ORBITNVS: HARNESSING VIDEO DIFFUSION PRIORS FOR NOVEL VIEW SYNTHESIS

Anonymous authors
Paper under double-blind review

ABSTRACT

Novel View Synthesis (NVS) aims to generate unseen views of a 3D object given a limited number of known views. Existing methods often struggle to synthesize plausible views for unobserved regions, particularly under single-view input, and still face challenges in maintaining geometry- and appearance-consistency. To address these issues, we propose OrbitNVS, which reformulates NVS as an orbit video generation task. Through tailored model design and training strategies, we adapt a pre-trained video generation model to the NVS task, leveraging its rich visual priors to achieve high-quality view synthesis. Specifically, we incorporate camera adapters into the video model to enable accurate camera control. To enhance two key properties of 3D objects, geometry and appearance, we design a normal map generation branch and use normal map features to guide the synthesis of the target views via attention mechanism, thereby improving geometric consistency. Moreover, we apply a pixel-space supervision to alleviate blurry appearance caused by spatial compression in the latent space. Extensive experiments show that OrbitNVS significantly outperforms previous methods on the GSO and OmniObject3D benchmarks, especially in the challenging single-view setting (e.g. , +2.9 dB and +2.4 dB PSNR).

1 INTRODUCTION

Novel View Synthesis (NVS) aims to generate photorealistic unseen views of a object using only a limited set of input views (Shi et al., 2023; Liu et al., 2023a). This capability promises to revolutionize a spectrum of applications, powering the creation of 3D assets for film and gaming (Zhao et al.,

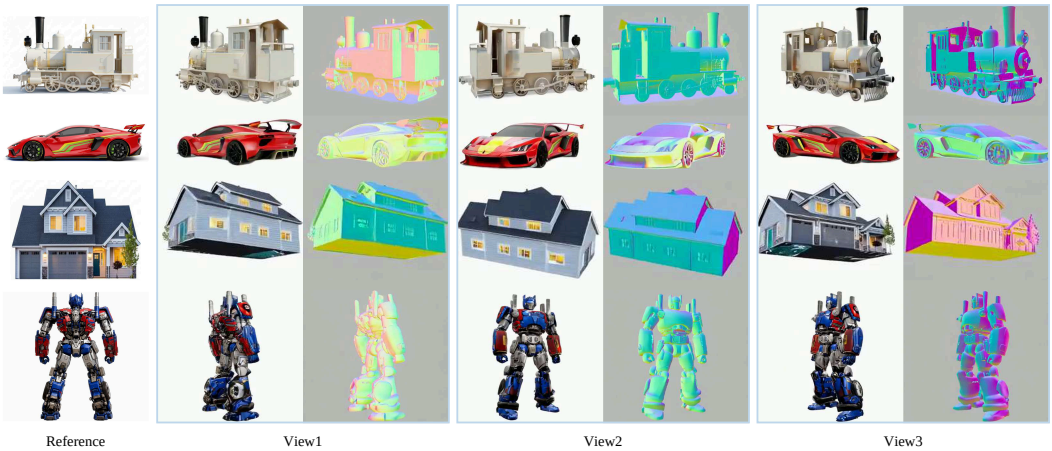

Figure 1: OrbitNVS leverages the rich visual prior of video generation models to achieve highly consistent NVS with precise camera control. Its strong prior enables impressive reasoning capabilities in the unseen views. For instance, it can infer the detailed front view of a robot from only its back, or deduce the presence of windows on the opposite side of a house from a single frontal image.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

2025), enabling immersive AR/VR experiences (Qu et al., 2024), and advancing robotic perception (Bar et al., 2025).

The NVS task is inherently an ill-posed problem. Its core challenge lies in "imagining" the appearance of unseen areas based on a limited set of seen views. Humans address this challenge by leveraging visual commonsense knowledge, which is acquired through extensive observation of the physical world, to infer occluded regions (Liu et al., 2025b). For instance, when observing the front of an object, humans can infer its back by relying on priors of similar objects, without explicitly constructing a precise 3D model in the mind. This observation indicates that the NVS is fundamentally a problem of visual commonsense reasoning rather than a geometric reconstruction. Given that video data is rich in visual commonsense and is significantly cheaper to acquire than 3D data, which requires expensive expert modeling or specialized scanning equipment, video data and the generative models pre-trained on it present a promising pathway for advancing NVS.

Several existing works have attempted to learn such visual commonsense from large-scale multi-view image data. For instance, methods like Zero-1-to-3 (Liu et al., 2023a; Shi et al., 2023), EscherNet (Kong et al., 2024), CAT3D (Gao et al., 2024) and SEVA (Zhou et al., 2025) fine-tune 2D image generative models (Rombach et al., 2022) on multi-view datasets to apply them to the NVS task. However, since these 2D models inherently lack cross-view perception, their priors provide limited help in inferring occluded regions and do not contribute to 3D consistency. While CAT4D (Wu et al., 2025) and AC3D (Bahmani et al., 2025) utilize video models and incorporate camera conditioning for NVS, they are primarily designed for general video generation, where objects may move or transform over time. Consequently, they achieve suboptimal performance on the object-centric NVS task we focus on. In contrast, SV3D (Voleti et al., 2024) pioneers a different paradigm by fine-tuning a pre-trained video generation model (Blattmann et al., 2023) to leverage its rich visual commonsense for novel view synthesis of static objects. Nevertheless, due to its relatively simple camera conditioning mechanism and training strategy, the method exhibits limitations in camera control accuracy, geometric consistency, and appearance clarity, ultimately failing to fully realize the potential of this paradigm.

To address these issues, we propose OrbitNVS, a framework that reframes NVS as an orbital video generation task. Our approach incorporates a leading video generation model and introduces tailored camera conditioning and training strategies to unleash its rich visual commonsense for NVS. Specifically, drawing inspiration from post-training strategies for video models (He et al., 2024; Bahmani et al., 2025), we integrate multiple camera adapters into the model to condition the generation process on the camera trajectory. These adapters modulate the feature maps in a pixel-wise manner based on the input camera pose, enabling effective camera control. Since normal maps have demonstrated their effectiveness in helping models understand 3D geometric structures (Long et al., 2024; Li et al., 2024), we introduce an auxiliary normal map generation branch and facilitate feature interaction between the two branches via self-attention mechanisms. This allows the normal map to guide the RGB synthesis, resulting in more coherent geometric structures. Furthermore, to improve fine appearance details, we incorporate a pixel-space loss that maps latent representations back to the pixel domain for direct supervision, thereby mitigating detail loss or blurring caused by resolution compression in latent spaces.

In summary, our contributions are as follows:

1) We propose OrbitNVS, a novel framework that incorporates a pre-trained video generation model into NVS via carefully designed camera conditioning and training mechanisms, effectively leveraging its rich visual commonsense for high-quality view synthesis.

2) To enhance both geometric and appearance fidelity, we introduce two dedicated techniques: a normal map generation branch to reinforce 3D coherence, and a pixel-space training loss to recover fine-grained texture details.

3) Extensive experiments demonstrate that OrbitNVS achieves state-of-the-art performance in NVS, particularly in the challenging single-image setting. For instance, under a level camera orbit, it improves PSNR by +2.9 dB and +2.4 dB on the GSO and Omniobject3D benchmarks, respectively.

2 Related Work

Novel View Synthesis. Existing NVS methods can be categorized into three main paradigms based on their underlying 3D representations. The first category relies on **Explicit 3D Representations**. Methods like those of Charatan et al. (2024); Held et al. (2025) directly optimize representations such as NeRF (Mildenhall et al., 2020) or 3DGS (Kerbl et al., 2023) via differentiable rendering from dense views. While effective with dense inputs, they struggle in sparse-view settings. Generalization efforts involve depth cues (Yu et al., 2021; Deng et al., 2022; Szymanowicz et al., 2024), regularization (Niemeyer et al., 2022; Yang et al., 2023; Jain et al., 2021), or SDS loss (Sargent et al., 2024; Poole et al., 2023; Roessle et al., 2024). However, their fundamental limitation remains: optimizing a separate 3D representation per scene impedes the transfer of multi-view knowledge across objects. The second category involves **Intermediate 3D Caches**. Methods such as those of Huang et al. (2025); Yu et al. (2024); Ren et al. (2025) first reconstruct a 3D cache (Wang et al., 2024a; 2025) (e.g., a coarse point cloud) from reference inputs, then render it from the target viewpoint to condition a diffusion model. While this incorporates 3D consistency, the two-stage process is computationally costly and prone to error accumulation, where reconstruction inaccuracies degrade the final output. Moreover, such caches lack the inherent visual common sense needed for reasoning about unobserved areas. The third category, **Implicit 3D Priors**, learns 3D common sense from multi-view image data, embedding 3D knowledge into generative model parameters. Early methods achieved single novel view synthesis (Jin et al., 2024; Liu et al., 2023a), while recent advances enable multi-view generation (Kong et al., 2024; Liu et al., 2023c; Ye et al., 2023; Zheng & Vedaldi, 2024). These approaches typically fine-tune 2D generative models, but their lack of cross-frame reasoning limits 3D consistency. Newer works like ViVid-1-to-3 (Kwak et al., 2024) and SV3D (Voleti et al., 2024) use video diffusion models for stronger spatial reasoning. Nevertheless, with relatively simple architectures and training strategies, these methods still underperform and have yet to fully realize this paradigm's potential.

Video Generation. Benefiting from the development of diffusion generation paradigms and transformer model architectures, video generation models have achieved impressive fidelity (Peebles & Xie, 2023; Blattmann et al., 2023). However, their application to NVS presents challenges. Firstly, although these models offer diverse conditioning options—e.g., text (Blattmann et al., 2023; Wang et al., 2024b; Chen et al., 2024; Zheng et al., 2024), image (Wan et al., 2025; Yang et al., 2025; Jiang et al., 2024; 2025; Liu et al., 2025a; Fei et al., 2025), speech (Wan et al., 2025), or customization (Zhou et al., 2024; Zhang et al., 2015; Wang et al., 2024c)—NVS requires more flexible and precise camera viewpoint control. Specifically, some existing methods (Wan et al., 2025; Kwak et al., 2024) only support relatively simple camera motion patterns, such as panning or zooming, which are inadequate for the arbitrary novel views or complex trajectories required in NVS. Although works such as CAT4D (Wu et al., 2025) and AC3D (Bahmani et al., 2025) introduce dedicated camera adapters or specialized architectures to achieve more flexible camera control, they are primarily tailored for general video generation. In these scenarios, objects often undergo temporal motion or transformations over time, leading to suboptimal performance on the static object NVS task that we focus on. Secondly, conventional video generation prioritizes temporal continuity and visual plausibility (Zheng et al., 2024; Wan et al., 2025; Wang et al., 2024b; Zhao et al., 2025), often overlooking inherent 3D geometric consistency, paramount for NVS (Voleti et al., 2024). This is evident in orbital videos, where models struggle to maintain consistent object geometry and scene structure across changing viewpoints. Therefore, substantial adaptation is required to meet NVS's stringent demands for flexible viewpoint control and 3D consistency.

3 Method

In this section, we first describe our problem and define the notation (§3.1), then introduce the video generation model we employ (§3.2), and finally, detail the model design and training strategy of our proposed OrbitNVS (§3.3).

3.1 Problem Formulation

We formulate the NVS task as an orbital video generation problem: given a camera trajectory and a few known reference frames on it, the goal is to generate the complete orbital video. This can be

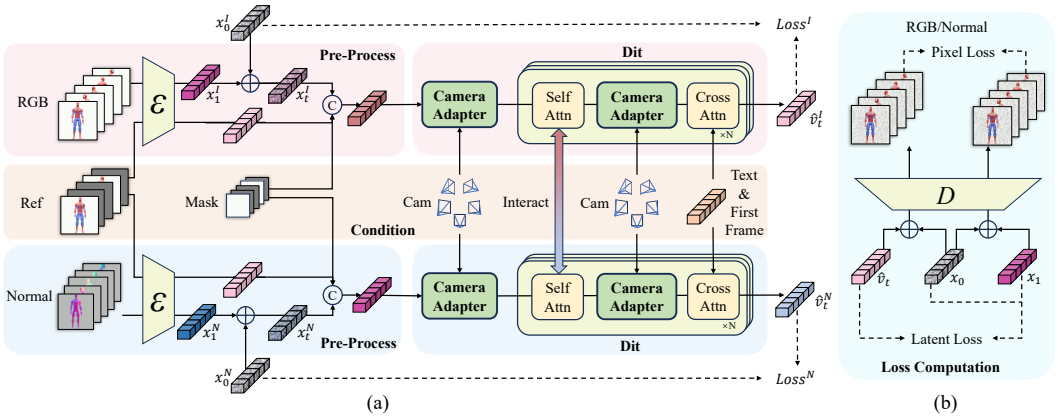

Figure 2: Overview of OrbitNVS. (a) Camera adapters condition a video diffusion backbone on the camera trajectory. A parameter-sharing normal branch interacts with the RGB branch through self-attention. (b) Training combines latent-space supervision with pixel-space supervision through the VAE decoder.

expressed as:

$$\mathcal{V}^{1:T} \sim \mathcal{P}\left(\mathcal{V}^{1:T} | \mathcal{F}^{R}, \mathcal{C}^{1:T}\right), \tag{1}$$

where $\mathcal{V}^{1:T}$ denotes the complete orbital video, $\mathcal{F}^{R}$ is the reference frame images, $\mathcal{C}^{1:T}$ is the camera trajectory, and T is the total number of frames, which we set to 61. While our formulation primarily assumes a continuous camera trajectory as input, the proposed method remains compatible with discrete viewpoints by interpolating them into a continuous trajectory.

From the perspective of controllable video generation, OrbitNVS is analogous to a video inpainting model. Specifically, it completes a masked video where only a few reference frames are provided, conditioned on the camera trajectory. In our setting, the first frame is always provided as a reference frame.

3.2 Foundation Model

We adopt Wan2.1-I2V-14B (Wan et al., 2025) as our foundation model, leveraging its video generation priors for unseen-view synthesis and cross-view consistency.

Framework. Following the latent diffusion paradigm (Rombach et al., 2022; Liang et al., 2024), a VAE (Kingma & Welling, 2013) encodes the video as $x_1 = \mathcal{E}(\mathcal{V}^{1:T})$. Rectified-flow training (Lipman et al., 2023; Liu et al., 2023b; Esser et al., 2024) interpolates between Gaussian noise $x_0 \sim \mathcal{N}(0, I)$ and x_1 at timestep $t \in [0, 1]$:

$$x_t = tx_1 + (1-t)x_0, \tag{2}$$

$$v_t = \frac{dx_t}{dt} = x_1 - x_0. \tag{3}$$

Here, x_t represents the noised latent, $x_0 \in \mathcal{N}(0, 1)$ is a random noise, $t \in [0, 1]$ represents the timestep, and v_t represents the ground truth velocity. The loss function can be formulated as the mean squared error between the model output and v_t at any point along the trajectory:

$$\mathcal{L}_{latent} := ||v_t - \hat{v}_\theta(x_t, t)||_2^2. \tag{4}$$

Here, $\hat{v}_\theta$ denotes the model's velocity prediction function.

Conditions. The initial frame, padded with zero frames, is fed into the VAE encoder to obtain the conditioning latent (Zhang et al., 2023; Blattmann et al., 2023; Yang et al., 2025). This latent and a binary conditioning mask are concatenated with the noised latent. Text embeddings and the initial frame's CLIP embedding (Radford et al., 2021) are injected through cross-attention in each DiT block (Peebles & Xie, 2023).

3.3 Model Design and Training

Despite its rich visual priors, the foundation model lacks explicit camera-pose conditioning and struggles to maintain geometric consistency and fine appearance details in orbital NVS. We therefore introduce a camera adapter for pose conditioning, together with normal-map generation and pixel-space post-training to improve geometric plausibility and appearance quality.

Camera Adapter. We represent camera poses using Plücker coordinates. Given an orbital video $\mathcal{V}^{1:T} \in \mathbb{R}^{3\times T\times H\times W}$ and camera poses $\mathcal{C}^{1:T}$, we compute a 6D Plücker coordinate for each pixel ray, yielding $\hat{\mathcal{C}}^{1:T} \in \mathbb{R}^{6\times T\times H\times W}$.

Following He et al. (2024); Wan et al. (2025), we insert camera adapters before the first DiT layer and between the self- and cross-attention blocks of each layer, as shown in Figure 2 (a). Each adapter predicts the shift and scale parameters for Adaptive Layer Normalization (AdaLN) (Peebles & Xie, 2023) from the Plücker tensor:

$$\mu, \sigma = \mathcal{A}(\hat{\mathcal{C}}^{1:T}), \tag{5}$$

$$\hat{z} = (1+\sigma)\cdot z + \mu, \tag{6}$$

where $\mathcal{A}$ denotes the camera adapter, and z and $\hat{z}$ are the original and camera-conditioned DiT features, respectively. The adapter consists of two convolutional layers with parallel residual connections and a zero-initialized output convolution, preserving the pretrained DiT feature distribution at initialization.

Normal Map Generation. Existing diffusion models directly align generated videos with ground-truth RGB observations, where geometry and texture are inherently entangled. Consequently, objects with similar geometry but different textures may appear distinct to the model, hindering geometric learning. To address this issue, we introduce a parameter-sharing normal generation branch, as shown in Figure 2 (a). It interacts with the RGB branch through self-attention and shares the same Plücker embeddings, enabling OrbitNVS to jointly generate pixel-aligned RGB and normal maps. Normal supervision provides texture-independent shape cues, while cross-branch interaction transfers these geometric cues to RGB generation, improving geometric consistency and geometric representation learning.

Pixel-Space Post-Training. Latent diffusion models are typically supervised in the compressed latent space, which improves efficiency but may compromise high-frequency details due to lossy VAE compression (Zhang et al., 2024). To improve reconstruction fidelity, we introduce Pixel-Space Post-Training, which decodes the latent representations and aligns their outputs through the VAE decoder, as shown in Figure 2 (b). By back-propagating the pixel-space loss through the VAE decoder, the DiT learns latent representations better aligned with the decoder:

$$\mathcal{L}_{pixel} := \left\| \mathcal{D}(tx_1 + (1-t)x_0) - \mathcal{D}(x_0 + t\hat{v}_\theta(x_t, t)) \right\|_2^2, \tag{7}$$

where $x_1 \in \mathcal{E}(\mathcal{V})$ is the encoded video latent, $x_0 \sim \mathcal{N}(0,1)$ is Gaussian noise, $t \in [0,1]$, $x_t = tx_1 + (1-t)x_0$, and $\hat{v}_\theta(x_t, t)$ is the predicted velocity.

Due to the computational cost of pixel-space supervision, we adopt a two-stage training strategy: the model is first trained with standard latent-space supervision and then post-trained with joint latent- and pixel-space supervision.

4 Experiments

4.1 Settings

In this section, we present our experimental setup in three aspects: the benchmarks and metrics (§4.1.1), baseline methods for comparison (§4.1.2), and implementation details of OrbitNVS (§4.1.3).

4.1.1 Benchmarks and Metrics.

Following SV3D (Voleti et al., 2024), we use Omniobject3D (Wu et al., 2023) and GSO (Downs et al., 2022), unseen during model training, as our test datasets. Omniobject3D comprises 6k 3D

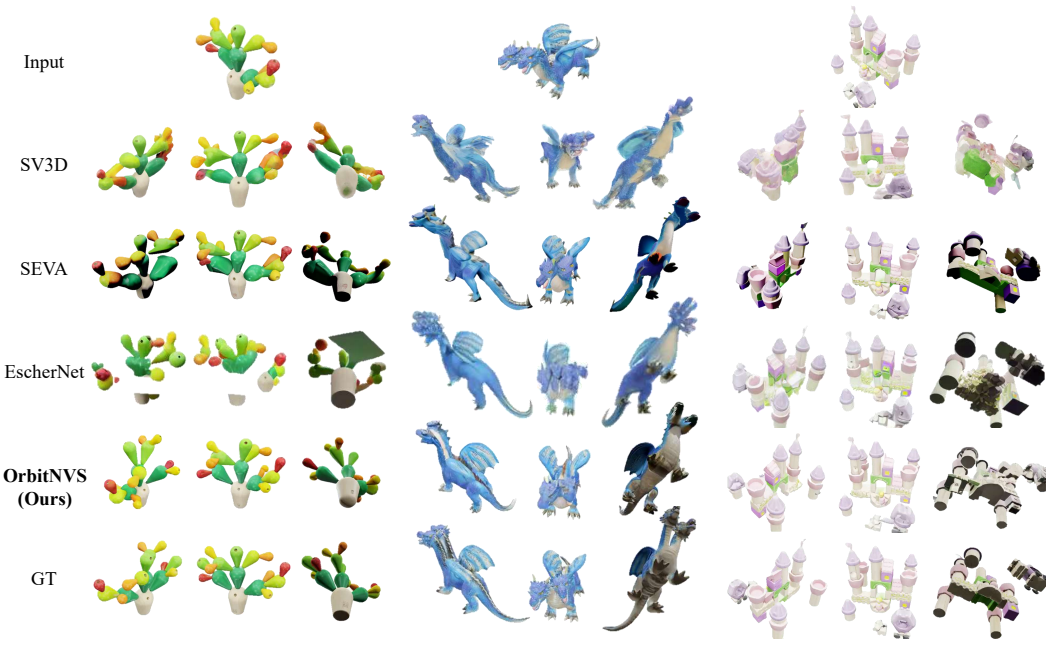

Figure 3: Qualitative comparison of NVS results generated by different methods under the single reference view setting.

assets, and GSO contains 1k 3D assets, both acquired by scanning real-world objects. To compare the NVS performance of different methods, we render the 3D assets into orbital videos. The initial rendering resolution is 1024 $\times$ 1024 with 61 frames, which are subsequently resized to the required resolution for different methods. The camera trajectory for our rendering involves the azimuth uniformly circling the object, while the elevation angle fluctuates sinusoidally, as expressed below:

$$a_i = \frac{2\pi \cdot i}{T_{val}}, \tag{8}$$

$$e_i = \mathcal{U}_{val} \cdot sin(f \cdot \frac{2\pi \cdot i}{T_{val}}). \tag{9}$$

Here, a_i and e_i represent the azimuth and elevation of the camera for the i-th frame, respectively. T_{val} denotes the number of frames, set to 61. f represents the frequency of the elevation's fluctuation, set to 3, and $\mathcal{U}_{val}$ indicates the amplitude of this fluctuation. We established three levels of difficulty by setting three different amplitudes for $\mathcal{U}_{val}$: 0°, 30°, and 60°. An amplitude of 0° signifies a level, circular camera orbit. Additionally, we ensure the camera always points towards the object's center, normalize the object's size, and set the camera's rotation radius to 4 to maintain a moderate object size in the video.

To evaluate the similarity between the generated videos and the real videos, we use image similarity metrics: Peak Signal-to-Noise Ratio (PSNR), Structural SIMilarity (SSIM), and Learned Perceptual Similarity (LPIPS) (Zhang et al., 2018).

4.1.2 BASELINES

We compare OrbitNVS with three state-of-the-art NVS baselines: SV3D (Voleti et al., 2024), SEVA (Zhou et al., 2025), and EscherNet (Kong et al., 2024). SV3D is most closely related to our setting, as it formulates NVS as orbital video generation. For fair comparison, we adapt the frame count and resolution to each baseline's input requirements. Specifically, our 61-frame videos are uniformly downsampled to 21 frames for SV3D, while SEVA and EscherNet use all 61 frames. The input resolutions are 576×576 for SV3D and SEVA, and 256×256 for EscherNet.

Table 1: Single-reference NVS results on GSO (top) and OmniObject3D (bottom) for camera trajectories with elevation oscillation amplitudes of 0°, 30°, and 60° (0° denotes a level orbit). *w/o normal* and *w/o pixel-loss* remove the normal map branch and pixel-space loss, respectively. Each entry reports PSNR↑, SSIM↑, and LPIPS↓; best results are in **bold**.

GSO			
Method	0°	30°	60°
SEVA	18.3 0.87 0.15	16.3 0.84 0.17	13.7 0.80 0.24
EscherNet	20.8 0.85 0.15	19.8 0.84 0.15	19.3 0.83 0.17
SV3D	18.9 0.88 0.18	17.5 0.86 0.22	17.6 **0.86** 0.22
OrbitNVS	**23.7 0.89 0.11**	**21.9 0.87 0.13**	**21.0 0.86 0.15**
w/o normal	22.3 **0.89** 0.12	20.8 **0.87** 0.14	19.9 **0.86** 0.16
w/o pixel-loss	22.8 **0.89 0.11**	21.2 **0.87 0.13**	20.0 0.85 0.16

OmniObject3D			
Method	0°	30°	60°
SEVA	16.7 0.83 0.19	16.1 0.82 0.19	15.9 0.82 0.19
EscherNet	18.6 0.82 0.19	18.2 0.81 0.19	18.1 0.81 0.20
SV3D	16.4 0.84 0.24	16.7 0.84 0.23	17.4 **0.85** 0.21
OrbitNVS	**21.0 0.87 0.14**	**20.4 0.86 0.14**	**19.9 0.85 0.16**
w/o normal	20.1 0.85 0.16	19.1 0.85 0.17	18.6 0.84 0.18
w/o pixel-loss	20.8 0.86 **0.14**	19.7 0.84 0.15	19.7 0.84 **0.16**

Table 2: NVS results on GSO (top) and OmniObject3D (bottom) under a level camera trajectory with varying numbers of reference views. Each entry reports PSNR↑, SSIM↑, and LPIPS↓; best results are in **bold**.

GSO			
Method	1 ref.	2 ref.	3 ref.
SV3D	18.9 0.88 0.18	-	-
SEVA	18.3 0.87 0.15	25.7 **0.92** 0.07	28.3 **0.94 0.04**
EscherNet	20.9 0.85 0.15	23.4 0.87 0.13	26.3 0.91 0.06
OrbitNVS	**23.7 0.89 0.11**	**26.5 0.92 0.06**	**28.8 0.94 0.04**

OmniObject3D			
Method	1 ref.	2 ref.	3 ref.
SV3D	16.4 0.84 0.24	-	-
SEVA	16.7 0.83 0.19	23.9 **0.90** 0.09	27.1 **0.92 0.05**
EscherNet	18.6 0.82 0.19	22.8 0.86 0.10	25.3 0.89 0.06
OrbitNVS	**21.0 0.87 0.14**	**24.4 0.90 0.08**	**27.3 0.92 0.05**

4.1.3 Implementation Details

We train on 320k assets from the Step1X-3D collection (Li et al., 2025), filtered from Objaverse-V1 (Deitke et al., 2023), using 61-frame RGB and normal-map sequences at 448×448 resolution. Qwen2.5-VL-7B (Bai et al., 2025) generates text conditions from training videos and, at test time, exclusively from the reference images. Data preparation, training schedules, and hyperparameters are provided in Appendix C.1.

4.2 Main Results

We evaluate OrbitNVS in terms of unseen-view synthesis, performance across camera trajectories, and the effect of additional reference views.

Reasoning on unseen views. OrbitNVS achieves the highest PSNR among the evaluated baselines across all single-reference settings on both benchmarks (Table 1). Under a level camera trajectory, it outperforms the strongest baseline, EscherNet, by 2.9 dB on GSO and 2.4 dB on OmniObject3D. Figure 3 further illustrates plausible unseen-view synthesis and improved geometric consistency, including recovery of the dragon's underside from an elevated reference view and more consistent structures in the plant and castle examples.

Performance across camera trajectories. OrbitNVS retains its advantage under larger elevation variations. On GSO, it exceeds EscherNet by 2.1 dB and 1.7 dB at oscillation amplitudes of 30° and 60°, respectively (Table 1). Increasing the amplitude from 0° to 60° reduces OrbitNVS's PSNR by 2.7 dB, compared with 4.6 dB for SEVA. These results demonstrate stronger synthesis performance under challenging camera trajectories, although larger viewpoint changes remain difficult.

Effect of additional reference views. Table 2 compares performance with uniformly spaced reference views. All multi-view methods improve as more observations become available, while OrbitNVS maintains the highest PSNR across the evaluated input counts on both datasets. With three references, OrbitNVS and SEVA reach 28.8 dB and 28.3 dB on GSO, respectively, with both attaining 0.94 SSIM. The narrower performance gap compared with single-reference synthesis suggests that OrbitNVS's advantage is most pronounced when observations are sparse and more content must be inferred.

4.3 Ablation

In this section, we conduct an ablation study on our two key designs: the normal map generation branch and the pixel-space post-training.

Impact of the normal map generation branch. Figure 4 demonstrates the novel view synthesis (NVS) results for a bamboo basket with complex geometry. As shown in Figure 4 (b), the model without the normal map generation branch fails to capture the intricate woven structure, resulting in generated novel views with missing geometric details and an overall poor shape. In contrast, with

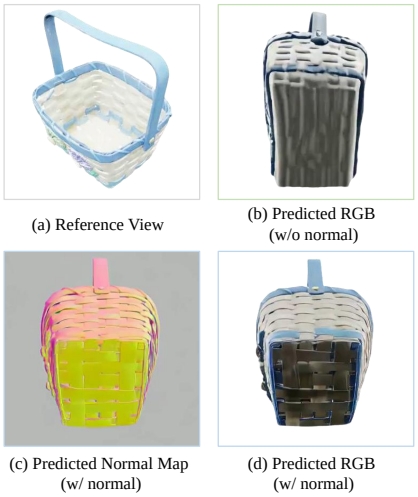

Figure 4: Ablation on the normal map generation branch. Results with the branch (c, d) show significantly clearer geometric details than those without it (b).

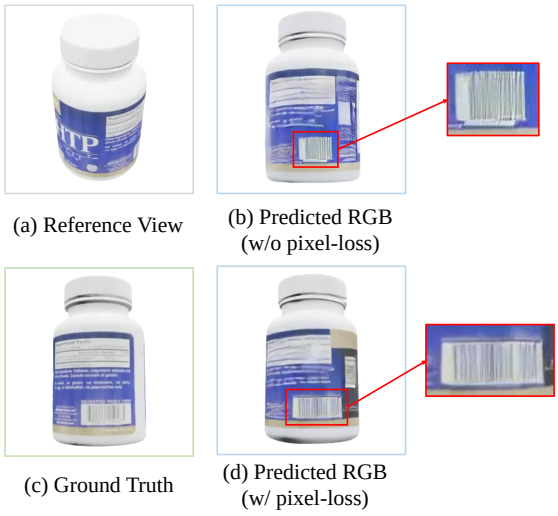

Figure 5: Ablation on the pixel-space post-training. The result with the pixel loss (d) shows superior plausibility in detailed textures compared to the result without it (b).

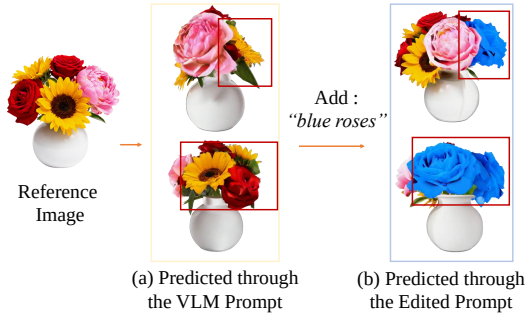

Figure 6: Controllable Novel View Synthesis is achieved by manipulating the text prompt. (a) Using a VLM to predict a prompt from the reference image as the condition for NVS. (b) Using a manually modified text prompt as the condition for NVS.

the introduction of this branch, OrbitNVS simultaneously generates both a normal map and an RGB image for the novel view that accurately reproduce the detailed geometry, as seen in Figure 4 (c) and (d). The results are highly realistic, closely resembling the actual bamboo weaving. This indicates that the normal map generation branch significantly enhances the geometric fidelity of the generated orbit. Quantitatively, the branch also leads to a substantial improvement in metrics. As presented in Table 1, it achieves a performance gain of over 1 dB in PSNR across most settings.

Impact of the pixel-space post-training. Figure 5 presents the NVS results for an object with fine-grained textures. As shown in Figure 5 (b), the model trained without the pixel-loss fails to preserve high-frequency details in areas like the barcode, resulting in blurred and irregular textures. It is important to note that this model has the same number of training steps as the model with pixel-loss, but uses only the latent loss. In contrast, the model with the pixel-loss generates a more plausible barcode, as depicted in Figure 5 (d). This demonstrates that pixel-space post-training effectively enhances the model's capability to generate high-frequency details, thereby improving visual clarity. Quantitatively, the data in Table 1 confirm that this component yields an improvement in metrics.

4.4 Controllable NVS

Conventional NVS methods rely entirely on the model's intrinsic imagination from the known view, leaving users with no control over the process. In contrast, OrbitNVS incorporates an additional text input, enabling more controllable generation through text conditions. For example, as shown in Figure 6 (a), if we use a caption generated by a VLM for the reference image as the text condition, the model generates a novel view with a red flower. However, if we manually edit the text prompt to include "blue roses," we can steer the model to generate a blue flower in the novel view, as shown in Figure 6 (b).

5 Conclusion

We present OrbitNVS, which reformulates the NVS task as an orbital video generation problem, leveraging the rich visual priors of pre-trained video generation models to achieve high-quality view synthesis. Our approach introduces three main adaptations: first, by incorporating camera adapters into the video model to achieve precise camera control; second, by designing a normal map generation branch and utilizing an attention mechanism to allow normal features to guide the synthesis of novel views, thereby enhancing geometric consistency; and third, by mapping the latent representations back to the pixel space for supervision, which improves the clarity and detail of the synthesized appearance. Extensive experiments demonstrate that our method significantly outperforms existing methods, particularly in its superior ability to reason about unseen views and its enhanced geometric and appearance consistency. We believe that leveraging the priors of video models to assist 3D vision tasks is a promising research avenue. OrbitNVS represents a solid step forward along this path. We will release our model to facilitate further research.

AI USE STATEMENT

We used generative AI primarily for language polishing and translation, with additional assistance in checking literature and references and checking manuscript formatting.

REFERENCES

Sherwin Bahmani, Ivan Skorokhodov, Guocheng Qian, Aliaksandr Siarohin, Willi Menapace, Andrea Tagliasacchi, David B Lindell, and Sergey Tulyakov. Ac3d: Analyzing and improving 3d camera control in video diffusion transformers. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 22875–22889, 2025.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, Jiabo Ye, Xi Zhang, Tianbao Xie, Zesen Cheng, Hang Zhang, Zhibo Yang, Haiyang Xu, and Junyang Lin. Qwen2.5-VL technical report. *arXiv preprint arXiv:2502.13923*, 2025.

Amir Bar, Gaoyue Zhou, Danny Tran, Trevor Darrell, and Yann LeCun. Navigation world models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 15791–15801, 2025.

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

David Charatan, Sizhe Lester Li, Andrea Tagliasacchi, and Vincent Sitzmann. pixelsplat: 3d gaussian splats from image pairs for scalable generalizable 3d reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 19457–19467, 2024.

Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. Videocrafter2: Overcoming data limitations for high-quality video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7310–7320, 2024.

Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 13142–13153, 2023.

Kangle Deng, Andrew Liu, Jun-Yan Zhu, and Deva Ramanan. Depth-supervised nerf: Fewer views and faster training for free. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 12882–12891, 2022.

Laura Downs, Anthony Francis, Nate Koenig, Brandon Kinman, Ryan Hickman, Krista Reymann, Thomas B McHugh, and Vincent Vanhoucke. Google scanned objects: A high-quality dataset of 3d scanned household items. In *2022 International Conference on Robotics and Automation (ICRA)*, pp. 2553–2560. IEEE, 2022.

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for high-resolution image synthesis. In *Forty-first international conference on machine learning*, 2024.

Zhengcong Fei, Debang Li, Di Qiu, Jiahua Wang, Yikun Dou, Rui Wang, Jingtao Xu, Mingyuan Fan, Guibin Chen, Yang Li, et al. Skyreels-a2: Compose anything in video diffusion transformers. *arXiv preprint arXiv:2504.02436*, 2025.

David F. Fouhey, Abhinav Gupta, and Martial Hebert. Data-driven 3d primitives for single image understanding. In *ICCV*, pp. 3392–3399, 2013.

Ruiqi Gao, Aleksander Holynski, Philipp Henzler, Arthur Brussee, Ricardo Martin-Brualla, Pratul Srinivasan, Jonathan T Barron, and Ben Poole. Cat3d: Create anything in 3d with multi-view diffusion models. *arXiv preprint arXiv:2405.10314*, 2024.

Hao He, Yinghao Xu, Yuwei Guo, Gordon Wetzstein, Bo Dai, Hongsheng Li, and Ceyuan Yang. Cameractrl: Enabling camera control for text-to-video generation. *arXiv preprint arXiv:2404.02101*, 2024.

Jan Held, Renaud Vandeghen, Adrien Deliege, Abdullah Hamdi, Silvio Giancola, Anthony Cioppa, Andrea Vedaldi, Bernard Ghanem, Andrea Tagliasacchi, and Marc Van Droogenbroeck. Triangle splatting for real-time radiance field rendering. *arXiv preprint arXiv:2505.19175*, 2025.

Tianyu Huang, Wangguandong Zheng, Tengfei Wang, Yuhao Liu, Zhenwei Wang, Junta Wu, Jie Jiang, Hui Li, Rynson WH Lau, Wangmeng Zuo, and Chunchao Guo. Voyager: Long-range and world-consistent video diffusion for explorable 3d scene generation. *arXiv preprint arXiv:2506.04225*, 2025.

Ajay Jain, Matthew Tancik, and Pieter Abbeel. Putting nerf on a diet: Semantically consistent few-shot view synthesis. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 5885–5894, 2021.

Yuming Jiang, Tianxing Wu, Shuai Yang, Chenyang Si, Dahua Lin, Yu Qiao, Chen Change Loy, and Ziwei Liu. Videobooth: Diffusion-based video generation with image prompts. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6689–6700, 2024.

Zeyinzi Jiang, Zhen Han, Chaojie Mao, Jingfeng Zhang, Yulin Pan, and Yu Liu. Vace: All-in-one video creation and editing. *arXiv preprint arXiv:2503.07598*, 2025.

Haian Jin, Hanwen Jiang, Hao Tan, Kai Zhang, Sai Bi, Tianyuan Zhang, Fujun Luan, Noah Snavely, and Zexiang Xu. Lvsm: A large view synthesis model with minimal 3d inductive bias. *arXiv preprint arXiv:2410.17242*, 2024.

Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D gaussian splatting for real-time radiance field rendering. *ACM Trans. Graph.*, 42(4):139:1–139:14, 2023.

Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Xin Kong, Shikun Liu, Xiaoyang Lyu, Marwan Taher, Xiaojuan Qi, and Andrew J Davison. Eschernet: A generative model for scalable view synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9503–9513, 2024.

Jeong-gi Kwak, Erqun Dong, Yuhe Jin, Hanseok Ko, Shweta Mahajan, and Kwang Moo Yi. Vivid-1-to-3: Novel view synthesis with video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6775–6785, 2024.

Peng Li, Yuan Liu, Xiaoxiao Long, Feihu Zhang, Cheng Lin, Mengfei Li, Xingqun Qi, Shanghang Zhang, Wei Xue, Wenhan Luo, et al. Era3d: High-resolution multiview diffusion using efficient row-wise attention. *Advances in Neural Information Processing Systems*, 37:55975–56000, 2024.

Weiyu Li, Xuanyang Zhang, Zheng Sun, Di Qi, Hao Li, Wei Cheng, Weiwei Cai, Shihao Wu, Jiarui Liu, Zihao Wang, et al. Step1x-3d: Towards high-fidelity and controllable generation of textured 3d assets. *arXiv preprint arXiv:2505.07747*, 2025.

Jinglin Liang, Jin Zhong, Hanlin Gu, Zhongqi Lu, Xingxing Tang, Gang Dai, Shuangping Huang, Lixin Fan, and Qiang Yang. Diffusion-driven data replay: A novel approach to combat forgetting in federated class continual learning. In *European Conference on Computer Vision*, pp. 303–319. Springer, 2024.

Yaron Lipman, Ricky TQ Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling. In *11th International Conference on Learning Representations, ICLR 2023*, 2023.

Lijie Liu, Tianxiang Ma, Bingchuan Li, Zhuowei Chen, Jiawei Liu, Gen Li, Siyu Zhou, Qian He, and Xinglong Wu. Phantom: Subject-consistent video generation via cross-modal alignment. *arXiv preprint arXiv:2502.11079*, 2025a.

Ruoshi Liu, Rundi Wu, Basile Van Hoorick, Pavel Tokmakov, Sergey Zakharov, and Carl Vondrick. Zero-1-to-3: Zero-shot one image to 3D object. In *ICCV*, pp. 9298–9309, 2023a.

Shikun Liu, Kam Woh Ng, Wonbong Jang, Jiadong Guo, Junlin Han, Haozhe Liu, Yiannis Douratsos, Juan C Pérez, Zijian Zhou, Chi Phung, et al. Scaling sequence-to-sequence generative neural rendering. *arXiv preprint arXiv:2510.04236*, 2025b.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. In *The Eleventh International Conference on Learning Representations (ICLR)*, 2023b.

Yuan Liu, Cheng Lin, Zijiao Zeng, Xiaoxiao Long, Lingjie Liu, Taku Komura, and Wenping Wang. Syncdreamer: Generating multiview-consistent images from a single-view image. *arXiv preprint arXiv:2309.03453*, 2023c.

Xiaoxiao Long, Yuan-Chen Guo, Cheng Lin, Yuan Liu, Zhiyang Dou, Lingjie Liu, Yuexin Ma, Song-Hai Zhang, Marc Habermann, Christian Theobalt, et al. Wonder3d: Single image to 3d using cross-domain diffusion. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9970–9980, 2024.

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. In *European Conference on Computer Vision*, pp. 405–421. Springer, 2020.

Michael Niemeyer, Jonathan T Barron, Ben Mildenhall, Mehdi SM Sajjadi, Andreas Geiger, and Noha Radwan. Regnerf: Regularizing neural radiance fields for view synthesis from sparse inputs. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5480–5490, 2022.

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

Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. DreamFusion: Text-to-3D using 2D diffusion. In *ICLR*, 2023.

Qiang Qu, Hanxue Liang, Xiaoming Chen, Yuk Ying Chung, and Yiran Shen. Nerf-nqa: No-reference quality assessment for scenes generated by nerf and neural view synthesis methods. *IEEE Transactions on Visualization and Computer Graphics*, 30(5):2129–2139, 2024.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pp. 8748–8763, 2021.

Xuanchi Ren, Tianchang Shen, Jiahui Huang, Huan Ling, Yifan Lu, Merlin Nimier-David, Thomas Müller, Alexander Keller, Sanja Fidler, and Jun Gao. Gen3c: 3d-informed world-consistent video generation with precise camera control. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 6121–6132, 2025.

Barbara Roessle, Norman Müller, Lorenzo Porzi, Samuel Rota Bulò, Peter Kontschieder, Angela Dai, and Matthias Nießner. L3dg: Latent 3d gaussian diffusion. In *SIGGRAPH Asia 2024 Conference Papers*, pp. 1–11, 2024.

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.

Kyle Sargent, Zizhang Li, Tanmay Shah, Charles Herrmann, Hong-Xing Yu, Yunzhi Zhang, Eric Ryan Chan, Dmitry Lagun, Li Fei-Fei, Deqing Sun, and Jiajun Wu. ZeroNVS: Zero-shot 360-degree view synthesis from a single image. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9420–9429, 2024.

Ruoxi Shi, Hansheng Chen, Zhuoyang Zhang, Minghua Liu, Chao Xu, Xinyue Wei, Linghao Chen, Chong Zeng, and Hao Su. Zero123++: a single image to consistent multi-view diffusion base model. *arXiv preprint arXiv:2310.15110*, 2023.

Stanislaw Szymanowicz, Chrisitian Rupprecht, and Andrea Vedaldi. Splatter image: Ultra-fast single-view 3d reconstruction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10208–10217, 2024.

Vikram Voleti, Chun-Han Yao, Mark Boss, Adam Letts, David Pankratz, Dmitry Tochilkin, Christian Laforte, Robin Rombach, and Varun Jampani. Sv3d: Novel multi-view synthesis and 3d generation from a single image using latent video diffusion. In *European Conference on Computer Vision*, pp. 439–457. Springer, 2024.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, et al. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 5294–5306, 2025.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20697–20709, 2024a.

Weimin Wang, Jiawei Liu, Zhijie Lin, Jiangqiao Yan, Shuo Chen, Chetwin Low, Tuyen Hoang, Jie Wu, Jun Hao Liew, Hanshu Yan, et al. Magicvideo-v2: Multi-stage high-aesthetic video generation. *arXiv preprint arXiv:2401.04468*, 2024b.

Zhao Wang, Aoxue Li, Lingting Zhu, Yong Guo, Qi Dou, and Zhenguo Li. Customvideo: Customizing text-to-video generation with multiple subjects. *arXiv preprint arXiv:2401.09962*, 2024c.

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.

Tong Wu, Jiarui Zhang, Xiao Fu, Yuxin Wang, Jiawei Ren, Liang Pan, Wayne Wu, Lei Yang, Jiaqi Wang, Chen Qian, et al. Omniobject3d: Large-vocabulary 3d object dataset for realistic perception, reconstruction and generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 803–814, 2023.

Jiawei Yang, Marco Pavone, and Yue Wang. Freenerf: Improving few-shot neural rendering with free frequency regularization. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 8254–8263, 2023.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. In *The Thirteenth International Conference on Learning Representations*, 2025.

Chongjie Ye, Lingteng Qiu, Xiaodong Gu, Qi Zuo, Yushuang Wu, Zilong Dong, Liefeng Bo, Yuliang Xiu, and Xiaoguang Han. StableNormal: Reducing diffusion variance for stable and sharp normal. *TOG*, 43(6):1–18, 2024.

Jianglong Ye, Peng Wang, Kejie Li, Yichun Shi, and Heng Wang. Consistent-1-to-3: Consistent image to 3d view synthesis via geometry-aware diffusion models. *arXiv preprint arXiv:2310.03020*, 2023.

Alex Yu, Vickie Ye, Matthew Tancik, and Angjoo Kanazawa. pixelnerf: Neural radiance fields from one or few images. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4578–4587, 2021.

Wangbo Yu, Jinbo Xing, Li Yuan, Wenbo Hu, Xiaoyu Li, Zhipeng Huang, Xiangjun Gao, Tien-Tsin Wong, Ying Shan, and Yonghong Tian. Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis. *arXiv preprint arXiv:2409.02048*, 2024.

Christina Zhang, Simran Motwani, Matthew Yu, Ji Hou, Felix Juefei-Xu, Sam Tsai, Peter Vajda, Zijian He, and Jialiang Wang. Pixel-space post-training of latent diffusion models. *arXiv preprint arXiv:2409.17565*, 2024.

Lan Zhang, Cheng Bo, Jiahui Hou, Xiang-Yang Li, Yu Wang, Kebin Liu, and Yunhao Liu. Kaleido: You can watch it but cannot record it. In *Proceedings of the 21st Annual International Conference on Mobile Computing and Networking*, pp. 372–385, 2015.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 586–595, 2018.

Shiwei Zhang, Jiayu Wang, Yingya Zhang, Kang Zhao, Hangjie Yuan, Zhiwu Qin, Xiang Wang, Deli Zhao, and Jingren Zhou. I2vgen-xl: High-quality image-to-video synthesis via cascaded diffusion models. *arXiv preprint arXiv:2311.04145*, 2023.

Zibo Zhao, Zeqiang Lai, Qingxiang Lin, Yunfei Zhao, Haolin Liu, Shuhui Yang, Yifei Feng, Mingxin Yang, Sheng Zhang, Xianghui Yang, et al. Hunyuan3d 2.0: Scaling diffusion models for high resolution textured 3d assets generation. *arXiv preprint arXiv:2501.12202*, 2025.

Chuanxia Zheng and Andrea Vedaldi. Free3d: Consistent novel view synthesis without 3d representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9720–9731, 2024.

Zangwei Zheng, Xiangyu Peng, Tianji Yang, Chenhui Shen, Shenggui Li, Hongxin Liu, Yukun Zhou, Tianyi Li, and Yang You. Open-sora: Democratizing efficient video production for all. *arXiv preprint arXiv:2412.20404*, 2024.

Jensen Zhou, Hang Gao, Vikram Voleti, Aaryaman Vasishta, Chun-Han Yao, Mark Boss, Philip Torr, Christian Rupprecht, and Varun Jampani. Stable virtual camera: Generative view synthesis with diffusion models. *arXiv preprint arXiv:2503.14489*, 2025.

Yufan Zhou, Ruiyi Zhang, Jiuxiang Gu, Nanxuan Zhao, Jing Shi, and Tong Sun. Sugar: Subject-driven video customization in a zero-shot manner. *arXiv preprint arXiv:2412.10533*, 2024.

APPENDIX

Beyond the main paper, we provide additional quantitative evaluations, generalization and failure analyses, efficiency and design studies, and representative generated sequences.

A ADDITIONAL QUANTITATIVE EVALUATION

A.1 FOUNDATION MODEL ANALYSIS

OrbitNVS uses the generative prior of a foundation model, but its performance is not tied to a particular backbone. Replacing Wan2.1-I2V-14B Wan et al. (2025) with CogVideoX-5B Yang et al. (2025) reduces PSNR by only 0.43 dB. Since the original Wan2.1 model does not support camera-pose conditioning for object-centric NVS, this result indicates that the camera adapter, normal-map branch, and pixel-space post-training adapt the video prior to the NVS task rather than merely inheriting its capability.

A.2 3D CONSISTENCY EVALUATION

We evaluate cross-view 3D consistency through reconstruction and camera-pose estimation. Figure 7 shows that views generated by OrbitNVS support a coherent 3D Gaussian Splatting (3DGS) reconstruction Kerbl et al. (2023) across substantially different viewpoints. We also use VGGT Wang et al. (2025) to estimate camera poses from generated videos. Table 3 shows the lowest rotation and translation errors among the evaluated methods; compared with SEVA Zhou et al. (2025), OrbitNVS reduces them by 0.97° and 0.303, respectively.

Table 3: Camera-pose accuracy estimated from generated videos using VGGT. Rotation and translation errors are computed against the ground-truth camera trajectory.

Method	Rot. Err. (°) ↓	Trans. Err. ↓
SEVA	11.29	0.589
SV3D	18.65	0.752
EscherNet	34.35	1.165
OrbitNVS	**10.32**	**0.286**

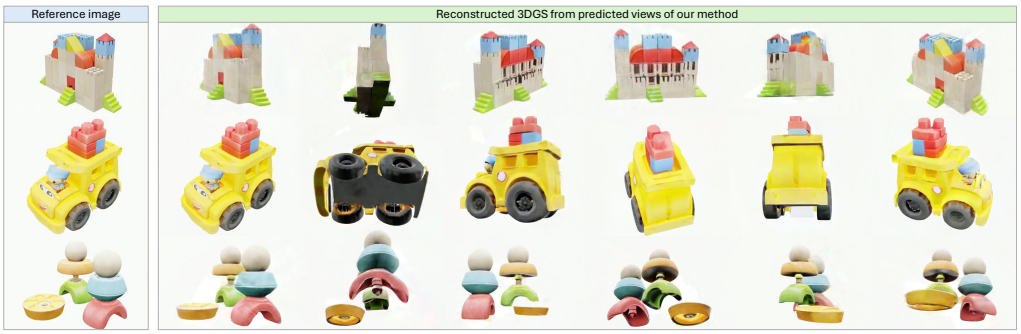

Figure 7: 3DGS reconstruction from views generated by OrbitNVS. The reference image is shown on the left, followed by renderings of the reconstructed 3DGS from multiple viewpoints.

A.3 NORMAL MAP GENERATION

OrbitNVS introduces a normal map generation branch that interacts with the main video synthesis branch to produce finer geometric details and improved consistency. We evaluate the quality of the generated normal maps both qualitatively and quantitatively. Qualitative results, provided in the supplementary videos, demonstrate that OrbitNVS generates high-quality normal maps. Quantitative results summarized in Table 4, using metrics adopted from prior work Ye et al. (2024); Fouhey et al.

(2013), further support this observation. For instance, on the GSO dataset, the mean angular error between the predicted and ground-truth normals is less than 16°, with over 85% of pixels exhibiting an angular error below 30°.

Table 4: Quantitative evaluation of normal map quality generated by OrbitNVS on both datasets. 'Mean' indicates the mean angular error between the predicted and ground-truth normals, and '$<x^\circ$' reports the percentage of angles with an error less than x°.

Dataset	Mean ↓	$<11.25^\circ$ ↑	$<22.5^\circ$ ↑	$<30^\circ$ ↑
GSO	15.91	59.18	79.13	85.48
OmniObject3D	18.72	48.41	73.78	82.16

A.4 VLM Ablation

OrbitNVS incorporates text as an additional control condition. During training, the text condition is generated by a VLM that captions the full videos, while during inference, the VLM produces captions only from the reference image(s). To investigate the role of the VLM, we replace the VLM-generated caption with a manually written prompt: "*Shooting around the object showcases its appearance from different perspectives.*"

Table 5 shows that inference without VLM captions yields comparable or marginally better performance. This implies that the capability for unseen-area synthesis is attributable to the strong visual priors in the video model, rather than the VLM. The captions mostly describe seen content and provide limited cues for unseen views. Another factor could be the limited capability of Qwen2.5-VL-7B Bai et al. (2025), where inaccurate captions might lead to suboptimal performance. Employing a more powerful VLM could be a promising direction for future work.

Table 5: VLM ablation on GSO (top) and OmniObject3D (bottom). Each entry reports PSNR↑, SSIM↑, and LPIPS↓.

GSO			
Method	0°	30°	60°
OrbitNVS	23.7 0.89 0.11	21.9 0.87 0.13	21.0 0.86 0.15
w/o VLM	24.0 0.90 0.10	21.9 0.87 0.12	21.0 0.87 0.14

OmniObject3D			
Method	0°	30°	60°
OrbitNVS	21.0 0.87 0.14	20.4 0.86 0.14	19.9 0.85 0.16
w/o VLM	21.0 0.87 0.13	20.6 0.86 0.14	19.8 0.85 0.15

B Generalization and Failure Analysis

B.1 In-the-Wild Generalization

We additionally evaluate OrbitNVS on real photographs captured with a phone. After removing the image background, each photograph is used as the reference view without scene-specific optimization. Figure 8 shows that OrbitNVS generalizes to these in-the-wild inputs and produces plausible novel views and normal maps despite being trained on rendered object videos.

B.2 Failure Cases

OrbitNVS can fail on uncommon or visually ambiguous objects. As illustrated in Figure 9, the model may infer an incorrect object category or depth structure, such as interpreting stacked building blocks as a brick wall. These failures reflect ambiguity in the single-view evidence and limitations of the visual prior used to infer unobserved content.

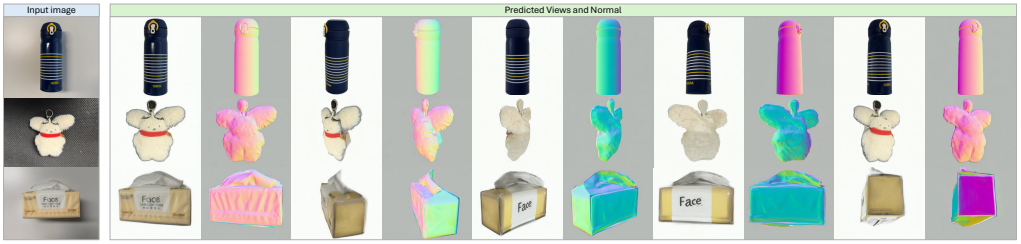

Figure 8: In-the-wild novel-view synthesis from real phone photographs. Each group shows the background-removed input image followed by novel-view RGB predictions and their corresponding normal maps.

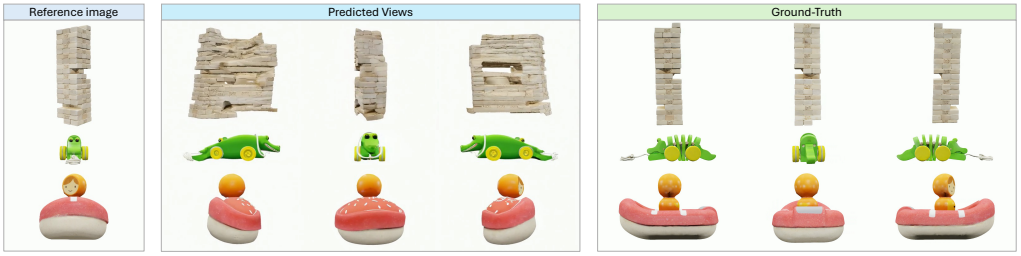

Figure 9: Failure cases on uncommon or ambiguous objects. From left to right, each group shows the reference image, predicted novel views, and ground-truth views.

C Efficiency and Design Analysis

C.1 Implementation Details

Our training data consists of 320k assets from the Step1X-3D dataset Li et al. (2025), which were filtered from the 800k assets in Objaverse-V1 Deitke et al. (2023). For each asset, we rendered a 129-frame orbital video and its corresponding normal maps at a 448×448 resolution. To ensure viewpoint diversity, we randomly employed one of two camera motion modes during rendering: the first was a random mode, where the camera orientation was adjusted with random offsets at each step, and the second was a sinusoidal oscillation mode, similar to Equations 8 and 9, but with random biases, frequencies, amplitudes, and initial phases. From each generated video, we randomly sampled 61 frames for training to further enhance the diversity of camera motion. For text input, during training, we used Qwen2.5-VL-7B Bai et al. (2025) to generate detailed captions describing the object in the video. At test time, captions were generated for the reference images to form the text prompt.

Our training process was divided into two stages. In the first stage, we trained the model for 3200 steps using only a latent-space loss, which required approximately 60 hours. In the second stage, we fine-tuned the model for an additional 1800 steps using a combination of latent- and pixel-space losses, which took an additional 66 hours. The model was trained with a total batch size of 64 on a cluster of 8 machines, each equipped with 8 NVIDIA A800 GPUs with 80 GB VRAM. We adopted a grouped learning rate strategy, applying a learning rate of 10^{-5} to parameters loaded from pretrained weights and 10^{-4} to newly initialized parameters, such as the camera adapter.

C.2 Efficiency and Training Behavior

Appendix C.1 reports the two-stage training schedule and hardware configuration. Here, we provide a more detailed runtime analysis. Inference takes approximately 15 minutes per video on one NVIDIA A800 GPU. The normal-map branch increases computation because it introduces additional high-resolution tokens: generating at 448 pixels with the normal branch takes approximately 17 minutes, which is comparable to generating at 640 pixels without the branch, while 448-pixel generation without the branch takes approximately 5 minutes. Pixel-space supervision increases the

per-step training time from approximately 68 seconds to 132 seconds, but it is used only during the second training stage and does not affect inference speed.

Figure 10 shows that both training stages converge stably. We also observe that training on the full Objaverse collection slightly degrades performance, likely because it introduces more low-quality assets; this motivates the filtered Step1X-3D training set used in the main paper. The mixture of random and sinusoidal camera trajectories improves the robustness of inference-time camera control, while removing either component degrades performance.



Figure 10: Training losses across the latent-space training stage and pixel-space post-training stage.

C.3 Architecture and Trajectory Analysis

Camera conditioning of the baselines. SV3D Voleti et al. (2024) conditions generation on azimuth and elevation, whereas EscherNet Kong et al. (2024) and SEVA use full camera extrinsics. The more restricted camera parameterization limits SV3D when the evaluation trajectory varies camera distance or intrinsics, which helps explain why its ranking differs from results reported under its original setting.

Shared normal-map branch. OrbitNVS shares parameters between RGB and normal-map generation. This is the simplest and most parameter-efficient design, and the quantitative and qualitative results in the main paper show that a shared transformer can model both modalities effectively. Decoupling modality-specific parameters may offer additional specialization, but is not required for the gains reported in this work.

Non-orbital trajectories. OrbitNVS supports non-orbital camera inputs, although smooth orbital trajectories are closer to the pretraining distribution of Wan2.1 and produce better reconstructions. As a diagnostic, the Wan VAE obtains approximately 32.1 dB reconstruction PSNR for orbital videos and 30.3 dB for non-orbital videos formed from random viewpoints. This gap indicates that part of the degradation on irregular trajectories originates from the foundation model's video representation. When applicable, unordered viewpoints can be reordered and interpolated into smoother camera paths.

D Additional Qualitative Results

We have uploaded the complete videos of all visual results (Figures 1, 3, 4, and 5) from the main text. For the qualitative comparison in Figure 3, we also include videos generated by the baseline methods. For the quantitative analyses in Tables 1 and 2, representative generated videos are provided due to storage constraints. Figures 11–14 show the reference images, predicted novel-view videos and normal maps, and the corresponding ground-truth videos and normal maps.

OrbitNVS supports both single- and multi-image inputs. In each demonstration, the "reference image(s)" column is shown as a masked video, where unmasked frames indicate the input observations. Figures 11–13 use a single reference image, whereas Figure 14 uses two reference images. The results demonstrate geometry- and appearance-consistent novel-view synthesis and the ability to infer unseen regions.

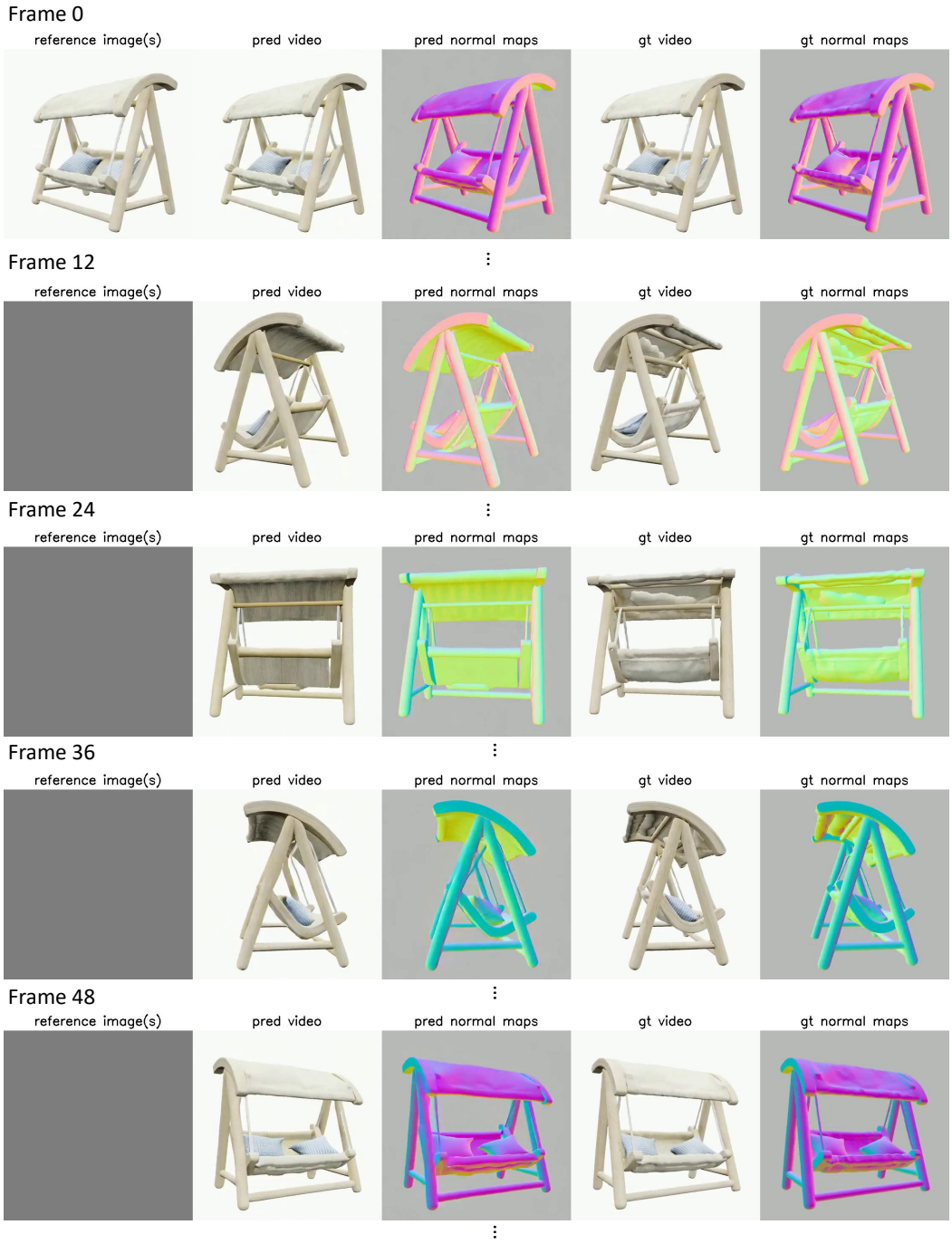

Figure 11: Additional single-view result. Each row shows the reference image, predicted novel-view RGB and normal map, and the corresponding ground-truth RGB and normal map.

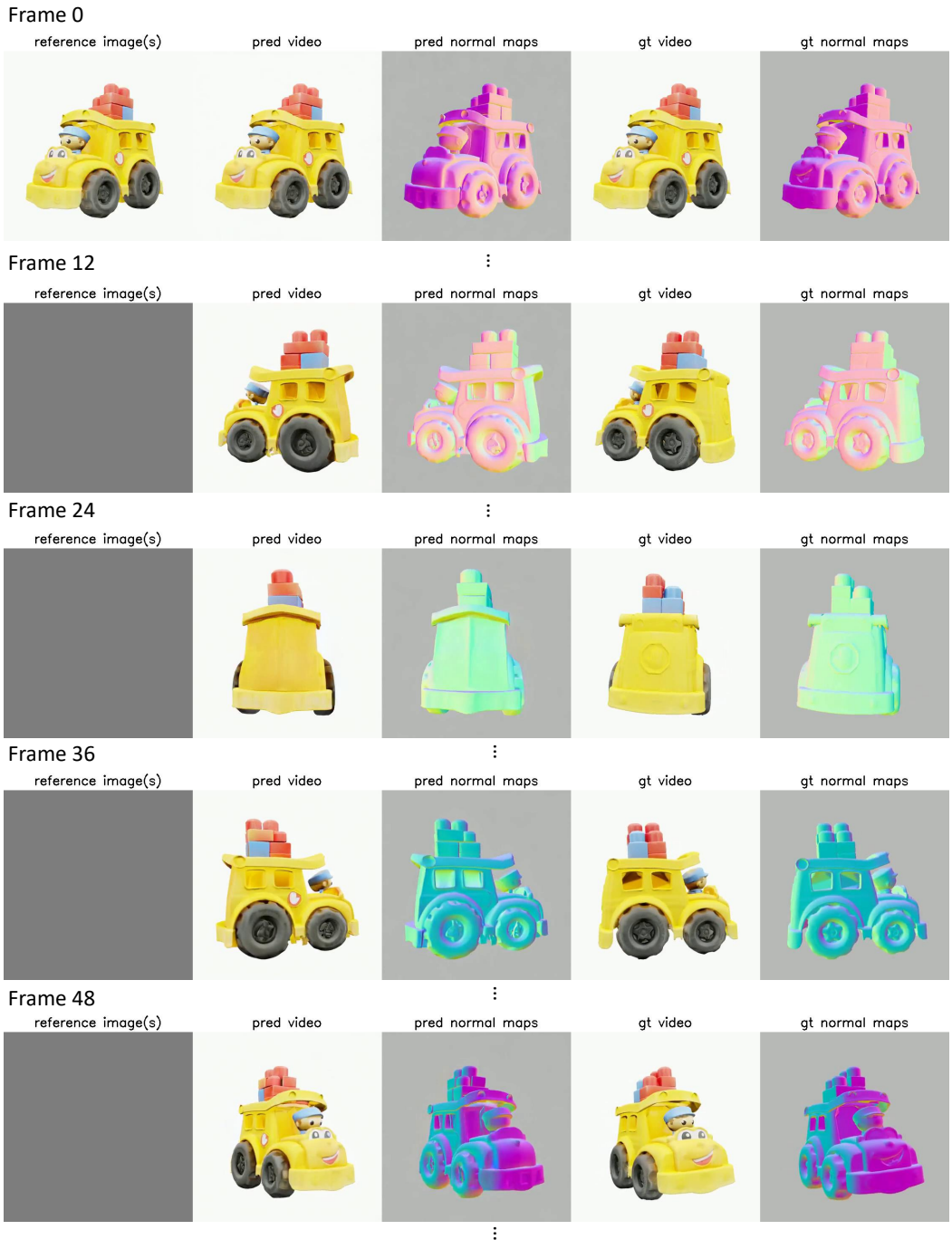

Figure 12: Additional single-view result using the same layout as Figure 11.

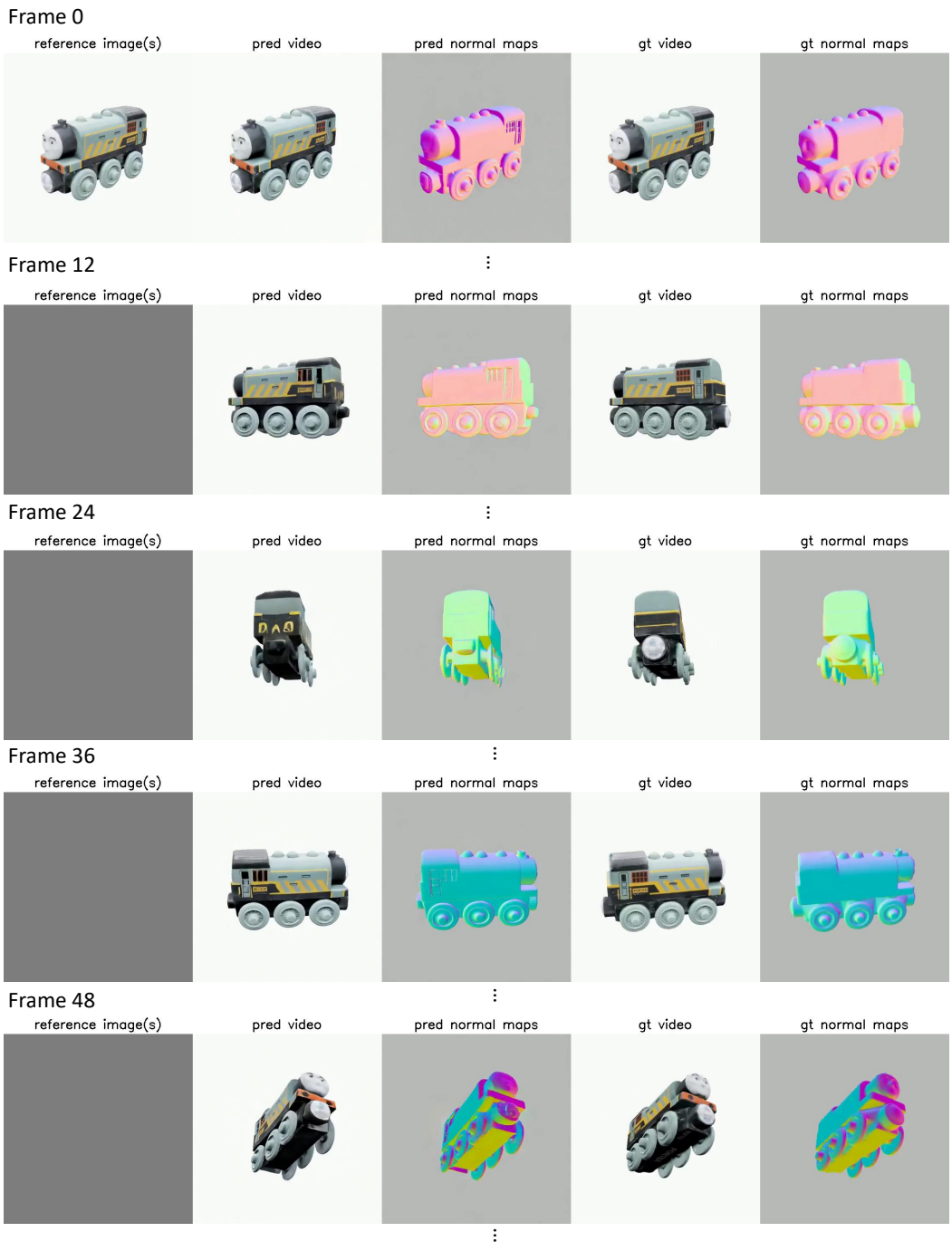

Figure 13: Additional single-view result using the same layout as Figure 11.

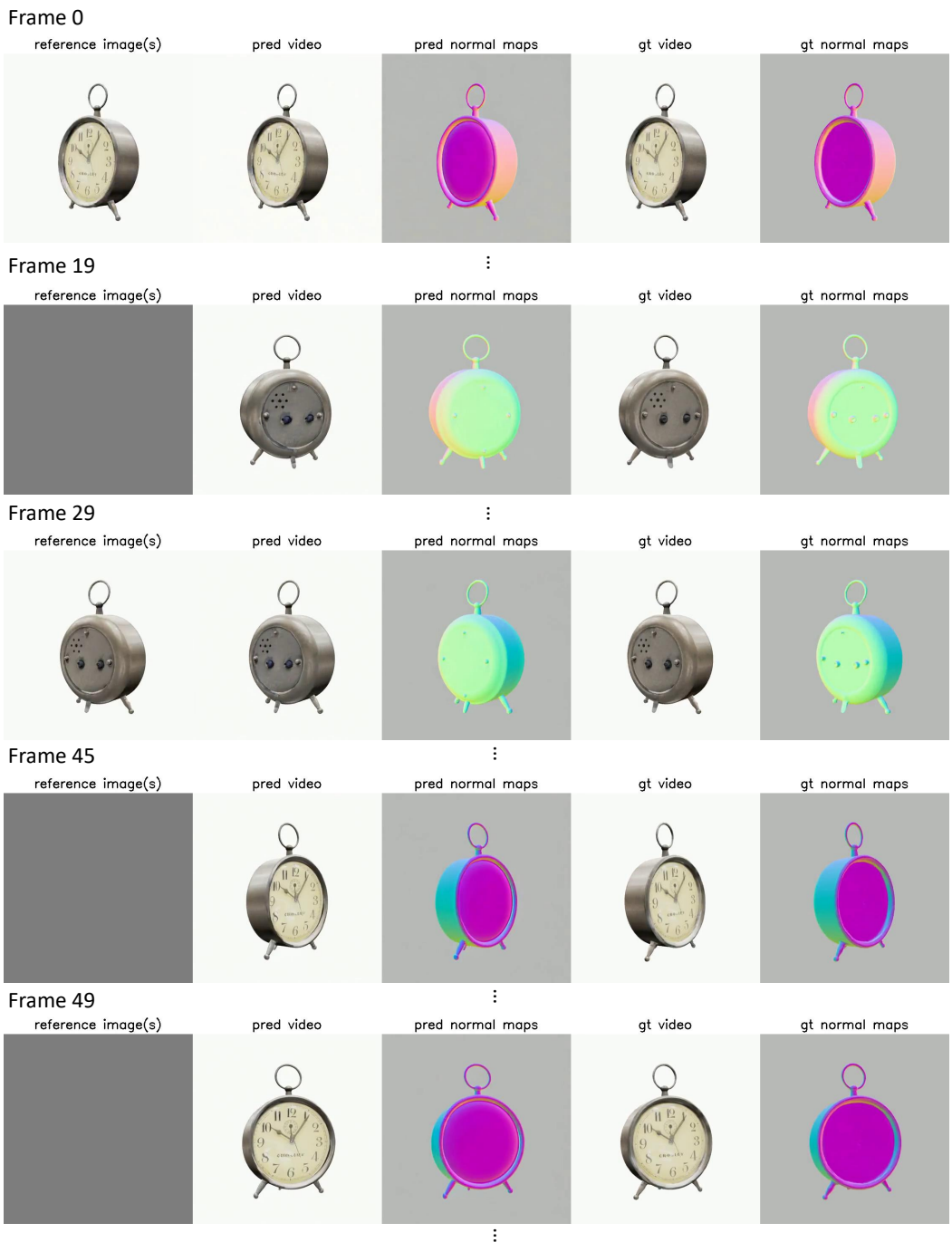

Figure 14: Additional two-view result. Unmasked frames in the reference column indicate the two input observations.