in Section C. We also include more video examples in **video.mp4** to better illustrate the effectiveness and qualitative performance of our method.

A Additional details in methodology

Autoregressive point cloud fusion. In autoregressive generation, the length of the generated video increases over time. At each step, a new static points map is produced and updated into our spatial memory. Since Mega-SAM performs reconstruction in the NDC coordinate system, it is impossible to directly merge results from different stages without alignment, and long video inference will also fail due to CUDA memory limitations. Therefore, unlike the precise reconstruction using Mega-SAM during the data construction stage, we employ CUT3R [5] for 4D reconstruction during the inference stage. CUT3R is a unified online 3D perception framework featuring a stateful recurrent model that incrementally updates a persistent internal state representation with each new observation. Given an image stream (video or unordered photos), the model simultaneously updates its state and predicts metric-scale pointmaps (per-pixel 3D points in a shared world coordinate system) and camera parameters for each input in an online manner. At each inference step, we save the state dict of the current CUT3R model and the parameters of the pose retriever to serve as initialization for the next inference step, ensuring that the reconstruction results of each step remain within the same coordinate system. Therefore, as shown in Figure S1, after tsdf-fusion, the filtered points cloud of the current step can be directly merged and aligned with the previous spatial static memory to achieve autoregressive point cloud fusion.

Details in static point extraction for the dataset. In Mega-SAM, we first resize the input video resolution to 384×672. Based on the initial results, we perform optical flow estimation to refine the estimated camera motion through pixel-level motion cues. Subsequently, a Covariance-based Variable Decomposition strategy is employed to further enhance the robustness and accuracy of the predicted results. In TSDF-fusion, we compute the initial grid dimensions based on the current voxel size and the scene bounds. If the maximum dimension exceeds a predefined threshold (1200), we proportionally scale the voxel size to ensure that the grid dimensions remain within the limit. The final adjusted voxel size is returned. The core principle is to control the grid resolution to prevent memory overflow.

B Additional experimental results

User study setup. We conducted a user study to evaluate the perceptual quality of our generated videos. Specifically, we invited 13 graduate students who are actively engaged in video generation

* denotes equal contribution.

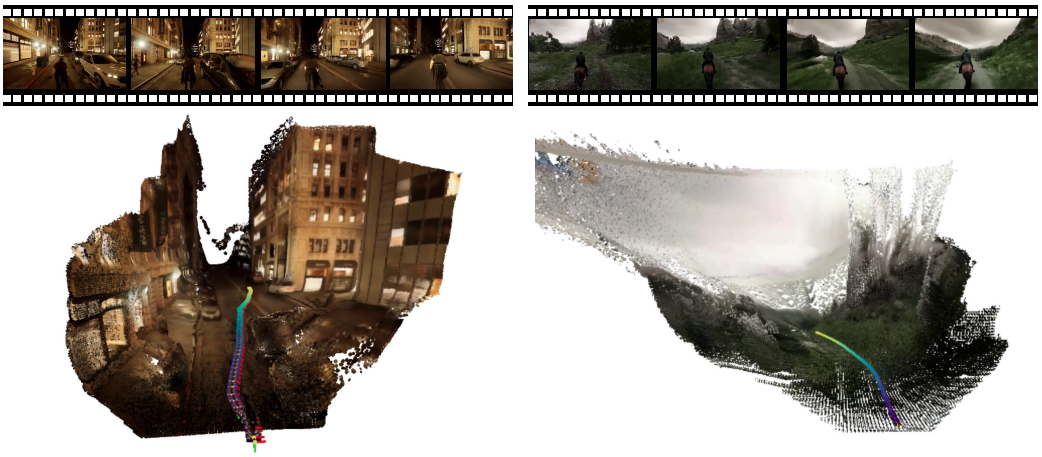

Figure S1: **Autoregressive point cloud fusion.** The system continuously updates spatial memory by integrating newly observed static maps. These maps are reconstructed online using CUT3R in a recurrent manner, while TSDF-Fusion filters out dynamic elements to maintain map consistency.

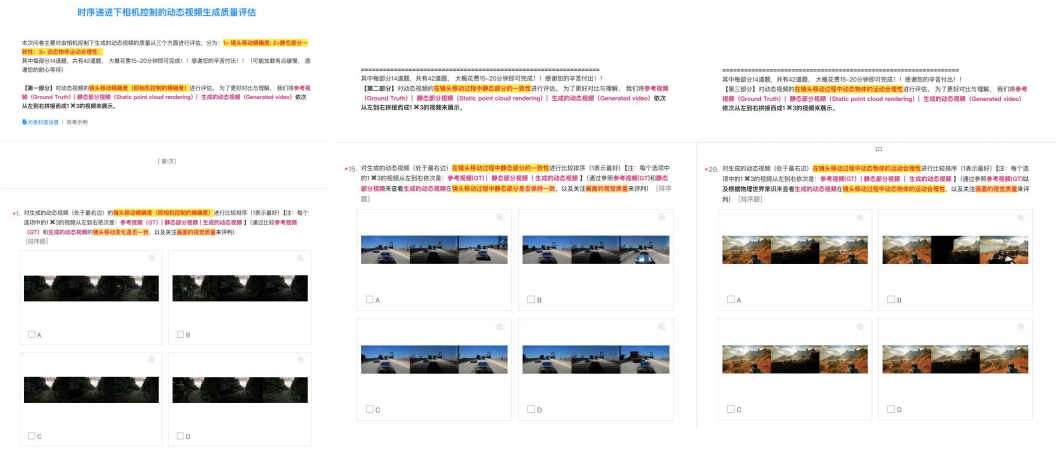

Figure S2: **User study questionnaire screenshots.**

research to participate in the evaluation. Each participant was presented with pairs of ground-truth and generated videos, and asked to rank them based on visual realism and temporal consistency. We report the average human ranking scores derived from their responses to reflect overall perceptual preference. To ensure transparency and clarity, we provide Figure S2, which shows partial screenshots of the user study questionnaire, and Figure S3, which presents partial ranking data for reference.

Camera accuracy measurement. We further evaluate camera pose accuracy along the reverse trajectory. Specifically, in our setting, each pose is visited twice during the sequence: once in the forward pass and once in the reverse, where ideally, the estimated camera poses should match. However, the presence of dynamic elements in the scene poses challenges for traditional SfM methods such as COLMAP. Therefore, we employ CUT3R to estimate the camera trajectories. To quantify the accuracy of the reverse trajectory, we compute both the rotation and translation errors between corresponding camera poses using the following metrics:

$$\text{RotErr} = \frac{1}{n}\frac{180}{\pi} \cdot \sum_{i=1}^{n} \arccos\left(\frac{\text{tr}(\mathbf{R}_0^i \mathbf{R}_{\text{rev}}^{i\top}) - 1}{2}\right), \tag{1}$$

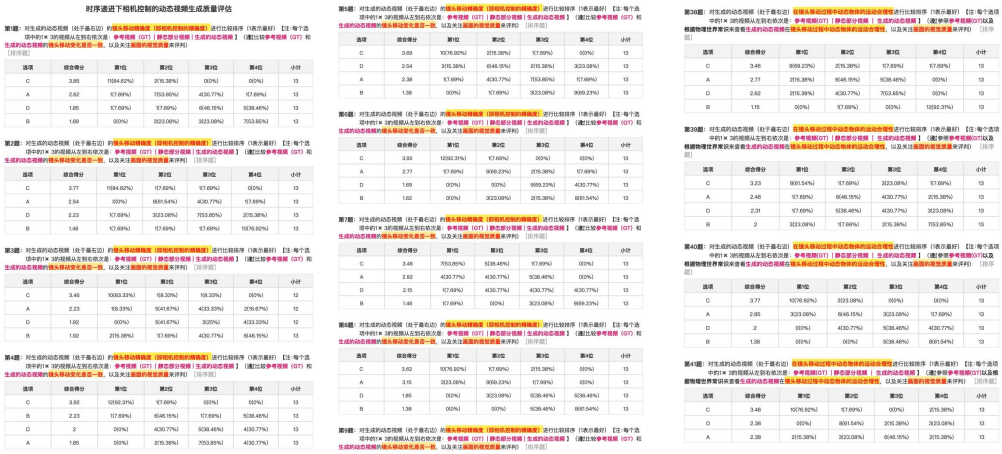

Figure S3: **Partial ranking data.**

Table S1: **Quantitative comparison of camera pose accuracy during revisits.** Our method achieves higher consistency in estimated camera poses when revisiting previously seen views.

Method	TransErr ↓	RotErr ↓
TrajectoryCrafter	0.490	8.938
DaS	2.456	9.454
Wan2.1-Inpainting	1.038	4.927
Ours	**0.431**	**4.011**

$$\text{TransErr} = \frac{1}{n}\sum_{i=1}^{n} \left\| \mathbf{T}_0^i - \mathbf{T}_{\text{rev}}^i \right\|_2^2, \tag{2}$$

where $\mathbf{R}_0^i$ and $\mathbf{R}_{\text{rev}}^i$ denote the first-time and revisiting pose for the same position along the trajectory, respectively, and $\mathbf{T}_0^i$, $\mathbf{T}_{\text{rev}}^i$ are the corresponding translation vectors. The trace operator $\text{tr}(\cdot)$ computes the sum of the diagonal elements of a matrix.

The quantitative results are summarized in Table S1, where our method demonstrates superior performance compared to the baselines. It is worth noting that the trajectories in the MiraData test set typically involve only small rotation angles, which results in generally low RotErr. Additionally, the Wan2.1 inpainting model often tends to generate relatively static videos, leading to limited variation in the estimated camera poses. As a result, despite not being specifically designed for geometry-grounded video generation, it still achieves reasonably good performance.

Since camera-embedded methods such as CameraCtrl [2] and ReCamMaster [1] are designed for tasks that are not aligned with ours, they are either focused on text- or image-conditioned camera-controlled video generation, or on dynamic novel view synthesis within a single temporal sequence. In addition, the sequence lengths they support are different from ours, making it infeasible to include them in a direct comparison.

C Additional discussion on related works.

Learning-Based 3D Reconstruction Traditional structure-from-motion (SfM) pipelines such as COLMAP [3] have long served as the gold standard for image-based 3D reconstruction due to their high accuracy, but they rely on incremental feature matching and bundle adjustment that do not scale well to large-scale or real-time applications. To overcome these limitations, learning-based approaches have emerged: the seminal DUT3R [6] method established a new paradigm of dense matching and optimization in an end-to-end deep learning framework, eliminating the need for separate SfM or multi-view stereo modules. Building on this, recent approaches [8, 4] introduce feed-forward architectures that significantly accelerate inference, achieving near real-time reconstruction

without iterative optimization. To further process dynamic scenes, more recent advances [7**?** , 5] further extend learning-based 3D reconstruction to dynamic scenes by incorporating recurrent models or persistent state, which improves efficiency and robustness in handling moving objects and changing environments. Some of these advances make it feasible to perform online estimation of camera poses and point clouds during video capture or generation.

In this work, we use Mega-SAM during dataset construction to obtain more stable and accurate point cloud and camera pose estimations. However, due to concerns about time efficiency and global alignment, we adopt CUT3R in our iterative video generation process. CUT3R jointly estimates per-frame point clouds while incorporating a dynamic-static disentanglement mechanism to preserve long-term static scene memory. Experimental results indicate that the difference in models does not lead to a significant performance gap.

References

[1] Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuozhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, and Di Zhang. Recammaster: Camera-controlled generative rendering from a single video, 2025. URL `https://arxiv.org/abs/2503.11647`.

[2] Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for video diffusion models. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=Z4evOUYrk7`.

[3] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[4] Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

[5] Qianqian Wang*, Yifei Zhang*, Aleksander Holynski, Alexei A. Efros, and Angjoo Kanazawa. Continuous 3d perception model with persistent state. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025.

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

[7] Junyi Zhang, Charles Herrmann, Junhwa Hur, Varun Jampani, Trevor Darrell, Forrester Cole, Deqing Sun, and Ming-Hsuan Yang. MonST3r: A simple approach for estimating geometry in the presence of motion. In *The Thirteenth International Conference on Learning Representations*, 2025. URL `https://openreview.net/forum?id=lJpqxFgWCM`.

[8] Shangzhan Zhang, Jianyuan Wang, Yinghao Xu, Nan Xue, Christian Rupprecht, Xiaowei Zhou, Yujun Shen, and Gordon Wetzstein. Flare: Feed-forward geometry, appearance and camera estimation from uncalibrated sparse views. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.