A Simple and Effective Zero-Shot Framework for Dynamic 3D World Modeling

Yuelei Li
Caltech

Zhengyang Lin
National University of Singapore

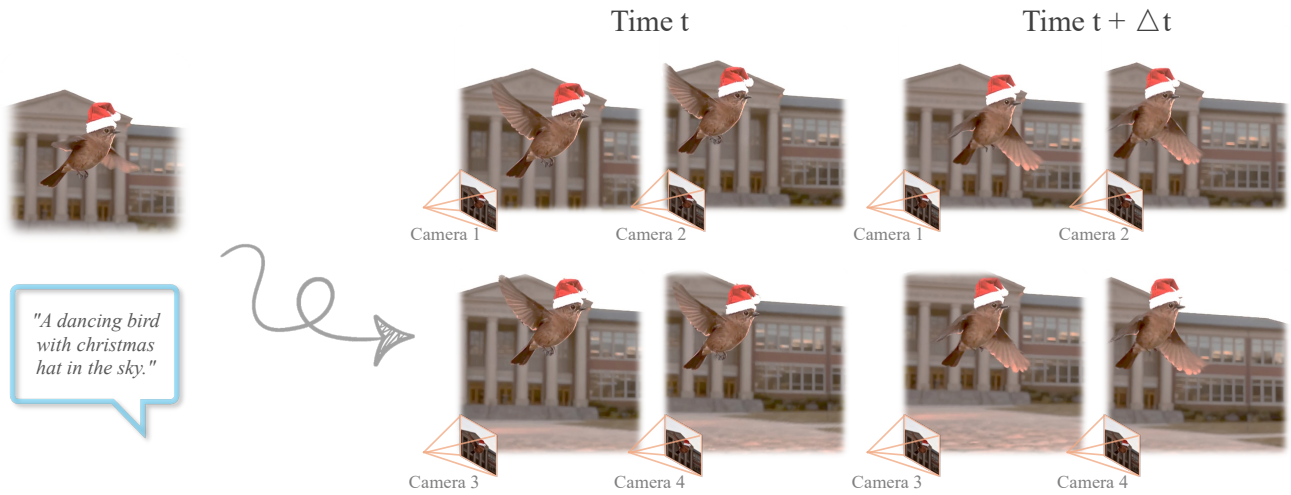

Figure 1. Our framework enables zero-shot generation of diverse dynamic 4D scenes from a single image and text prompt. By leveraging monocular video generation and enforcing geometric stability through view-plane constrained Gaussian splatting, our method produces spatially and temporally consistent 4D representations with clean geometry, controllable dynamics, and high-fidelity novel-view renderings.

Abstract

We present a novel framework for dynamic 3D scene generation conditioned on image and text. Existing methods on dynamic 3D scene generation either require a physics simulator to solve the scene dynamics based on the dynamic objects' materials that are not generalizable to complicated real-world scenes, or require computationally expensive large-scale training. In contrast, we present a zero-shot, simulation-free pipeline that takes advantage of powerful pretrained video generation models to generalize effectively to complex real-world scenes. Our method first uses a pretrained image and text to video model to synthesize a monocular video, and subsequently reconstructs the dynamic 3D scene using 4D Gaussian Splatting. However, directly applying vanilla 4D Gaussian Splatting to monocular videos often leads to artifacts due to incomplete geometry and view inconsistency. To address these problems, we introduce 1) a geometry-based 4D Gaussian initialization to enhance structural accuracy, 2) an inpainting layering strategy to resolve disocclusion hole artifacts that emerge under novel-view rendering, and 3) a view-plane–constrained splatting to avoid floating Gaussian points. The resulting 4D Gaussians enable controllable rendering from different viewing angles.

1. Introduction

In recent years, substantial progress has been made in image [4, 5, 8, 32], video [10, 30], and 3D generation [37]. This has directly motivated recent interest in generative world modeling, which has broad applications in areas such as video games, embodied AI, and AR/VR. Despite the progress, generating dynamic 3D scenes from a single image or text prompt remains highly challenging, as it requires maintaining both temporal coherence and geometric consistency. Furthermore, there are no existing large-scale datasets of dynamic 3D scenes as they are extremely hard to

collect, which means it is infeasible to use brute-force training to endow models with robust generative capabilities.

Most existing works on 4D generation focus on dynamic objects [1] that neglect the background and scene context. Compared to object-level generation, scene-level 4D generation is considerably more challenging due to the presence of complex geometries, diverse motions, and interactions between objects and the environment. One line of research uses physics simulators [16, 17] such as PBD (Position Based Dynamics) and MDM (Material Point Method) to simulate the dynamic part of the scene. However, these approaches struggle to generalize to complex real-world motions and materials, and are typically restricted to clean, synthetic environments. Another direction directly generates large-scale spatial- and temporal-variant data for training purposes [38], but this is computationally expensive and its performance is heavily dependent on the quality of the generated data. Recent work has also explored zero-shot 4D generation [18], which first synthesizes multi-view videos and then performs 4D reconstruction from them. However, despite various engineering efforts, this method highly relies on the performance of multi-view video generation, which usually struggles to achieve precise geometric alignment for complex scenes, resulting in blurry or distorted geometry.

To address these challenges, we propose a framework for zero-shot dynamic 3D generation as shown in Fig. 1. We turn to use monocular video generation, which provides inherently stronger spatial consistency within a single clip compared to multiple generated videos from different views. Nevertheless, performing 4D reconstruction from monocular videos remains highly challenging due to the absence of multi-view supervision. Applying standard 4D gaussian reconstruction [34] or its existing variants directly leads to severe floating-point artifacts when the camera deviates even slightly from the training trajectory, as well as empty regions around dynamic objects, i.e., disocclusion artifacts.

To address these issues, we introduce a view-plane constrained Gaussian splatting (VC-GS) that restricts the movement of Gaussian positions along the depth (z) axis relative to the training camera during optimization. This constraint is motivated by the observation that most floating-point artifacts occur along the depth direction, which is hard to remove via optimization under a monocular setting. Although the movement of Gaussian points along the depth direction is constrained, our experiment shows VC-GS is still capable of modeling dynamic objects moving along the depth direction. Furthermore, to address the issue of disocclusion artifacts around dynamic objects, we introduce a disentangled design that separates the static and dynamic layers. Specifically, we employ an offline model RoMo [6] to disentangle the static background and dynamic objects of the video, with VGGT [31] to estimate the camera poses and depth maps of both the layers respectively. With the geometric information, the background video is further inpainted using [39] to fill the disocclusion regions in a view-consistent manner.

Overall, our contributions are threefold:

- We introduce a zero-shot dynamic 3D scene generation framework from a single image, without requiring multi-view supervision or physical modeling.
- We propose a view-plane constrained Gaussian splatting (VC-GS) that effectively suppresses floating artifacts and improves geometric stability under monocular settings.
- We develop a disentangled static–dynamic layers method to handle disocclusion artifacts, yielding significantly cleaner and more complete reconstructions of dynamic scenes.

2. Related Work

2.1. 3D/4D World Generation

Recent work in 3D/4D world generation has prioritized speed and efficiency, largely shifting from implicit volumetric fields [20] to explicit Gaussian Splatting [13] variants. For large-scale static environments, methods like WonderJourney [35] pioneered perpetual scene generation using LLMs for narrative coherence and structure, while WonderWorld [36] addressed interactivity by introducing the FLAGS (Fast LAyered Gaussian Surfels) representation and Guided Depth Diffusion to achieve fast, seamless scene composition in approximately 9.5 seconds per scene. In the dynamic 4D domain, the movement is toward removing optimization overhead in reconstruction: L4GM [24] introduced the first large reconstruction model capable of predicting 4D Gaussian Splattings [33] in a single feed-forward pass, achieving near-instantaneous reconstruction in approximately 3 seconds. Complementarily, Instant4D [19] focuses on efficient monocular reconstruction, achieving a 30$\times$ speed-up via a Grid Pruning Strategy and a streamlined, isotropic 4DGS formulation. In terms of 4D generation, Free4D [18] approaches generative synthesis using a tuning-free distillation framework that leverages pre-trained 2D diffusion models. DimensionX [29] addresses single-image 4D scene creation via a decoupled video diffusion framework finetuned from pre-trained video diffusion models. PhysGen [17] achieves rigid physics-plausible videos via a training-free pipeline that uses zero-shot visual foundation models. WonderPlay [16] further supports 4D scene generation conditioned on physical actions by leveraging physics simulation and a video diffusion model that refines renderings from the physics simulator.

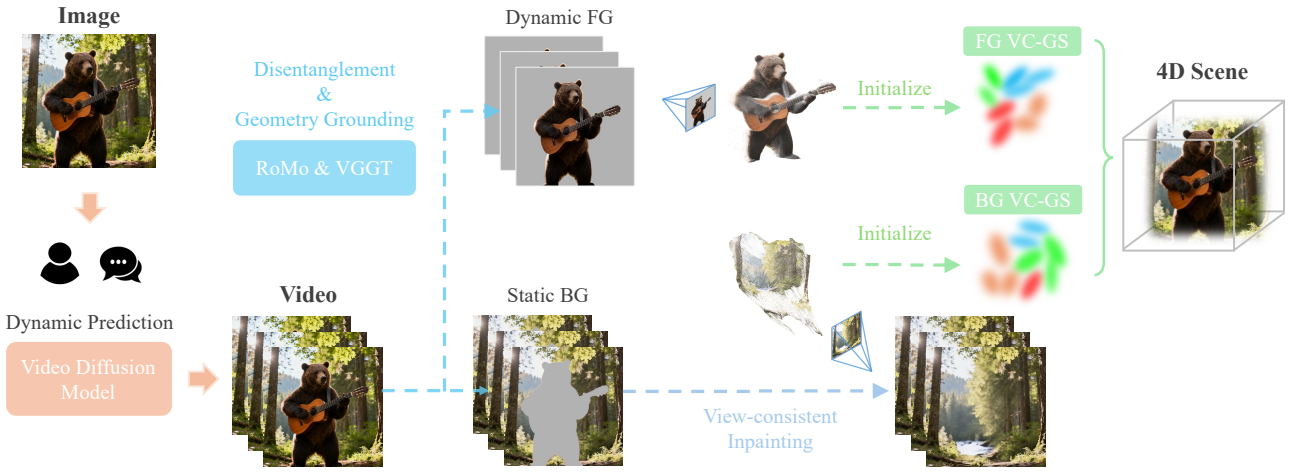

Figure 2. With a single image as input, we prompt a video diffusion model to predict a dynamic video. RoMo [6] and VGGT [31] are further used to decompose the video into static and dynamic layers with geometry parameters like camera poses and depth maps. The static background images with holes are filled via a view-consistent inpainting. Finally, a view-plane constraint Gaussian splatting is well initialized and is further optimized to represent the foreground and background which are combined to represent the full 4D scene.

2.2. Video Generation Models

Video generation has advanced rapidly in recent years, with modern approaches mostly adopting the paradigms of denoising diffusion. Video diffusion has been explored in both pixel space [9, 28] and latent space [2, 3], with architectures evolving from early Space–Time U-Nets [2, 10] to more recent DiT-based designs [7, 22]. Significant industrial investment has driven the development of large video diffusion models, leading to several multi-billion parameter models, including open-source models such as Wan [30] and Hunyuan [15], and closed-source models such as Sora [21], Movie-Gen [23], and Seaweed [27]. These models demonstrate impressive capabilities in synthesizing high-fidelity, temporally coherent videos that exhibit visually plausible physics and motion dynamics.

3. Method

Problem Formulation. Given a single image that specifies the content of the scene and a text prompt that describes how the scene should change over time, our goal is to generate an interactable 4D scene representation as shown in Fig. 2. For example, when provided with an image of a garden filled with flowers and the prompt "let the wind breeze through the garden", our framework synthesizes 4DGS representation that captures how the flowers sway in response to the breeze. The resulting 4DGS can then be rendered from arbitrary camera trajectories, enabling immersive and controllable visualization of the generated dynamic scene.

3.1. Image and Text to Video.

Given user-provided images and text descriptions, we employ a large pretrained video generation model [30] to generate a 5-second video. This process is performed in a zero-shot manner, requiring no architectural modification or additional fine-tuning. We denote the generated video frames as $I = \{I_t\}_{t=1}^{N}$ where N is the total frame number.

3.2. Dynamic & Static Disentanglement.

Based on the video generated from the first stage, we then use 4DGS to reconstruct the scene, which is represented as $G = \{G_t\}_{t=1}^{N}$. At each timestamp t, the scene is decomposed into dynamic and static components, denoted as $G_t = D_t \cup S_t$, where D_t and S_t correspond to the dynamic and static regions of the scene, respectively.

To obtain the dynamic regions, we employ a motion segmentation model [6] to estimate a set of binary masks $M = \{M_t\}_{t=1}^{N}$, where each $M_t = 1$ indicates pixels classified as dynamic. Consequently, the dynamic region at timestamp t can be expressed as $D_t = I_t \cap M_t$. However, a straightforward formulation of the static region as $D_t = I_t \cap (1 - M_t)$ will lead to artifacts such as holes and distortions around the boundaries of dynamic objects, primarily due to the insufficient multiview coverage in a monocular 4D reconstruction setting.

Consistent Inpaiting To address this problem, we propose to inpaint the static background regions that are occluded by dynamic foreground objects at each frame. Although our setting assumes a dynamic foreground and a static background (which holds for nearly all the scenarios), the same method can be applied in the reverse case, i.e., a dynamic background with a static foreground, by inpainting

the dynamic background regions. Naively inpainting each frame independently using a 2D inpainting model will introduce inconsistencies across frames. To address this, we input both the masks $M = \{M_t\}_{t=1}^N$ and the video frames $I = \{I_t\}_{t=1}^N$ into an object-removal video model [39] (removing dynamic objects is equivalent to inpainting their occluded background regions) to recover the complete static regions $S = \{S_t\}_{t=1}^N$. This effectively resolves inter-frame inconsistencies.

3.3. Monocular 4D Reconstruction.

This sub-section defines our **View-plane Constrained Gaussian Splatting** for effective *Monocular 4D Reconstruction*. Notably, each scene $\mathcal{E}$ is decomposed into a static part and a dynamic part as described above. To perform monocular 4D reconstruction, we use VGGT to estimate the camera poses and depth maps of the video.

Motion Modeling. Each part contains a set of 4D Gaussians, each parameterized by a 4D location $\boldsymbol{\mu} = (\mu_x, \mu_y, \mu_z, \mu_t)^\top \in \mathbb{R}^4$, a diagonal scale vector $\mathbf{s} = (s_{xyz},\ s_t)^\top$, a scalar opacity α, and a rotation matrix $R \in \mathbb{R}^{4\times4}$. Unlike prior 4DGS [34] work that relies on high-order Spherical Harmonics function, we model the Gaussian's appearance with RGB value as in [36].

Similar to [16, 36], we adopt a surfel-like Gaussian representation with a covariance matrix as below:

$$\boldsymbol{\Sigma} = \mathbf{Q}\mathrm{diag}\left(s_{\mathrm{x}}^2, s_{\mathrm{y}}^2, \epsilon^2, s_{\mathrm{t}}^2\right)\mathbf{Q}^{\mathrm{T}}, \tag{1}$$

where $\epsilon \ll \min(s_{\mathrm{x}}, s_{\mathrm{y}})$ is a tiny number. Notice that this form can be seen as a variant of 4DGS, where every Gaussian kernel's z-axis is shrunk to a tiny number, and it removes view-dependent colors.

We condition a multivariate 4D Gaussian primitive towards a 3D Gaussian primitive at timestamp t during the rendering time [34], which can be formulated as follows:

$$\boldsymbol{\mu}_{xyz|t} = \boldsymbol{\mu}_{1:3} + \boldsymbol{\Sigma}_{1:3,4}\,\boldsymbol{\Sigma}_{4,4}^{-1}\,(t - \mu_t), \tag{2}$$

$$\boldsymbol{\Sigma}_{xyz|t} = \boldsymbol{\Sigma}_{1:3,1:3} - \boldsymbol{\Sigma}_{1:3,4}\,\boldsymbol{\Sigma}_{4,4}^{-1}\,\boldsymbol{\Sigma}_{4,1:3}. \tag{3}$$

This conditioned form encodes continuous motion without explicit trajectory storage.

View-plane Constrained Gaussian Splatting. With the generated monocular video, we propose View-plane Constraint Gaussian Splatting (VC-GS) to effectively represent and reconstruct the dynamic 3D scene. Reconstructing a 4D scene from a single moving camera is an under-constrained problem, without guaranteed geometric triangulation. Thus, for monocular reconstruction, regular 4DGS are only able to overfit the training views while struggling to recover the dynamic scene geometry. As a result, there will be "cheating floaters", which become exposed when rendering from a new viewpoint as shown in Fig. 3. Notably, our initialization mechanism provides a good initial geometry of 4DGS, but we empirically found that the geometry will degrade clearly by the offset term during optimization, i.e., $\{\Delta x, \Delta y, \Delta z\} = \boldsymbol{\Sigma}_{1:3,4}\,\boldsymbol{\Sigma}_{4,4}^{-1}\,(t - \mu_t)$ in Eq. 2. For this case, we may consider naively setting all Gaussian offsets $\{\Delta x, \Delta y, \Delta z\}$ to zero, only letting the covariance term (in Eq. 3) be time-dependent. However, we find that this strategy will lead to "needle" artifacts during rendering, as only covariance can be optimized to model object motion.

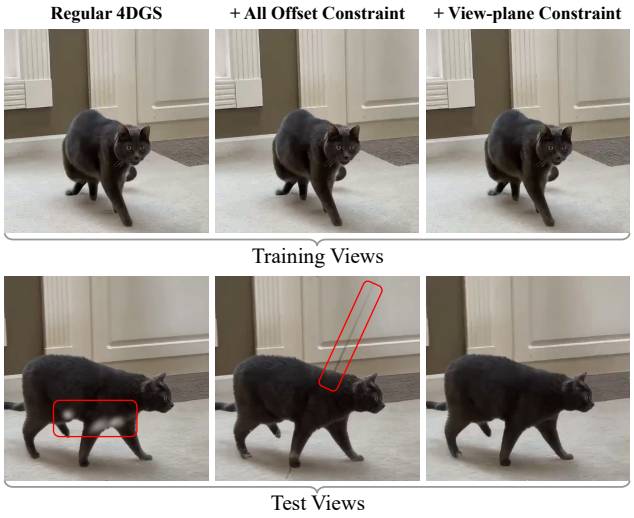

Figure 3. Results of applying different constraints to Gaussian optimization. Left is regular 4D Gaussian Splatting, which is able to fit training views but presents clear floater artifacts for novel view rendering. Middle is zero-setting the offset of dynamic Gaussian positions, which struggles to overfit training views and presents needle artifacts. Left is our view-plane constraint Gaussian Splatting, which both fit training views and present high-quality rendering for novel views.

On the other hand, we empirically find that the floater artifacts are caused by the movement of Gaussian positions along the depth direction. To this end, we propose to constrain the Gaussian offset on the **view plane**, with limited flexibility along the depth direction as shown in Fig. 4. Here we denote the Gaussian offset in camera coordinate as $\Delta x_c, \Delta y_c, \Delta z_c$, the camera rotation and transition as R and t. The relation of offsets in world coordinate and camera coordinate can be written as:

$$\Delta x_c,\ \Delta y_c,\ \Delta z_c\ =\ R(\Delta x,\ \Delta y,\ \Delta z)\ +\ t. \tag{4}$$

Then the constraint offset in camera coordinate can be written as:

$$\Delta x_c',\ \Delta y_c',\ \Delta z_c'\ =\ \Delta x_c,\ \Delta y_c,\ \varepsilon * \Delta z_c, \tag{5}$$

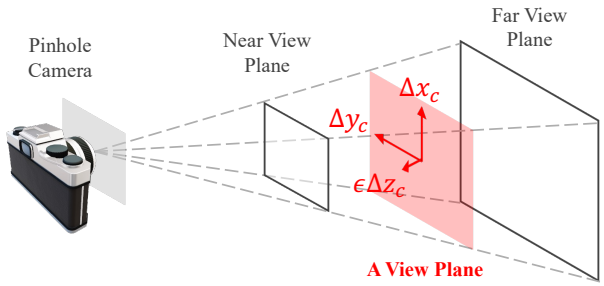

Figure 4. Illustration of the view-plane constraint Gaussian splatting (VC-GS). The Gaussian offset is mainly constrained to the **view plane** as denoted by Δx_c and Δy_c, with limited flexibility along the depth direction as denoted by $\varepsilon \Delta z_c$, where ε is a small value to constrain the offset Δz_c.

where ε is a small value to constrain the offset along z direction. We set it as 0.05 empirically. So the constraint offset in world coordinate is derived as:

$$\Delta x', \Delta y', \Delta z' = R^\top([\Delta x'_c,\ \Delta y'_c,\ \Delta z'_c] - t), \quad (6)$$

which serves as the final offset for the Gaussian positions during optimization.

This method gives flexibility to the 4DGS to fit moving objects but also constrains the undesired offset learning, which will lead to geometry degradation. Notably, we also show that this method also allows to model the motions along depth direction, although the movement of Gaussian along the direction is constrained. Please see experiment section 4 for more details.

Initialization. Optimizing dynamic 3D scene representations from monocular videos requires dealing with severe geometry uncertainty. We thus design a careful initialization scheme for dynamic scene representation, so that the optimization serves as a step to refine the Gaussians and is not required to learn the geometry from scratch.

The first design is the pixel-aligned geometry initialization. Specifically, given the estimated depth, camera poses, and disentangled static & dynamic images, we unproject the pixels to the 3D space to initialize the VC-GS positions. Therefore, the color of a VC-GS can be initialized as the RGB values of the pixel.

For the orientations and scales, we extend the initialization methods in WonderWorld [36] to 4D setting. Specifically, the orientation is initialized according to the per-pixel normal estimated by Marigold Normal [12]. The scale is initialized based on the estimated normal and Nyquist sampling theorem, aiming to minimize rendering aliasing caused by very small scales and avoid slowing down the optimization caused by overly big Gaussians.

Notably, this pixel-aligned initialization relies on the training frame, which actually provides initialization information for a camera pose at a certain timestep, instead of across all timesteps. On the other hand, we have disentangled the static and dynamic regions of training frames. Thus, for the static region, we initialize the VC-GS across different timesteps with the same position, color, orientations, and scales. For the dynamic region, we only initialize the VC-GS at the corresponding timestep.

Optimization. Our optimization of the layers goes from static background to dynamic foreground. We first optimize the background with the photometric loss against the inpainted static scene. Then, we optimize the foreground layer $\mathcal{L}_{\text{dynamic}}$ on top of both the frozen static layer $\mathcal{L}_{\text{static}}$, against the original scene images. We optimize for the opacity, orientation, and scales, but not for colors and 3D positions of VC-GS (time-dependent locations can be optimized with optimizable orientation). Our training video has 30 frames, which is optimized using Adam [14]. There is pruning to remove redundant Gaussians points initialized from camera unprojection, without incorporating the densification mechanism in 3DGS [13].

During rendering, we view the scene $\mathcal{E}$ as a union of static and dynamic layers, i.e.,

$$\mathcal{E} = \mathcal{L}_{\text{dynamic}} \cup \mathcal{L}_{\text{static}} = \{\mathbf{u}_i, \mathbf{q}_i, \mathbf{s}_i, o_i, \mathbf{c}_i\}_{i=1}^{N_{\text{dyanmic}}+N_{\text{static}}}, \quad (7)$$

where $N_{\text{fg}}/N_{\text{bg}}$ denotes the number of Gaussians. VC-GS generally follows the structure of 4DGS, where the time-dependent Gaussian motion along the view plane is constrained to a small value, removing floaters during novel view rendering. Thus, we can utilize the same differentiable rendering pipeline (i.e., 4D-to-3D, 3D-to-2D projection and alpha blending) as 4DGS [34] for rendering VC-GS.

In monocular 4DGS primitive modeling, static background primitives can vanish once they leave the camera frustum unless they are explicitly distinguished from moving objects. We apply the motion mask obtained from RoMo [6] as described in Sec. 3.2. To make our 4D primitive aware of the underlying motion in the monocular dynamic scene. Considering opacity and Equation 7, we can find that in the case of our isotropic Gaussians, the temporal scaling would be the only term in the covariance that affects the Gaussian attribute related to time.

4. Experiments

Baselines. We compare our framework with current state-of-the-art methods 4D scene generation methods. This includes DimensionX [29], which can generate 4D scenes using controllable video diffusion, and Free4D [18], a tuning-free framework designed for 4D scene generation. Both of them take an image and a text prompt as input. For fair comparison, for DimensionX, during evaluation we input

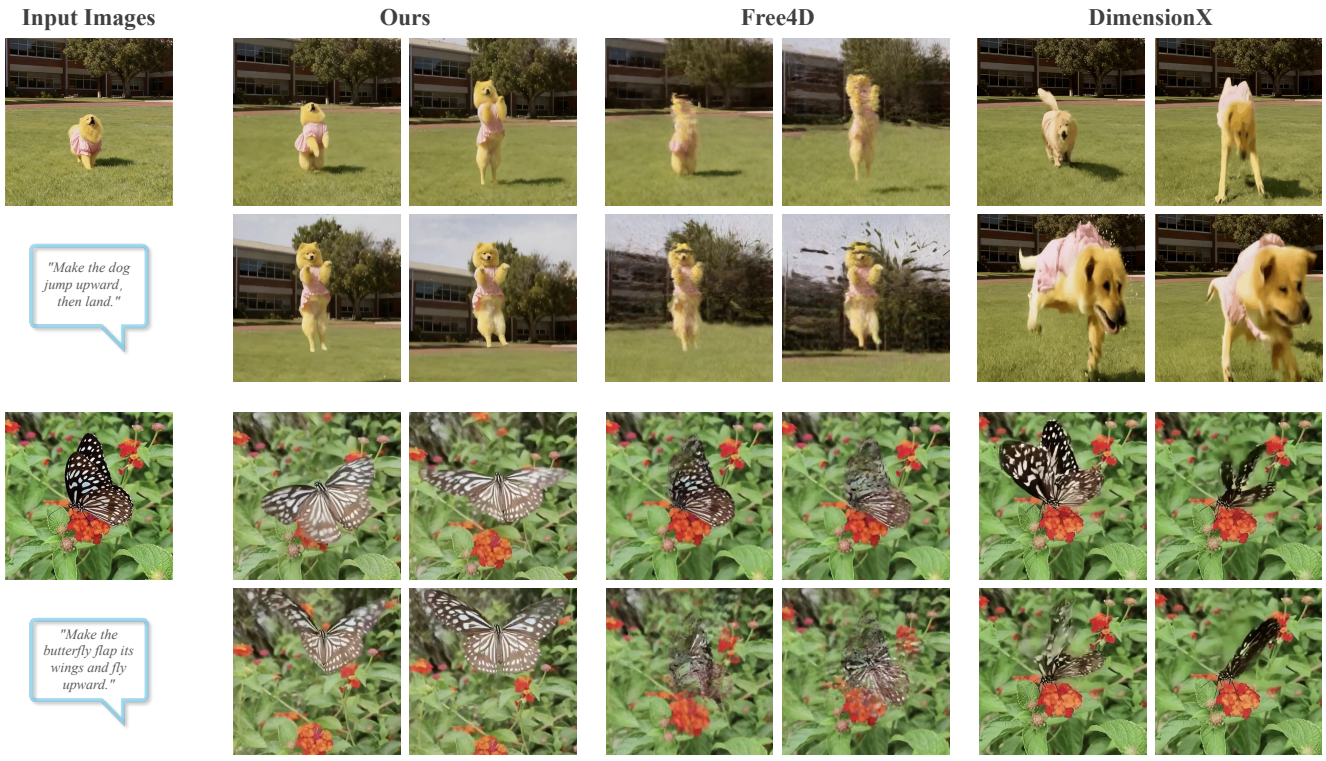

Figure 5. Qualitative comparisons of image-to-4D. We present the results using the same image as input.

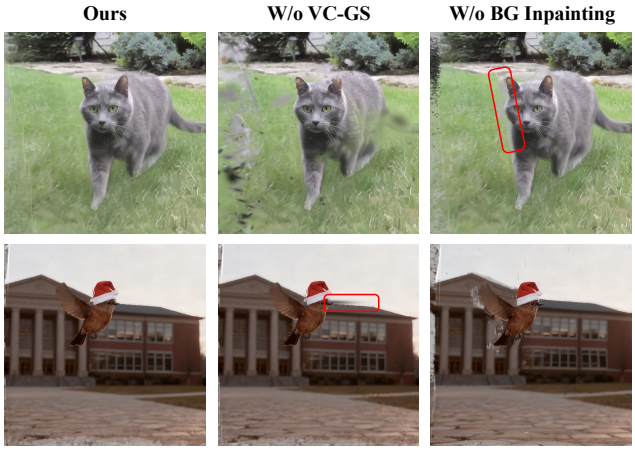

Figure 6. Ablation study of our designed components. The images are rendered in the same novel view. **W/o VC-GS** denotes the regular implementation of 4DGS without view-plane constraint, **W/o BG Inpainting** denotes remove static & dynamic disentanglement and inpainting mechanism.

the same image and text prompt as ours; for Free4D, it also involves a standalone image and text conditioned video generation first step as ours, so we directly input our generated video to Free4D instead of generating a different one.

Notably, our framework (including VC-GS) is designed for scene generation, and therefore we do not compare VC-GS with monocular reconstruction methods, following the evaluation protocol adopted in WonderWorld (FLAGS) [36].

Evaluation Metrics. To assess the quality of videos rendered from our 4D representations, we report standard VBench [11] metrics, including Consistency (averaged over subject and background consistency), Aesthetic Score, and Dynamic Degree. Following common practice, we curate 15 samples for evaluation, consisting of both real photographs and several AI-generated images from [5]. We additionally use text prompts to control camera trajectories, such as move up, move left, move right, and follow the subject, to generate the test videos. In our experiments, we observe that VBench metrics can be highly unreliable for evaluating video generation. Nevertheless, we report them for completeness and comparability. We strongly encourage the readers to inspect our video results in the supplementary materials, as human perception remains the most trustworthy criterion for judging visual quality.

Implementation Details. For video diffusion models, we use Wan2.5 [30] to predict a 5s video with fps 30 from a single image with text prompts. We then sample 30 frames from it to reconstruct the 4D scene. For static & dynamic disentanglement and geometry grounding, we use RoMo [6]

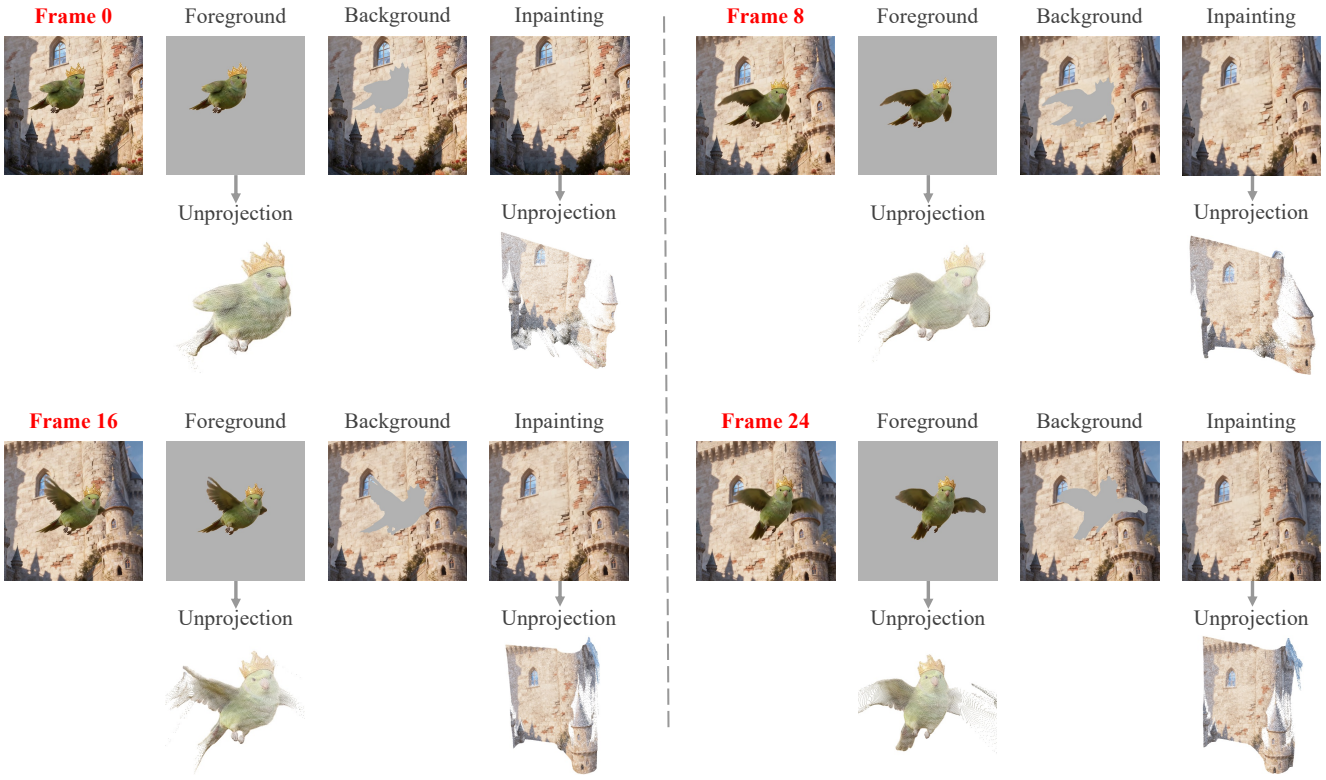

Figure 7. Illustration of our results of static & dynamic disentanglement, background inpainting, and points unprojection. The disentanglement is accurate, and the inpainting results are consistent across different views, e.g., frames 1, 8, 16, 24.

	Consistency ↑	Aesthetic ↑	Dynamic ↑
Ours	0.94	0.63	0.50
DimensionX [29]	0.94	0.60	0.46
Free4D [18]	0.90	0.57	0.40

Table 1. Quantitative Results on VBench [11]

and VGGT [31], respectively. Notably, we estimate the depth maps and camera poses of the generated video to initialize and train the VC-GS. For the inpainted region of background, we obtain the depth by estimating the depth of the inpainted background and aligning it with the generated video depth. For 4D scene representation, we train VC-GS for 5000 iterations in total.

Quantitative Evaluation. We report our results on VBench [11] metrics (see Tab. 1). Ours achieves the highest Aesthetic and Dynamic scores, and a Consistency score comparable to DimensionX, which requires training on large scale videos. Compared to DimensionX and Free4D that rely on multi-view video generation to optimize the scene, our monocular-based VC-GS reconstruction produces fewer artifacts and maintains stable geometry across viewpoints.

For DimensionX, we found it can't respond to the camera trajectory text prompt well – for dynamic scenes the generated camera motion remains very limited when larger movements are specified. Also, sometimes the generated scene gets blurry after a few timesteps. For Free4D, we found the common failure case results from inaccurate multiview videos, which leads to inaccurate camera pose estimation and thus blurry 4D reconstruction. In some cases, the colmap part [25, 26] in Free4D can't run successfully on the generated multiview videos, so we use VGGT instead [31] and export the camera poses to COLMAP format to continue with their pipeline.

Qualitative Evaluation. Figure 5 illustrates representative visual results. Ours produces clean geometry, sharper textures, and stable view renderings even for scenes with complex motions such as a butterfly flapping its wings. In contrast, DimensionX often suffers from geometry drift and spatial misalignment across time, while Free4D usually produce blurry surfaces and floating structures due to imperfect multi-view generation. Our view-plane constraint effectively suppresses floaters, and the layered representation prevents background tearing or hole artifacts.

Demonstration We also visualize our intermediate re-

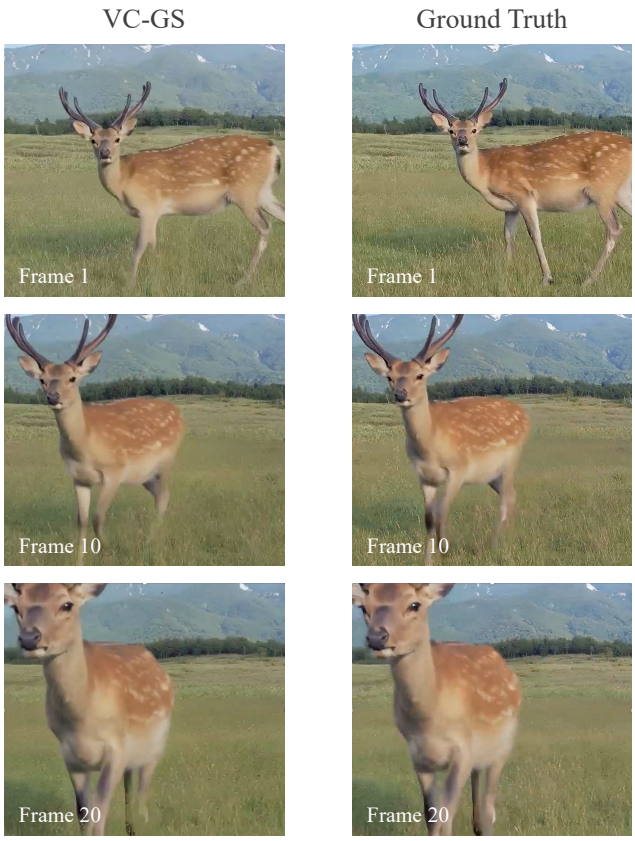

Figure 8. Experiments on a dynamic scene with an object moving towards the camera along the depth direction. Our VC-GS is also able to represent this type of dynamic scenario, although the Gaussian movement along the depth direction is constrained.

sults of foreground & background disentanglement and background inpainting as shown in Fig. 7. It demonstrates accurate disentanglement of static and dynamic parts, and the view-consistent and time-consistent inpainting results, which finally facilitate the subsequent optimization of 4D representation.

Additionally, given the concern that VC-GS may struggle with scenes where dynamic components move along the depth axis, we further evaluate our model on dynamic scenes of this type, as shown in Fig. 8. Although the Gaussian movement along the depth direction is constrained in VC-GS, the representation still handles such cases effectively, achieving high-quality results. This is enabled by the optimization of time-dependent opacity, orientation, and scale parameters, which compensate for the limited depth-wise movement.

Ablation Study. We conduct extensive ablations to analyze the impact of our key components as shown in Fig. 6. (1) *Without VC-GS (x,y,z all change)*: removing the constraint on depth movement leads to severe floaters and inconsistent geometry, significantly degrading Consistency. (3) *Without Background Inpainting*: the reconstruction suffers from disocclusion holes around dynamic regions when the render camera pose deviates from the training camera pose. These results confirm that each component in our pipeline, including VC-GS and layered disentanglement & inpainting, is crucial for high-quality zero-shot 4D reconstruction.

5. Limitations and Future Work

Our framework inherits several limitations from its monocular video generation backbone. When the generated video exhibits temporal drift, lighting inconsistencies, or implausible motion, these errors propagate into the 4D reconstruction. As our layered static–dynamic disentanglement depends on RoMo, motion blur, fast-moving objects, or cluttered regions may degrade mask quality. Our geometry grounding relies on VGGT, which is typically designed for static scenarios, resulting in degraded depth and camera pose estimation. Future improvements may come from stronger video and geometric priors, such as multi-frame consistent video diffusion models, and more robust 4D foundation models for depth/normal estimation. Moreover, tighter coupling between video diffusion and 4D reconstruction, e.g., adding geometric constraints during video synthesis, may further reduce artifacts and improve overall scene fidelity.

6. Conclusion

In this work, we introduced a zero-shot framework for dynamic 3D scene generation from a single image and text prompt, addressing the fundamental limitations of existing approaches that rely on physics simulators, large-scale 4D datasets, or multi-view video synthesis. By leveraging monocular video generation for strong spatial consistency and coupling it with a geometry-grounded initialization, a disentangled static–dynamic reconstruction pipeline with consistent inpainting, and our proposed View-plane Constrained Gaussian Splatting (VC-GS) for stable monocular 4D reconstruction, our framework effectively suppresses floating artifacts, resolves disocclusion holes, and produces coherent, high-quality 4D scenes. Extensive experiments demonstrate that our method achieves superior geometric stability, temporal consistency, and visual fidelity across diverse scenarios, offering a practical step toward scalable generative world modeling. We hope our framework inspires future exploration into richer, more controllable, and more interactive 4D modeling.

References

[1] 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*, pages 7996–8006, 2024. 2

[2] 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. 3

[3] Andreas Blattmann, Robin Rombach, Huan Ling, Tim Dockhorn, Seung Wook Kim, Sanja Fidler, and Karsten Kreis. Align your latents: High-resolution video synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 22563–22575, 2023. 3

[4] Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025. 1

[5] Yu Gao, Lixue Gong, Qiushan Guo, Xiaoxia Hou, Zhichao Lai, Fanshi Li, Liang Li, Xiaochen Lian, Chao Liao, Liyang Liu, Wei Liu, Yichun Shi, Shiqi Sun, Yu Tian, Zhi Tian, Peng Wang, Rui Wang, Xuanda Wang, Xun Wang, Ye Wang, Guofeng Wu, Jie Wu, Xin Xia, Xuefeng Xiao, Zhonghua Zhai, Xinyu Zhang, Qi Zhang, Yuwei Zhang, Shijia Zhao, Jianchao Yang, and Weilin Huang. Seedream 3.0 technical report, 2025. 1, 6

[6] Lily Goli, Sara Sabour, Mark Matthews, Marcus A Brubaker, Dmitry Lagun, Alec Jacobson, David J Fleet, Saurabh Saxena, and Andrea Tagliasacchi. Romo: Robust motion segmentation improves structure from motion. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6155–6164, 2025. 2, 3, 5, 6

[7] Agrim Gupta, Lijun Yu, Kihyuk Sohn, Xiuye Gu, Meera Hahn, Fei-Fei Li, Irfan Essa, Lu Jiang, and José Lezama. Photorealistic video generation with diffusion models. In *European Conference on Computer Vision*, pages 393–411. Springer, 2024. 3

[8] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851, 2020. 1

[9] Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 3

[10] 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. 1, 3

[11] Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21807–21818, 2024. 6, 7

[12] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9492–9502, 2024. 5

[13] 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. 2, 5

[14] Diederik P Kingma. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 5

[15] Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xiong, Xin Li, Bo Wu, Jianwei Zhang, et al. Hunyuanvideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024. 3

[16] 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*, 2025. 2, 4

[17] Shaowei Liu, Zhongzheng Ren, Saurabh Gupta, and Shenlong Wang. Physgen: Rigid-body physics-grounded image-to-video generation. In *European Conference on Computer Vision*, pages 360–378. Springer, 2024. 2

[18] Tianqi Liu, Zihao Huang, Zhaoxi Chen, Guangcong Wang, Shoukang Hu, Liao Shen, Huiqiang Sun, Zhiguo Cao, Wei Li, and Ziwei Liu. Free4d: Tuning-free 4d scene generation with spatial-temporal consistency. *arXiv preprint arXiv:2503.20785*, 2025. 2, 5, 7

[19] Zhanpeng Luo, Haoxi Ran, and Li Lu. Instant4d: 4d gaussian splatting in minutes. *arXiv preprint arXiv:2510.01119*, 2025. 2

[20] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM*, 65(1):99–106, 2021. 2

[21] OpenAI. Sora. https://openai.com/sora, 2024. 3

[22] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205, 2023. 3

[23] Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024. 3

[24] Jiawei Ren, Cheng Xie, Ashkan Mirzaei, Karsten Kreis, Ziwei Liu, Antonio Torralba, Sanja Fidler, Seung Wook Kim, Huan Ling, et al. L4gm: Large 4d gaussian reconstruction model. *Advances in Neural Information Processing Systems*, 37:56828–56858, 2024. 2

[25] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 7

[26] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *European Conference on Computer Vision (ECCV)*, 2016. 7

[27] Team Seawead, Ceyuan Yang, Zhijie Lin, Yang Zhao, Shanchuan Lin, Zhibei Ma, Haoyuan Guo, Hao Chen, Lu Qi, Sen Wang, et al. Seaweed-7b: Cost-effective training of video generation foundation model. *arXiv preprint arXiv:2504.08685*, 2025. 3

[28] Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. Make-a-video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022. 3

[29] Wenqiang Sun, Shuo Chen, Fangfu Liu, Zilong Chen, Yueqi Duan, Jun Zhang, and Yikai Wang. Dimensionx: Create any 3d and 4d scenes from a single image with controllable video diffusion. *arXiv preprint arXiv:2411.04928*, 2024. 2, 5, 7

[30] 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. 1, 3, 6

[31] 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*, pages 5294–5306, 2025. 2, 3, 7

[32] Zhen Wang, Yuelei Li, Jia Wan, and Nuno Vasconcelos. Diffusion-based data augmentation for object counting problems, 2024. 1

[33] 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*, pages 20310–20320, 2024. 2

[34] Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time photorealistic dynamic scene representation and rendering with 4d gaussian splatting. In *The Twelfth International Conference on Learning Representations*, 2024. 2, 4, 5

[35] 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*, pages 6658–6667, 2024. 2

[36] 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*, pages 5916–5926, 2025. 2, 4, 5, 6

[37] Jiahui Zhang, Yuelei Li, Anpei Chen, Muyu Xu, Kunhao Liu, Jianyuan Wang, Xiao-Xiao Long, Hanxue Liang, Zexiang Xu, Hao Su, Christian Theobalt, Christian Rupprecht, Andrea Vedaldi, Kaichen Zhou, Paul Pu Liang, Shijian Lu, and Fangneng Zhan. Advances in feed-forward 3d reconstruction and view synthesis: A survey, 2025. 1

[38] Yuyang Zhao, Chung-Ching Lin, Kevin Lin, Zhiwen Yan, Linjie Li, Zhengyuan Yang, Jianfeng Wang, Gim Hee Lee, and Lijuan Wang. Genxd: Generating any 3d and 4d scenes, 2024. 2

[39] Bojia Zi, Weixuan Peng, Xianbiao Qi, Jianan Wang, Shihao Zhao, Rong Xiao, and Kam-Fai Wong. Minimax-remover: Taming bad noise helps video object removal. *arXiv preprint arXiv:2505.24873*, 2025. 2, 4