FLUIDWORLD: FLUID-LIKE INTERACTIVE DYNAMICS FOR 4D WORLDS

Anonymous authors
Paper under double-blind review

ABSTRACT

Recent advances in generative models have enabled the construction of large-scale, controllable, and realistic 3D scenes and videos. However, these approaches typically produce static scenes or coherent 2D sequences, without maintaining an explicit world state with editable dynamics. In this paper, we propose **FluIDWorld**, an interactive framework for constructing coherent 4D worlds from a single image, designed to support real-time observation and control of fluid-like dynamics. To achieve stable and controllable dynamics during continuous world expansion, it is crucial to obtain reliable motion estimates for newly revealed regions while maintaining consistency with the existing world state. To this end, FluIDWorld incrementally estimates view-grounded motion, aligns each estimate into a consistent global frame via fast geometric alignment, and updates a compact Eulerian velocity field to preserve temporal coherence. This design enables memory-efficient and scalable 4D world generation with low latency, allowing a static scene to be expanded into a temporally coherent 4D world in just 8 seconds on a single GPU, while supporting intuitive motion editing and real-time user feedback.

1 INTRODUCTION

Building models that can represent and simulate environments has been a central goal in interactive simulation, embodied learning, and decision-making. For many downstream applications, such models require environments that are not only visually realistic, but also persistent, manipulable, and capable of evolving over time. One promising direction toward constructing such environments is generative modeling, which enables explicit synthesis of environment structure from data at scale. Recent advances in generative modeling and 3D representations have enabled interactive 3D scene generation (Yu et al., 2025), allowing users to construct and explore spatially coherent environments from minimal input with low latency. However, most existing approaches remain limited to static scenes, making it difficult to capture the temporal evolution of dynamic phenomena within the environment. In parallel, video generation methods (Li et al., 2025a; He et al., 2025) have demonstrated impressive control over camera motion and scene dynamics from single images, but operate on sequences of 2D frames and lack an explicit, reusable environment representation. As a result, current approaches do not yield dynamic world representations that can persist as coherent environments. This gap motivates methods that construct environments with explicit 3D structure and persistent dynamics in open-ended, incremental settings, where real-time interaction is essential for stable and controllable world evolution.

In this paper, we introduce **FluIDWorld**, an interactive framework for 4D scene generation that enables real-time creation and user-driven control of temporally coherent dynamic worlds. We focus on fluid-like environmental dynamics, which are tightly coupled with scene geometry and appearance and cannot be easily decomposed or composed as isolated objects, making them particularly suitable for scene-level interactive modeling. Figure 1 presents our input and interaction setup, along with examples of the generated 4D scenes and 3D scene flows. The framework maintains both global temporal consistency and spatial coherence as the scene expands, allowing intuitive motion editing and large-scale dynamic world construction from a single input image. Starting from a single input image, FluIDWorld employs a single-image animation model (Holynski et al., 2021) to estimate 2D motion in user-specified dynamic regions. As the 3D scene is expanded via FLAGS-based outpainting, each newly synthesized view reveals geometry unseen in the original input. For every such view, we predict a 2D optical flow field and lift it into 3D using the corresponding depth map, yielding sparse scene flow samples over the growing world. Since these flows are predicted independently

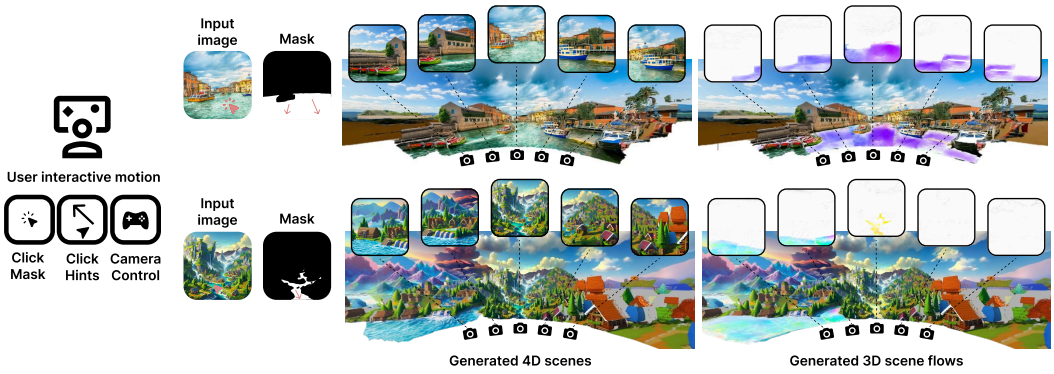

Figure 1: Given a single input image, our method generates a spatiotemporally consistent 4D scene by extending static content into a dynamic world. On the right, we visualize the impact of our alignment module, which preserves motion coherence across newly revealed regions and motion discontinuities.

across views with varying baselines, their lifted 3D motions often differ in scale and orientation, leading to inconsistent global dynamics. To maintain temporal coherence, we introduce a motion alignment module that stabilizes the evolving 3D motion field. The module first applies a closed-form Kabsch initialization to align the current flow to the previously aggregated motion, followed by a lightweight refinement that minimizes residual discrepancies across corresponding motion vectors. This two-stage process provides a fast and robust mechanism for propagating consistent motion as the scene expands.

To further achieve efficiency and scalability, we introduce a compact motion field representation modeled via a hash-encoded neural field, which learns a continuous 3D velocity function from the aligned sparse scene flows. Unlike prior dynamic Gaussian approaches that rely on heavy deformation networks (Wu et al., 2024a; Bae et al., 2024; Yang et al., 2024; Xu et al., 2024) or per-Gaussian optimization (Li et al., 2024; Lee et al., 2024; Yang et al., 2023)—both of which model motion in a Gaussian-centric or frame-specific manner—our method learns a spatially continuous motion field that maps 3D positions to velocity vectors, decoupling motion representation from individual Gaussians. This enables bidirectional motion propagation, where motion is applied forward to future frames and backward to newly emerged regions that lack previous correspondences, effectively filling motion gaps caused by forward warping and enhancing temporal coherence. Parameterized by a compact hash-encoded MLP, the field supports fast spatial queries and scalable motion transfer across large scenes, enabling real-time 4D world synthesis with minimal overhead.

Our contributions are summarized as follows:

- We present **FluIDWorld**, an interactive framework for 4D scene generation from a single image that enables real-time user-driven motion editing and dynamic world construction with low latency.
- We introduce a motion alignment module that stabilizes the evolving 3D motion field by combining closed-form Kabsch initialization with lightweight refinement, ensuring globally coherent motion across expanding views.
- We introduce a compact hash-based motion field that enables bidirectional propagation of continuous 3D motion, filling warping gaps and supporting scalable, coherent 4D scene generation without costly deformation networks or per-Gaussian optimization.

2 Related Works

2.1 3D Scene Generation

The goal of 3D scene generation is to synthesize geometrically consistent and explorable environments from limited inputs such as single images or text prompts. Early methods focused on single-image static scene expansion (Kaneva et al., 2010; Liu et al., 2021). With the rise of diffusion-based generative models, research has progressed toward world-scale, procedurally controllable environments. Some approaches aim to directly construct expansive scenes via terrain synthesis and structured layout generation (Wu et al., 2024b; Hua et al., 2025; Zhou et al., 2025). In parallel, a dominant paradigm follows a render-refine-repeat strategy, where scenes are iteratively rendered,

refined via depth alignment, and updated with newly predicted geometry. Text2Room (Höllein et al., 2023) applies this pipeline to reconstruct mesh-based indoor environments from text prompts. While LucidDreamer (Chung et al., 2023) and Text2Immersion (Ouyang et al., 2023) extends it to realistic indoor-to-outdoor worlds using Gaussian Splatting (Kerbl et al., 2023). WonderJourney (Yu et al., 2024) further incorporates large language models (LLMs) to generate semantically diverse and interconnected scenes. A major breakthrough was introduced by WonderWorld (Yu et al., 2025), which proposed the first interactive framework for real-time, user-controllable 3D world generation. Despite these advancements, most methods remain confined to static environments without temporal dynamics or motion consistency. The integration of temporally coherent motion into interactive 3D environments remains an open challenge, highlighting the need for a new direction toward interactive 4D world generation.

2.2 Single Image Animation

Single-image animation aims to generate dynamic visual effects from static images, particularly focusing on fluid-like elements such as clouds, water, or smoke. Early approaches relied on manual layer decomposition or simple physical models, later replaced by deep networks that inferred motion fields directly from static images (Chuang et al., 2005; Jhou & Cheng, 2015; Endo et al., 2019; Logacheva et al., 2020). Holynski et al. (Holynski et al., 2021) proposed an Eulerian-based model that predicts dense, seamlessly looping flow fields for realistic 2D animations. Building on this, methods such as Text2Cinemagraph (Mahapatra et al., 2023) and StyleCineGAN (Choi et al., 2024) introduced semantic and controllable motion synthesis, and diffusion-based frameworks (Shi et al., 2024; Xing et al., 2025; Shi et al., 2025; Jin et al., 2025) further improved temporal consistency. These 2D approaches produce visually compelling results but remain confined to the image plane and lack geometric understanding. To enhance spatial realism and immersive experience, recent works have extended single-image animation into 3D space by incorporating geometric representations. 3D-Cinemagraphy (Li et al., 2023) and Make-It-4D (Shen et al., 2023) learn implicit 3D scene flow from depth-based layer decomposition, enabling joint modeling of motion and novel view synthesis. 4DGS-Cinemagraphy (Jin et al., 2025) further introduces an explicit Gaussian Splatting (Kerbl et al., 2023) representation and a motion optimization module to generate temporally coherent 3D scene flow across views. However, prior 3D single-image animation methods (Li et al., 2023; Shen et al., 2023) remain limited to localized scene regions and cannot propagate motion into newly exposed areas, lacking global spatial awareness. 4DGS-Cinemagraphy (Jin et al., 2025) that enforce multi-view consistency partially address this issue, but their reliance on dense, iterative optimization makes real-time or interactive use impractical. In contrast, our method targets scalable and interactive 4D world generation by introducing a simple yet effective motion alignment strategy. This alignment preserves both global and geometric consistency across the scene, enabling rapid motion field initialization and efficient optimization.

2.3 4D Scene Generation

Generating temporally coherent and geometrically consistent 4D scenes remains a challenging problem in computer vision and graphics. Early research extended static 3D representations into dynamic sequences by transferring motion priors from video diffusion models. MAV3D (Singer et al., 2023) proposed a hybrid score distillation strategy to inject temporal motion into 3D assets, while subsequent studies refined score-distillation sampling and trajectory-aware optimization to improve motion fidelity (Bahmani et al., 2024b; Zheng et al., 2024; Bahmani et al., 2024a; Zeng et al., 2024). Although these methods generate visually appealing results, they are mostly constrained to object-centric or bounded scenes and struggle to maintain global temporal coherence under large viewpoint variations or long sequences.

Recent works like CAT4D (Wu et al., 2025) extended this direction by leveraging diffusion-based frameworks for dynamic 4D synthesis from limited viewpoints. 4DNeX (Chen et al., 2025b) further simplified 4D generation into a feed-forward paradigm, enabling single-image-to-4D synthesis without iterative optimization. However, because these models generate content within a restricted spatial extent and lack explicit scene geometry, they often suffer from motion inconsistency and temporal flickering when applied to wider camera trajectories. WonderPlay (Li et al., 2025b) explored a complementary perspective by coupling physics-based simulation with generative video modeling, enabling dynamic 3D scene generation from a single image and user-defined actions. While effective for localized interaction and physically plausible motion, this hybrid framework relies on explicit video generation and supervision to guide 3D dynamics. In contrast, our method employs a hash-based encoder to learn continuous motion fields, which are aligned via a lightweight initialization

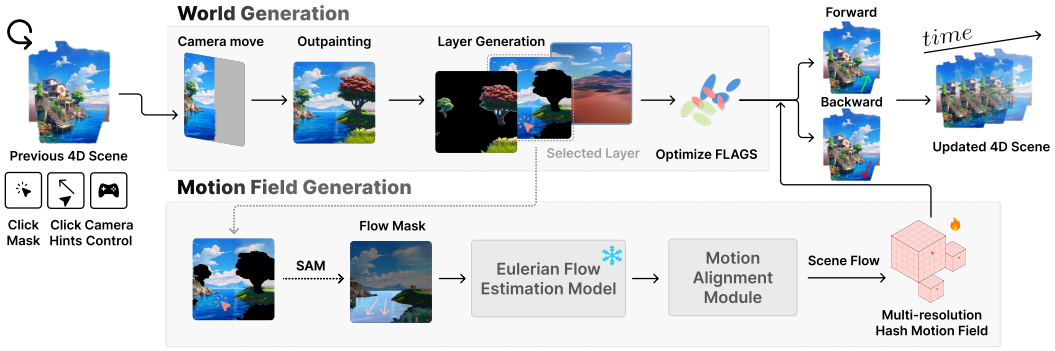

Figure 2: **Overall framework of the proposed method.** The training stage learns a position-to-motion MLP using the 3D Motion Optimization module-generated motion field, while the rendering stage updates Gaussian positions via Euler integration for motion-driven rendering.

step and applied through bidirectional warping of the 4D Gaussian representation. This enables real-time, scalable dynamic scene synthesis with coherent motion propagation across large-scale 3D environments—without requiring video supervision or expensive optimization.

3 METHOD

Before diving into the details, we briefly summarize the structure of our method. Section 3.2 introduces our FLAGS-based 3D scene generation pipeline, which constructs and expands the world layer by layer from a single input image. Section 3.3 then describes how we estimate 2D motion, lift it into 3D, and refine it through our motion alignment module to obtain globally consistent scene flows. Finally, Section 3.4 presents our hash-based motion field and bidirectional integration strategy, which enable continuous, real-time motion propagation and dynamic 4D rendering.

3.1 OVERVIEW

FluIDWorld constructs an interactive 4D world by jointly expanding a 3D scene and propagating motion as new content is synthesized. As illustrated in Fig. 2, our pipeline consists of four tightly coupled components—*layer-wise 3D scene generation*, *2D motion estimation*, *motion alignment and field learning*, and *motion-driven rendering*—which operate together in an iterative loop. Starting from a single input image, our system generates an initial FLAGS-based 3D scene and then iteratively grows both geometry and dynamics as the user navigates through the world. User-controlled camera poses C_{gen} and text prompts $\mathcal{U}$ guide the spatial and semantic expansion of the scene, while sparse motion cues—specified through seed points and directional hints—determine which regions should exhibit dynamic behavior. As each novel view exposes geometry not visible in the original image, FluIDWorld predicts 2D optical flow for the newly revealed regions using a single-image animation model. These 2D flows are lifted into 3D via depth, forming sparse scene-flow samples over the expanded world. Because these flows are estimated independently for each view, they may be inconsistent across baselines; thus, we refine them using a motion alignment module that aligns current flows to the previously accumulated global motion.

The aligned scene flows supervise the learning of a multi-resolution hash-based motion field F_θ, which provides a continuous 3D velocity function over the entire world. By querying this field, all Gaussians in the scene can be updated in real time, enabling coherent motion propagation throughout the expanding environment. Through this unified pipeline—scene expansion, motion estimation, alignment, and continuous field learning—FluIDWorld achieves interactive 4D scene generation with consistent dynamics at world scale.

3.2 SCENE GENERATION VIA FLAGS REPRESENTATION

We adopt the FLAGS representation (Yu et al., 2025) as our underlying 3D scene structure. FLAGS organizes a scene $\mathcal{E}$ as a union of three spatial layers—foreground ($\mathcal{L}_{\text{fg}}$), background ($\mathcal{L}_{\text{bg}}$), and sky ($\mathcal{L}_{\text{sky}}$)—each composed of simplified 3D Gaussian surfels parameterized by position, scale, orientation, opacity, and color. Compared to 3D Gaussian Splatting (3DGS) (Kerbl et al., 2023), FLAGS removes view-dependent color components and flattens surfels along the z-axis to enable faster rendering and optimization.

Single-view Scene Construction. To build a structured 3D scene from a single image I_{init}, we follow the FLAGS pipeline. The input image is decomposed into semantic layers using depth edges, segmentation masks, and sky detection. Occluded regions are inpainted using prompt-guided diffusion, and each resulting layer is converted into surfels to form the 3D scene.

Layer-wise Optimization. Each layer is optimized in a back-to-front order to match the original image while maintaining consistency across layers. Specifically, the sky and background layers are first optimized using masked ℓ_1 and perceptual losses, followed by the foreground. We optimize only a subset of surfel attributes (opacity, orientation, and scale), keeping color and position fixed.

Scene Extrapolation. To expand the scene beyond the initial view, we perform camera extrapolation and generate depth via guided depth diffusion (GDD) (Yu et al., 2025). To enforce geometric consistency with the existing scene, we render a partial depth map D_{guide} and visibility mask M_{guide} from the current 3D scene. These are used to guide depth prediction for the new view, resulting in (I_i, D_i) pairs that are converted to FLAGS surfels and merged into the global scene. Although our scene construction process adopts components from prior work, we integrate and tailor them to serve as a reliable and editable 3D foundation for our dynamic world generation framework.

3.3 Motion Field Generation

Flow Estimation from a Single Image. We represent motion using an *Eulerian flow* (Holynski et al., 2021) formulation, a technique widely used in single-image animation to model fluid-like dynamics. Unlike Lagrangian methods that track individual points over time, Eulerian flow defines a dense motion field directly over the image domain by assigning a velocity vector to each pixel.

Given an image I, the model predicts a per-pixel motion field $\mathbf{M}_t(\mathbf{x})$ that represents the instantaneous velocity at location $\mathbf{x}$ and time t. The future pixel location is estimated via Euler integration:

$$\mathbf{x}_{t+1} = \mathbf{x}_t + \mathbf{M}_t(\mathbf{x}_t), \tag{1}$$

where $\mathbf{x}_t$ denotes the pixel coordinates at time t. To obtain cumulative displacement from the initial frame, we recursively integrate the flow:

$$\mathbf{F}_{0\to t}(\mathbf{x}_0) = \mathbf{F}_{0\to t-1}(\mathbf{x}_0) + \mathbf{M}_t(\mathbf{x}_0 + \mathbf{F}_{0\to t-1}(\mathbf{x}_0)), \tag{2}$$

where $\mathbf{F}_{0\to t}(\cdot)$ denotes the accumulated flow from time 0 to t. This formulation allows us to estimate motion across both the input image I_{init} and extrapolated views $\{I_i\}$ using only image-space velocities. We denote the Eulerian flow predictor as $\text{EF}(\cdot)$ and apply it to each view to obtain a sparse 2D motion field:

$$\mathbf{F}_i = \text{EF}(I_i). \tag{3}$$

These sparse flows are not directly used as final motion cues; instead, they serve as intermediate guidance signals in our Motion Alignment Module, which lifts and refines them into a consistent 3D motion field over time. We now describe how this process constructs a temporally coherent 4D motion representation.

Motion Alignment Module. As illustrated in Figure 3, users first provide a set of 2D seed points $\{p_i\}$ that indicate spatial regions intended to exhibit motion. These seed points are used to extract binary segmentation masks $\{m_i\}$ (e.g., via SAM (Kirillov et al., 2023)) that define the spatial extent of dynamic areas. Each mask m_i is paired with a directional motion hint h_i indicating the desired flow direction within the region. Given this input, we apply the Eulerian Flow model to each generated scene image I_i, along with the set of motion-guided masks $\{m_i, h_i\}$, to estimate a sparse 2D flow field:

$$\mathbf{F}_i = \text{EF}(I_i, \{m_i, h_i\}). \tag{4}$$

To lift the 2D flow $\mathbf{F}_i$ into a 3D scene flow $\mathbf{S}_i$, we first unproject it using the corresponding depth map D_i. However, since 2D flows are predicted independently for each view, the resulting 3D scene flows often

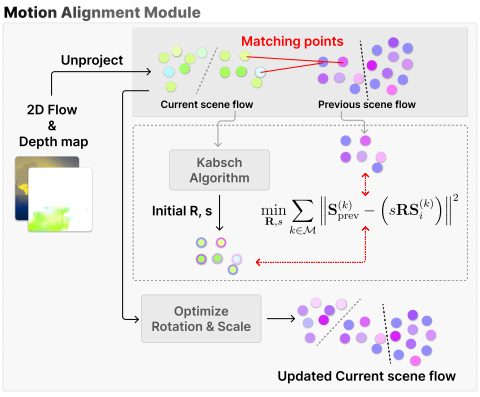

Figure 3: Motion Alignment Module for resolving local ambiguity and propagating temporally consistent motion.

exhibit inconsistencies in motion direction and magnitude, even across geometrically similar regions. A similar alignment objective is also explored in the 4DGS-Cinemagraphy (3D-MOM) (Jin et al., 2025), which optimizes motion consistency across views via reprojected 2D flows. However, due to random initialization and image-space supervision, this approach suffers from slow convergence and unstable alignment in non-overlapping or view-specific regions. To enforce spatial consistency, we treat the previously generated scene flows $\mathbf{S}_{\text{prev}} = \{\mathbf{S}_1, \ldots, \mathbf{S}_{i-1}\}$ as pseudo-ground-truth and align the current flow $\mathbf{S}_i$ accordingly. This alignment problem is formulated as finding the optimal rotation $\mathbf{R} \in SO(3)$ and uniform scale s that minimize the discrepancy between corresponding 3D motion vectors at spatially matched points:

$$\arg\min_{\mathbf{R},s} \sum_{k\in\mathcal{M}} \left\| \mathbf{S}_{\text{prev}}^{(k)} - \left(s\mathbf{R}\mathbf{S}_i^{(k)} \right) \right\|^2, \tag{5}$$

where $\mathcal{M}$ denotes the set of 3D point correspondences between the current and previously generated scenes.

To solve this efficiently, we adopt the *Kabsch algorithm* (Kabsch, 1976) to obtain a closed-form solution for $\mathbf{R}$ via singular value decomposition (SVD), and compute s by solving a 1D least-squares problem. Since the Kabsch algorithm has linear complexity with respect to the number of points (i.e., $O(n)$ when the dimension is fixed), this provides a fast and stable initialization for alignment. We further refine the transformation via lightweight gradient-based optimization to minimize residual misalignments. This strategy effectively enforces globally coherent 3D motion, even in regions where image-space flow is unreliable.

Multi-resolution Hash Motion Field. Prior approaches to dynamic 3D scene modeling—such as Deformation Networks (Wu et al., 2024a; Bae et al., 2024) or time-parameterized functions (Li et al., 2024)—represent temporal changes in each Gaussian using per-Gaussian deformation coefficients or large MLPs conditioned on time. These methods follow a Lagrangian view of motion, tracking each primitive individually over time. While expressive, they typically require dense video supervision and involve costly optimization for each Gaussian, making them less suitable for interactive generation.

In contrast, we adopt an Eulerian formulation extended to 3D space. Rather than tracking Gaussians, we define a neural model F_θ that maps any 3D location $\mathbf{x}$ to a motion vector, effectively modeling a continuous motion field over space. This enables global reasoning over motion and allows consistent application to any Gaussian primitive without per-instance fitting. However, extending Eulerian flow to unstructured 3D space poses two key challenges: (1) 3D space lacks a regular grid structure, and (2) motion supervision is only available sparsely through estimated scene flow. To efficiently model motion across unstructured 3D space with sparse supervision, we parameterize F_θ using a multi-resolution hash encoding (Müller et al., 2022) and a lightweight MLP. At each resolution level, the input position $\mathbf{x} \in \mathbb{R}^3$ is discretized and mapped into a compact hash table via spatial hashing:

$$h_\ell(\mathbf{x}) = \left(\bigoplus_{j=1}^{3} \lfloor x_j^\ell \rfloor \cdot \pi_j \right) \bmod T, \tag{6}$$

where π_j are large primes, $\bigoplus$ denotes bitwise XOR, and T is the size of the hash table (e.g., $T = 2^{19}$). This produces compact feature indices into a set of learnable embedding tables across multiple levels. Features from all levels are interpolated and concatenated, and the aggregated feature vector is passed to an MLP to regress the motion:

$$F_\theta(\mathbf{x}) = \text{MLP}_\theta \left(\bigoplus_{\ell=1}^{L} \text{Interp}_\ell \left(\text{HashEnc}_\ell(\mathbf{x}) \right) \right). \tag{7}$$

Given sparse scene flow samples $\{(\mathbf{x}_i, \mathbf{s}_i)\}_{i=1}^N$, the model is trained with a simple regression objective:

$$\mathcal{L}_{\text{motion}} = \sum_{i=1}^{N} \| F_\theta(\mathbf{x}_i) - \mathbf{s}_i \|_2^2 . \tag{8}$$

After training, the motion model F_θ can predict temporally coherent motion vectors for arbitrary 3D locations. This allows us to assign motion to any Gaussian primitive in the scene without requiring expensive per-Gaussian optimization, enabling fast, scalable motion propagation across dynamic scenes.

Bidirectional Integration of Motion Field. Similar to 2D flow integration in image space (Eq. 1, 2), we update the position of each Gaussian over time by performing discrete Euler integration on the learned motion field $F_\theta : \mathbb{R}^3 \to \mathbb{R}^3$. This converts the spatially defined velocity field into temporally continuous motion trajectories suitable for rendering. Given a Gaussian primitive g with center position $\mathbf{p}_g(t) \in \mathbb{R}^3$ at time t, we define its bidirectional trajectories using forward and backward integration:

$$\mathbf{p}_g^f(t) = \mathbf{p}_g^f(t-1) + \boldsymbol{\psi} \odot F_\theta\big(\mathbf{p}_g^f(t-1)\big), \tag{9}$$

$$\mathbf{p}_g^b(t) = \mathbf{p}_g^b(t-1) - \boldsymbol{\psi} \odot F_\theta\big(\mathbf{p}_g^b(t-1)\big), \tag{10}$$

where $\boldsymbol{\psi} \in \mathbb{R}^3$ denotes a per-axis step size vector, and $\odot$ indicates element-wise multiplication. The forward term $\mathbf{p}_g^f(t)$ advances each Gaussian along the velocity direction predicted by F_θ, while the backward term $\mathbf{p}_g^b(t)$ integrates in the opposite direction to maintain temporal symmetry. This bidirectional formulation reduces integration drift accumulated over time and enforces temporal symmetry. As a result, the motion trajectories remain stable even during long-term propagation.

After computing both trajectories, we construct the final set of Gaussians by merging them with temporal alignment:

$$\mathcal{P}(t) = \big\{\mathbf{p}_g^f(t), \mathbf{p}_g^b(T-t) \mid \mathbf{m}_g = 1\big\} \cup \big\{\mathbf{p}_g^{\text{static}} \mid \mathbf{m}_g = 0\big\}. \tag{11}$$

Only Gaussians within motion regions ($\mathbf{m}_g = 1$) are updated and merged, while static ones remain fixed, effectively reducing rendering overhead. This bidirectional design ensures coverage of both forward- and backward-warped regions, filling in areas that might otherwise suffer from density loss due to one-way motion propagation. By complementing the motion field from both directions, our method preserves spatial continuity at motion boundaries and maintains temporal coherence throughout the scene.

4 Experiments

4.1 Baselines

In principle, we aim to compare our framework against existing state-of-the-art methods. However, to the best of our knowledge, no prior work directly tackles the task of *interactive 4D world generation* from a single image with user-controllable camera and motion hints. Therefore, we organize our baselines into two complementary groups to evaluate different aspects of our approach.

4D Scene-based Baselines. To analyze how each component contributes to spatiotemporal consistency and efficient 4D world generation, we consider two representative scene-based baselines built upon WonderWorld (Yu et al., 2025). The first is a naive scene flow variant that directly lifts predicted 2D optical flow into 3D using depth, without any refinement or alignment across views. The second replaces our alignment module with the optimization-based 3D Motion Optimization Module (3D-MOM) from *4DGS-Cinemagraphy* (Jin et al., 2025), which enforces reprojection consistency across overlapping views but assumes strong view overlap.

Video Generation Baselines. In addition, to compare physical realism and perceptual visual quality against video-only generation approaches, we include two recent video generation methods, *PuppetMaster* (Li et al., 2025a) and *Matrix-game2.0* (He et al., 2025), as baselines. Since these methods do not produce explicit 3D representations or scene flow, we evaluate them using non-reference metrics that assess physical plausibility and visual fidelity.

4.2 Metrics

We evaluate motion consistency using Mean Cosine Alignment (MCA) and Flow Magnitude Variance (FMV). MCA measures directional coherence of scene flow by computing cosine similarity between neighboring motion vectors, while FMV captures local consistency in flow magnitude across neighboring points. To assess physical realism and perceptual visual quality for video-based comparisons, we adopt GPT-4o-based non-reference metrics (Chen et al., 2025a). Specifically, *PhysReal* evaluates the physical plausibility of generated dynamics, and *PhotoReal* measures perceptual visual quality. Further details of these metrics are provided in the Supplementary.

4.3 Implementation Details

We adopt the Stable Diffusion Inpainting model (Rombach et al., 2022) for outpainting and text-to-image generation. We employ an Eulerian flow prediction model from 3D Cinemagraphy (Li

Method	Time (s)	MCA (↑)	FMV (↓)
WonderWorld (Yu et al., 2025) + Naive Scene Flow	**16.2**	0.0550	1.91
WonderWorld (Yu et al., 2025) + 3D-MOM (Jin et al., 2025)	600.0	0.0597	1.66
Ours (full)	18.5	**0.0742**	**0.29**

Table 1: **Runtime and motion consistency comparison.** Our method achieves the best motion alignment (MCA) and magnitude consistency (FMV) with minimal runtime overhead.

Method	PhysReal ↑	PhotoReal ↑
WonderWorld (Yu et al., 2025) + Naive	0.655	0.609
WonderWorld (Yu et al., 2025) + 3D-MoM (Jin et al., 2025)	0.882	0.768
PuppetMaster (Li et al., 2025a)	0.582	0.427
Matrix-game2.0 (He et al., 2025)	0.877	0.723
Ours	**0.927**	**0.860**

Table 2: **Comparison of physical realism and visual quality.** Our method achieves higher physical realism (PhysReal) and perceptual visual quality (PhotoReal) than video generation baselines.

et al., 2023) to estimate dense 2D motion. To extract motion region masks from user-provided points, we use the Segment Anything Model (SAM) (Kirillov et al., 2023). Depth is estimated using the pretrained MoGeV2 model (Wang et al., 2025). All models are publicly available and used without fine-tuning to isolate the contribution of our motion alignment and generation pipeline.

4.4 Quantitative Results

Runtime and Motion Consistency. Table 1 presents quantitative results on generation time and motion consistency. WonderWorld + Naive Scene Flow constructs a 4D scene by accumulating flow in newly revealed regions without motion alignment, achieving the shortest runtime but exhibiting the lowest MCA and highest FMV. This indicates unstable directional coherence and inconsistent local motion strength across the scene. WonderWorld + 3D-MOM improves MCA through reprojection-based alignment across overlapping views; however, the optimization is restricted to overlapping regions, which degrades global motion consistency in settings with freely varying camera trajectories. In addition, per-point optimization leads to inconsistent flow magnitudes, resulting in relatively high FMV values. In contrast, our method constructs a complete 4D scene in 18.5 seconds, including the entire scene generation pipeline. Motion generation takes 8 seconds in total, with only 2 seconds devoted to motion alignment. By leveraging deterministic initialization and lightweight 3D alignment, our approach minimizes iterative optimization cost while achieving the highest MCA and lowest FMV, yielding more stable and consistent motion.

Comparison with Video Generation Models. To enable a direct comparison with video-only generation approaches, we further evaluate the physical realism and perceptual visual quality of the generated dynamic scenes using GPT-4o-based metrics (Table 2). Specifically, we assess *PhysReal* and *PhotoReal* and compare our method against video generation baselines. Our approach consistently outperforms these methods on both metrics, demonstrating that explicitly modeling dynamics in 3D space leads to more physically coherent and visually realistic world evolution than video-only generation models that lack an underlying 3D motion structure.

4.5 Qualitative Results

Appendix C provides supplementary qualitative results that cover diverse fluid-like motions such as fire, smoke, and clouds. In Fig. 4 we visualize the full 3D scene flow across the generated world to compare the spatiotemporal consistency of different approaches. WonderWorld + Naive Scene Flow directly lifts independent 2D predictions into 3D without refinement, causing severe discontinuities and poor motion propagation across views. For WonderWorld + 3D-MOM, motion refinement is performed via reprojection consistency across overlapping views. It achieves smoother motion in overlapping views but lacks boundary awareness, often producing fragmented motion near dynamic regions. Next, in the Ours w/o Initialization setting, optimization provides partial alignment across views, but the randomly initialized rotation and scale often lead to unstable convergence. This instability is especially pronounced in geometrically unconstrained regions or areas with complex structure, where misaligned initial flow can propagate inconsistently. In contrast, our full model uses a deterministic initialization via the SVD-based Kabsch algorithm, providing a well-aligned starting point. This enables faster convergence and more stable optimization, leading to coherent motion

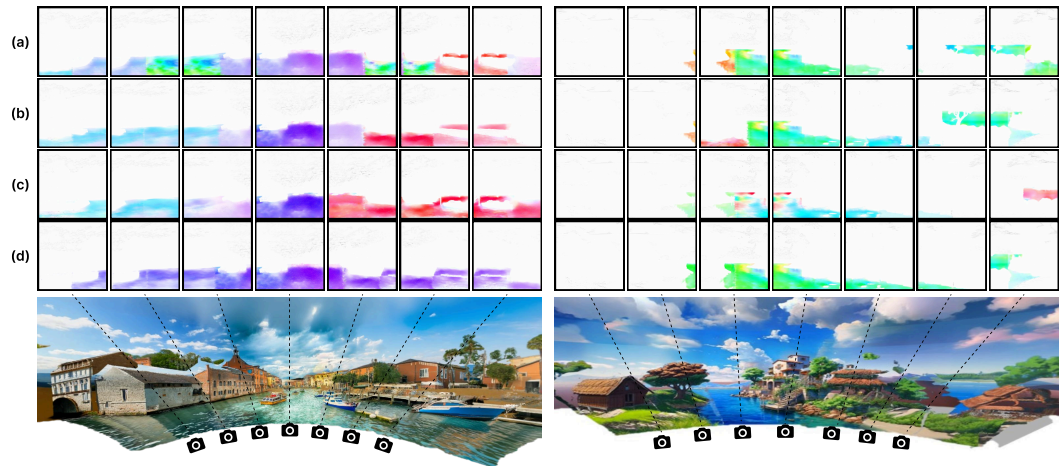

Figure 4: Qualitative comparison of motion consistency across different methods on the same input image. (a) Naive Scene Flow, (b) WonderWorld (Yu et al., 2025) + 3D-MOM, (c) Ours w/o initialization, and (d) Ours.

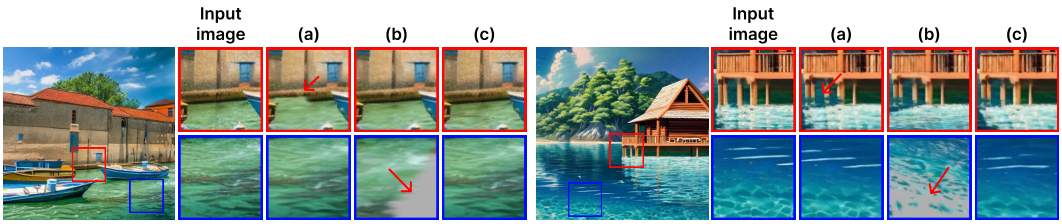

Figure 5: Ablation study of motion consistency across different methods on the same input image. (a) w/o hash-based motion field, (b) w/o bidirectional motion integration, (c) Ours

even in high-frequency or boundary regions. As a result, the model achieves superior spatiotemporal motion consistency across the entire scene.

4.6 Ablation Study

To evaluate the effectiveness of our key design components, we conduct an ablation study on (1) the hash-based motion field and (2) bidirectional motion integration. We first assess a variant that directly applies the 3D scene flow as the motion field via Euler integration. Without learning a dedicated field, the same motion vector is accumulated over time for each Gaussian, leading to unrealistic motion magnitudes and severe temporal artifacts—especially in regions with complex dynamics. Next, we ablate our bidirectional formulation by using only a unidirectional (forward) motion field. While this variant produces temporally smooth motion within visible regions, it fails to preserve density in highly dynamic areas. Forward-only integration causes Gaussians to drift or vanish over time, resulting in visible holes and unstable rendering near motion boundaries. In contrast, our full model learns a compact, hash-based motion field that defines motion over the global 3D space (Fig. 5). Each Gaussian retrieves motion based on its current spatial location, avoiding redundant accumulation and ensuring temporal consistency. Combined with bidirectional integration, our method maintains density in dynamic regions and produces temporally stable, artifact-free renderings.

5 Conclusion

We present a real-time framework for generating dynamic 4D scenes from a single image by extending static 3D geometry into motion-rich worlds. Our method first predicts 2D motion using a pre-trained animation model, then incrementally builds a 4D scene with a lightweight, deterministically initialized alignment module that ensures consistent motion propagation. It preserves fine spatial detail and coherent temporal dynamics, achieving high perceptual quality with minimal computational cost—making it suitable for interactive applications. However, our method models dynamics using a continuous Eulerian motion field, which is well suited for fluid-like motions but does not explicitly enforce rigid-body constraints. Extending the framework with object-centric motion models remains an important direction for future work.

REFERENCES

Jeongmin Bae, Seoha Kim, Youngsik Yun, Hahyun Lee, Gun Bang, and Youngjung Uh. Per-gaussian embedding-based deformation for deformable 3d gaussian splatting. In *European Conference on Computer Vision*, pp. 321–335. Springer, 2024.

Sherwin Bahmani, Xian Liu, Wang Yifan, Ivan Skorokhodov, Victor Rong, Ziwei Liu, Xihui Liu, Jeong Joon Park, Sergey Tulyakov, Gordon Wetzstein, et al. Tc4d: Trajectory-conditioned text-to-4d generation. In *European Conference on Computer Vision*, pp. 53–72. Springer, 2024a.

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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7996–8006, 2024b.

Boyuan Chen, Hanxiao Jiang, Shaowei Liu, Saurabh Gupta, Yunzhu Li, Hao Zhao, and Shenlong Wang. Physgen3d: Crafting a miniature interactive world from a single image. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 6178–6189, 2025a.

Zhaoxi Chen, Tianqi Liu, Long Zhuo, Jiawei Ren, Zeng Tao, He Zhu, Fangzhou Hong, Liang Pan, and Ziwei Liu. 4dnex: Feed-forward 4d generative modeling made easy. *arXiv preprint arXiv:2508.13154*, 2025b.

Jongwoo Choi, Kwanggyoon Seo, Amirsaman Ashtari, and Junyong Noh. Stylecinegan: Landscape cinemagraph generation using a pre-trained stylegan. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7872–7881, 2024.

Yung-Yu Chuang, Dan B Goldman, Ke Colin Zheng, Brian Curless, David H Salesin, and Richard Szeliski. Animating pictures with stochastic motion textures. In *ACM SIGGRAPH 2005 Papers*, pp. 853–860. 2005.

Jaeyoung Chung, Suyoung Lee, Hyeongjin Nam, Jaerin Lee, and Kyoung Mu Lee. Luciddreamer: Domain-free generation of 3d gaussian splatting scenes. *arXiv preprint arXiv:2311.13384*, 2023.

Yuki Endo, Yoshihiro Kanamori, and Shigeru Kuriyama. Animating landscape: self-supervised learning of decoupled motion and appearance for single-image video synthesis. *arXiv preprint arXiv:1910.07192*, 2019.

Xianglong He, Chunli Peng, Zexiang Liu, Boyang Wang, Yifan Zhang, Qi Cui, Fei Kang, Biao Jiang, Mengyin An, Yangyang Ren, et al. Matrix-game 2.0: An open-source real-time and streaming interactive world model. *arXiv preprint arXiv:2508.13009*, 2025.

Lukas Höllein, Ang Cao, Andrew Owens, Justin Johnson, and Matthias Nießner. Text2room: Extracting textured 3d meshes from 2d text-to-image models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 7909–7920, 2023.

Aleksander Holynski, Brian L Curless, Steven M Seitz, and Richard Szeliski. Animating pictures with eulerian motion fields. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 5810–5819, 2021.

Tongyan Hua, Lutao Jiang, Ying-Cong Chen, and Wufan Zhao. Sat2city: 3d city generation from a single satellite image with cascaded latent diffusion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27978–27988, 2025.

Wei-Cih Jhou and Wen-Huang Cheng. Animating still landscape photographs through cloud motion creation. *IEEE Transactions on Multimedia*, 18(1):4–13, 2015.

In-Hwan Jin, Haesoo Choo, Seong-Hun Jeong, Park Heemoon, Junghwan Kim, Oh-joon Kwon, and Kyeongbo Kong. Optimizing 4d gaussians for dynamic scene video from single landscape images. In *The Thirteenth International Conference on Learning Representations*, 2025.

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

Biliana Kaneva, Josef Sivic, Antonio Torralba, Shai Avidan, and William T Freeman. Infinite images: Creating and exploring a large photorealistic virtual space. *Proceedings of the IEEE*, 98(8): 1391–1407, 2010.

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

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 4015–4026, 2023.

Junoh Lee, ChangYeon Won, Hyunjun Jung, Inhwan Bae, and Hae-Gon Jeon. Fully explicit dynamic gaussian splatting. *Advances in Neural Information Processing Systems*, 37:5384–5409, 2024.

Ruining Li, Chuanxia Zheng, Christian Rupprecht, and Andrea Vedaldi. Puppet-master: Scaling interactive video generation as a motion prior for part-level dynamics. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 13405–13415, 2025a.

Xingyi Li, Zhiguo Cao, Huiqiang Sun, Jianming Zhang, Ke Xian, and Guosheng Lin. 3d cinemagraphy from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4595–4605, 2023.

Zhan Li, Zhang Chen, Zhong Li, and Yi Xu. Spacetime gaussian feature splatting for real-time dynamic view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 8508–8520, 2024.

Zizhang Li, Hong-Xing Yu, Wei Liu, Yin Yang, Charles Herrmann, Gordon Wetzstein, and Jiajun Wu. Wonderplay: Dynamic 3d scene generation from a single image and actions. *arXiv preprint arXiv:2505.18151*, 2025b.

Andrew Liu, Richard Tucker, Varun Jampani, Ameesh Makadia, Noah Snavely, and Angjoo Kanazawa. Infinite nature: Perpetual view generation of natural scenes from a single image. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 14458–14467, 2021.

Elizaveta Logacheva, Roman Suvorov, Oleg Khomenko, Anton Mashikhin, and Victor Lempitsky. Deeplandscape: Adversarial modeling of landscape videos. In *European Conference on Computer Vision*, pp. 256–272. Springer, 2020.

Aniruddha Mahapatra, Aliaksandr Siarohin, Hsin-Ying Lee, Sergey Tulyakov, and Jun-Yan Zhu. Text-guided synthesis of eulerian cinemagraphs. *ACM Transactions on Graphics (TOG)*, 42(6): 1–13, 2023.

Thomas Müller, Alex Evans, Christoph Schied, and Alexander Keller. Instant neural graphics primitives with a multiresolution hash encoding. *ACM transactions on graphics (TOG)*, 41(4): 1–15, 2022.

Hao Ouyang, Kathryn Heal, Stephen Lombardi, and Tiancheng Sun. Text2immersion: Generative immersive scene with 3d gaussians. *arXiv preprint arXiv:2312.09242*, 2023.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10684–10695, 2022.

Liao Shen, Xingyi Li, Huiqiang Sun, Juewen Peng, Ke Xian, Zhiguo Cao, and Guosheng Lin. Make-it-4d: Synthesizing a consistent long-term dynamic scene video from a single image. In *Proceedings of the 31st ACM International Conference on Multimedia*, pp. 8167–8175, 2023.

Shuwei Shi, Biao Gong, Xi Chen, Dandan Zheng, Shuai Tan, Zizheng Yang, Yuyuan Li, Jingwen He, Kecheng Zheng, Jingdong Chen, et al. Motionstone: Decoupled motion intensity modulation with diffusion transformer for image-to-video generation. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 22864–22874, 2025.

Xiaoyu Shi, Zhaoyang Huang, Fu-Yun Wang, Weikang Bian, Dasong Li, Yi Zhang, Manyuan Zhang, Ka Chun Cheung, Simon See, Hongwei Qin, et al. Motion-i2v: Consistent and controllable image-to-video generation with explicit motion modeling. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–11, 2024.

Uriel Singer, Shelly Sheynin, Adam Polyak, Oron Ashual, Iurii Makarov, Filippos Kokkinos, Naman Goyal, Andrea Vedaldi, Devi Parikh, Justin Johnson, et al. Text-to-4d dynamic scene generation. *arXiv preprint arXiv:2301.11280*, 2023.

Ruicheng Wang, Sicheng Xu, Yue Dong, Yu Deng, Jianfeng Xiang, Zelong Lv, Guangzhong Sun, Xin Tong, and Jiaolong Yang. Moge-2: Accurate monocular geometry with metric scale and sharp details. *arXiv preprint arXiv:2507.02546*, 2025.

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, 2024a.

Rundi Wu, Ruiqi Gao, Ben Poole, Alex Trevithick, Changxi Zheng, Jonathan T. Barron, and Aleksander Holynski. Cat4d: Create anything in 4d with multi-view video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 26057–26068, 2025.

Zhennan Wu, Yang Li, Han Yan, Taizhang Shang, Weixuan Sun, Senbo Wang, Ruikai Cui, Weizhe Liu, Hiroyuki Sato, Hongdong Li, et al. Blockfusion: Expandable 3d scene generation using latent tri-plane extrapolation. *ACM Transactions on Graphics (ToG)*, 43(4):1–17, 2024b.

Jinbo Xing, Long Mai, Cusuh Ham, Jiahui Huang, Aniruddha Mahapatra, Chi-Wing Fu, Tien-Tsin Wong, and Feng Liu. Motioncanvas: Cinematic shot design with controllable image-to-video generation. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pp. 1–11, 2025.

Jiawei Xu, Zexin Fan, Jian Yang, and Jin Xie. Grid4d: 4d decomposed hash encoding for high-fidelity dynamic gaussian splatting. *Advances in Neural Information Processing Systems*, 37: 123787–123811, 2024.

Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. *arXiv preprint arXiv:2310.10642*, 2023.

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, 2024.

Hong-Xing Yu, Haoyi Duan, Junhwa Hur, Kyle Sargent, Michael Rubinstein, William T Freeman, Forrester Cole, Deqing Sun, Noah Snavely, Jiajun Wu, et al. Wonderjourney: Going from anywhere to everywhere. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6658–6667, 2024.

Hong-Xing Yu, Haoyi Duan, Charles Herrmann, William T Freeman, and Jiajun Wu. Wonderworld: Interactive 3d scene generation from a single image. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5916–5926, 2025.

Bohan Zeng, Ling Yang, Siyu Li, Jiaming Liu, Zixiang Zhang, Juanxi Tian, Kaixin Zhu, Yongzhen Guo, Fu-Yun Wang, Minkai Xu, et al. Trans4d: Realistic geometry-aware transition for compositional text-to-4d synthesis. *arXiv preprint arXiv:2410.07155*, 2024.

Yufeng Zheng, Xueting Li, Koki Nagano, Sifei Liu, Otmar Hilliges, and Shalini De Mello. A unified approach for text- and image-guided 4d scene generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7300–7309, 2024.

Mengqi Zhou, Yuxi Wang, Jun Hou, Shougao Zhang, Yiwei Li, Chuanchen Luo, Junran Peng, and Zhaoxiang Zhang. Scenex: Procedural controllable large-scale scene generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 10806–10814, 2025.

In this supplementary material, we show the following contents:

- Algorithms of FluIDWorld (A)
- Motion Consistency Metrics (B)
- Additional experiment results (C)

A ALGORITHMS

For clarity, we present the complete algorithmic formulation of FluIDWorld. Algorithm 1 describes the overall control loop for interactive 4D scene generation, including scene expansion, flow lifting, motion alignment, and bidirectional motion propagation.

Algorithm 1 FluIDWorld control loop

Input: Initial scene image I_0
Output: Generated 4D dynamic scenes $\mathcal{G}$
Runtime output: Real-time rendered image I_{rend}
User control: Rendering pose C_{rend}, generation pose C_{gen}, motion seeds $S = \{p_i, m_i, h_i\}$, optional text U

$C_{\text{rend}} \leftarrow \text{Identitymatrix}()$ ▷ Initialize rendering camera
$C_{\text{gen}} \leftarrow \text{Identitymatrix}()$ ▷ Initialize generation camera
$(L_{\text{fg}}, L_{\text{bg}}, L_{\text{sky}}) \leftarrow \text{LayerDecompose}(I_0)$
$\mathcal{G} \leftarrow \text{InitFLAGS}(L_{\text{fg}}, L_{\text{bg}}, L_{\text{sky}})$
$S_{\text{prev}} \leftarrow \emptyset$ ▷ No Previous scene flow yet
$F_\theta \leftarrow \text{InitMotionField}()$ ▷ Initialize hash-encoded 3D motion field
in parallel do
  Thread 1: Real-time rendering loop
  while true **do**
    $I_{\text{rend}} \leftarrow \text{Render}(C_{\text{rend}}, \mathcal{G}, F_\theta)$ ▷ Render Gaussians with motion field
    $C_{\text{rend}} \leftarrow \text{UpdateByUser}(C_{\text{rend}})$
  end while
end parallel
in parallel do
  Thread 2: Scene expansion and motion update
  $C_{\text{gen}} \leftarrow C_{\text{rend}}$
  $I_{\text{partial}} \leftarrow \text{RenderPartial}(C_{\text{gen}}, \mathcal{G})$
  $M_{\text{empty}} \leftarrow \text{FindEmptyPixels}(I_{\text{partial}})$
  $(I_{\text{new}}, D_{\text{new}}) \leftarrow \text{OutpaintAndDepth}(I_{\text{partial}}, M_{\text{empty}}, U)$
  $\mathcal{G} \leftarrow \text{UpdateFLAGS}(\mathcal{G}, I_{\text{new}}, D_{\text{new}}, M_{\text{empty}})$
  $F_{2D} \leftarrow \text{EulerianFlow}(I_{\text{new}}, S)$ ▷ 2D single-image motion prediction
  $S_{\text{cur}} \leftarrow \text{Lift2Dto3D}(F_{2D}, D_{\text{new}})$ ▷ Unproject to 3D scene flow
  if $S_{\text{prev}} = \emptyset$ **then**
    $(R, s) \leftarrow \text{IdentityTransform}()$ ▷ First update: no alignment needed
  else
    $(R, s) \leftarrow \text{KabschAlign}(S_{\text{cur}}, S_{\text{prev}})$
    $(R, s) \leftarrow \text{RefineAlignment}(R, s)$ ▷ Lightweight optimization for residual correction
  end if
  $S_{\text{cur}} \leftarrow \text{ApplyTransform}(S_{\text{cur}}, R, s)$ ▷ Apply aligned rotation/scale to all new flow
  $S_{\text{prev}} \leftarrow S_{\text{prev}} \cup S_{\text{cur}}$ ▷ Expand scene flow
  $F_\theta \leftarrow \text{TrainHashMotionField}(F_\theta, S_{\text{prev}})$
  for all Gaussian g in $\mathcal{G}$ **do**
    if g is dynamic **then**
      $g^f \leftarrow g^f + \psi \odot F_\theta(g^f)$ ▷ (Eq. 9)
      $g^b \leftarrow g^b - \psi \odot F_\theta(g^b)$ ▷ (Eq. 10)
    end if
  end for
  $\mathcal{G} \leftarrow \text{MergeForwardBackward}(\mathcal{G})$ ▷ Compose final dynamic Gaussian set (Eq. 11)
end parallel

B Motion Consistency Metrics

In the main paper, we evaluate motion consistency using two metrics: Mean Cosine Alignment (MCA) and Flow Magnitude Variance (FMV). Both metrics quantify different aspects of spatiotemporal coherence in 3D scene flow. Below, we provide their detailed definitions and the exact formulations used in our implementation.

B.1 Mean Cosine Alignment (MCA)

Given a set of 3D scene flow vectors $\{\mathbf{s}_i\}_{i=1}^{N}$ and their corresponding 3D positions, we compute the K-nearest neighbors $\mathcal{N}(i)$ for each point based on Euclidean distance. To measure the directional coherence of the scene flow, we first normalize flow vectors:

$$\hat{\mathbf{s}}_i = \frac{\mathbf{s}_i}{\|\mathbf{s}_i\|}.$$

MCA is defined as the average cosine similarity between each flow direction and those of its spatial neighbors:

$$\text{MCA} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{K}\sum_{j\in\mathcal{N}(i)}\hat{\mathbf{s}}_i\cdot\hat{\mathbf{s}}_j\right).$$

A higher MCA value indicates stronger angular consistency across the 3D motion field, reflecting globally coherent motion propagation.

B.2 Flow Magnitude Variance (FMV)

Let $m_i = \|\mathbf{s}_i\|$ denote the magnitude of the scene flow at point i. Using the same neighborhood structure $\mathcal{N}(i)$, FMV measures the local smoothness of motion magnitude as:

$$\text{FMV} = \frac{1}{N}\sum_{i=1}^{N}\left(\frac{1}{K}\sum_{j\in\mathcal{N}(i)}(m_i - m_j)^2\right).$$

Lower FMV indicates that the flow magnitudes vary smoothly across neighboring points, which corresponds to stable motion strength without local discontinuities.

C Additional Experiment Results

Additional Fluid-like Motion Results. We include an extra qualitative result (Fig. 6) that presents additional motion examples beyond water, covering fluid-like phenomena such as clouds, smoke, and fire. This figure demonstrates that the proposed framework generalizes to a broader class of dynamic motions and is not limited to a single type of fluid behavior.

Effect of Depth Model Modification. We additionally present a qualitative analysis (Fig. 7) that examines the effect of modifying the depth estimation model from Marigold to MoGeV2. As shown in the figure, using the updated depth model reduces rendering artifacts observed in bird's-eye view visualizations, leading to improved geometric consistency.

Failure Case on Rigid Objects. Finally, we present a failure case (Fig. 8) in which motion is applied to a rigid object (a car). As illustrated in the figure, the proposed method struggles to preserve rigid-body consistency, resulting in visually implausible deformations. This failure case highlights a key limitation of our current formulation, which models dynamics as continuous velocity fields and is therefore better suited for fluid-like or deformable motions than strictly rigid objects.

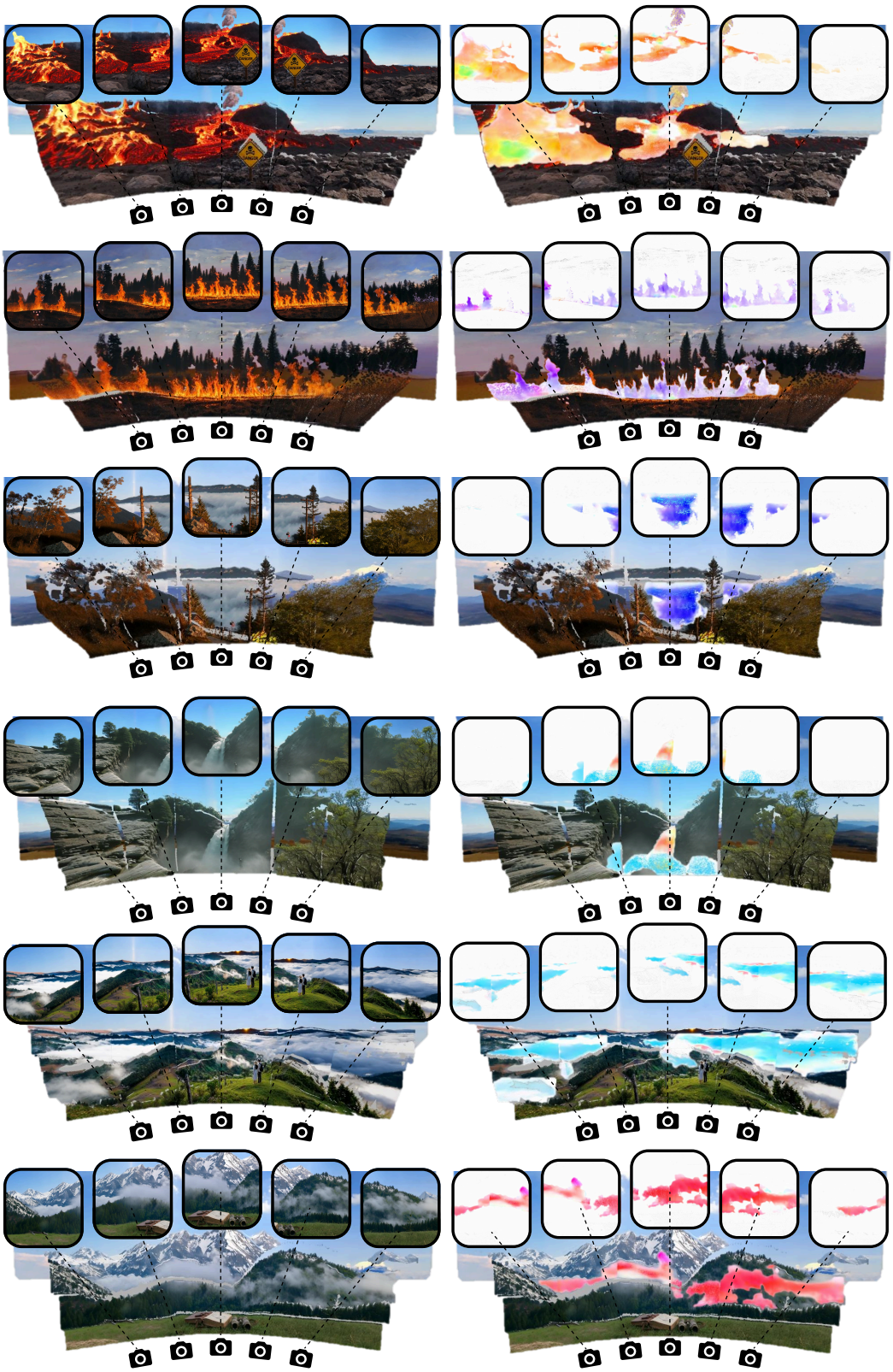

Figure 6: Additional fluid-like dynamics, including mixed scenes.

Figure 7: **Depth model ablation.** Replacing the depth estimation model from Marigold to MoGeV2 reduces rendering artifacts and improves geometric consistency, particularly in bird's-eye view visualizations.

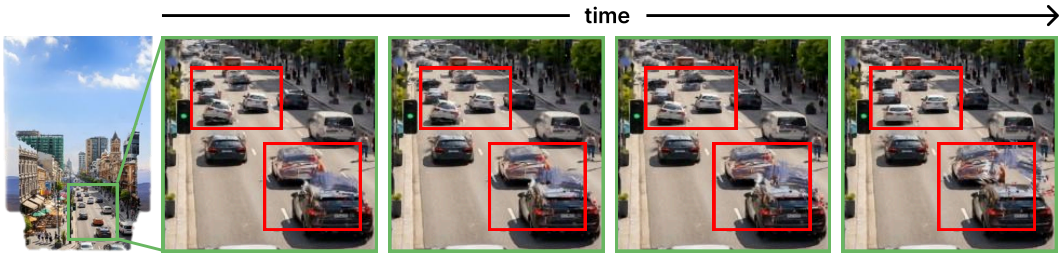

Figure 8: **Failure case.** Applying motion to a rigid object (a car) leads to visually implausible deformations, as the proposed method models dynamics as continuous velocity fields and does not explicitly enforce rigid-body constraints.