

ConfCtrl: Enabling Precise Camera Control in Video Diffusion via Confidence-Aware Interpolation

Liudi Yang¹, George Eskandar⁴, Fengyi Shen³, Mohammad Altillawi⁴,
Yang Bai^{2,5}, Chi Zhang⁴, Ziyuan Liu^{4*}, and Abhinav Valada¹

¹ University of Freiburg, Germany

² Ludwig Maximilian University of Munich, Heidelberg, Germany

³ Technical University of Munich, Germany

⁴ Huawei Heisenberg Research Center (Munich)

⁵ Munich Center for Machine Learning (MCML)

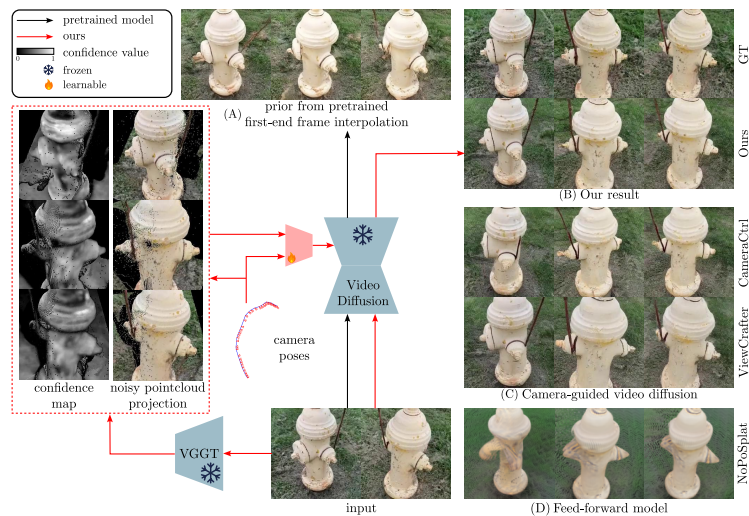


Fig. 1: Overview of ConfCtrl. (A) demonstrates the strong prior of a first- and end-frame video interpolation model. This is fully inherited by our model for further geometric improvement. (B) **Our ConfCtrl enables precise camera control in video diffusion models.** (C) shows that camera-guided video diffusion models struggle to strictly follow the target camera poses. (D) indicates that feedforward models underperform due to limited generative capacity, leading to inaccurate estimation of Gaussian Splat parameters.

Abstract. We address the challenge of novel view synthesis from only two input images under large viewpoint changes. Existing regression-based methods lack the capacity to reconstruct unseen regions, while camera-guided diffusion models often deviate from intended trajectories due to noisy point cloud projections or insufficient conditioning

* Corresponding author

from camera poses. To address these issues, we propose **ConfCtrl**, a confidence-aware video interpolation framework that enables diffusion models to follow prescribed camera poses while completing unseen regions. ConfCtrl initializes the diffusion process by combining a confidence-weighted projected point cloud latent with noise as the conditioning input. It then applies a Kalman-inspired *predict-update* mechanism, treating the projected point cloud as a noisy measurement and using learned residual corrections to balance pose-driven predictions with noisy geometric observations. This allows the model to rely on reliable projections while down-weighting uncertain regions, yielding stable, geometry-aware generation. Experiments on multiple datasets show that ConfCtrl produces geometrically consistent and visually plausible novel views, effectively reconstructing occluded regions under large viewpoint changes.

Keywords: Novel View Synthesis · Camera-guided Video Diffusion

1 Introduction

Novel view synthesis from sparse input views, especially under large viewpoint changes, remains a challenging problem that has attracted significant attention. Although recent methods [1, 4, 5, 14, 18, 25, 26, 54, 65, 69, 78] have achieved remarkable progress, their performance often degrades when the input views are captured from widely separated viewpoints. Overcoming this limitation is crucial for real-world applications, where only a few images with large viewpoint differences are typically available.

Existing approaches can be broadly categorized into *regression-based* and *diffusion-based* methods. Regression-based methods learn explicit 3D scene representations in a feedforward manner and render images conditioned on target camera poses. While these methods can accurately follow the specified camera trajectory, they often lack sufficient generative capability under sparse or incomplete observations, resulting in noticeable rendering artifacts (Figure 1(D)). In contrast, diffusion-based approaches benefit from powerful generative priors and strong completion ability gained through large-scale pretraining. However, current camera-guided diffusion models, such as CameraCtrl [18] (Figure 1(C)), still struggle to precisely follow target camera poses, limiting their controllability in view synthesis.

A potential solution is to either enhance regression-based methods with stronger generative capabilities or enable diffusion-based methods to strictly follow desired camera trajectories. We observe that the first approach requires substantial training data to achieve satisfactory performance, even when integrating pretrained video diffusion to inject generative latents into a Gaussian decoder [15, 36]. Alternatively, the second approach demands a carefully designed camera-conditioning mechanism that ensures precise adherence to target camera motions while generating plausible scene contents with temporal and spatial consistency at the same time. Although this approach is advantageous in reusing the rich priors of pretrained video diffusion models, thereby mitigating

the need for extensive training data, the main challenge is twofold: (1) the single-image-to-video generation paradigm tends to accumulate deviation over frames, and (2) the absence of explicit 3D priors limits accurate geometric consistency.

To address problem (1), we observe that employing a video interpolation model can naturally enhance geometric consistency through its head-tail interpolation mechanism (Figure 1(A)). For problem (2), prior works [4, 48, 71, 76] incorporate projected point clouds obtained from monocular depth estimation models. With this strong 3D prior in pixel space, the model can achieve better camera following capability if the projected point cloud is sufficiently accurate. In practice, however, these approaches often suffer from distortion and scale ambiguity of real point cloud estimation, and conditioning on such noisy inputs often leads to suboptimal performance (ViewCrafter in Figure 1(C)). Recent 3D foundation models [33, 59, 60] offer more reliable 3D priors by estimating projected point clouds from paired images, leading to improved geometric accuracy. However, these models are still imperfect, as residual artifacts and warping can degrade the quality of the conditioning signal. This motivates a camera condition mechanism that can compensate for *uncertainty* in projected point clouds.

Building on these insights, we propose ConfCtrl, a confidence-aware framework for novel view synthesis under large viewpoint changes. ConfCtrl builds on a pretrained video interpolation model and introduces a confidence-guided inductive bias based on point clouds projected from 3D foundation models. By incorporating confidence-aware conditioning, ConfCtrl enables the video diffusion model to more precisely follow target camera poses, effectively exhibiting feedforward model-like behavior. Concretely, instead of initializing the rectified flow model from pure noise, we start from a sum of a confidence-weighted projected point cloud latent and noise. This initialization improves the adaptation from pretrained temporal interpolation knowledge toward novel view synthesis. The confidence maps act as reliability weights that quantify the trustworthiness of each projected point in the latent space.

Second, inspired by the well-known Kalman Filter [27], we introduce a camera conditioning mechanism that jointly encodes projected point clouds and camera poses to mitigate uncertainty in 3D geometric priors. This mechanism operates through two submodules: a prediction submodule, which conditions the latent state on the target camera pose, and an update submodule, which uses the projected point cloud as a noisy measurement to refine the prediction, improving stability and precision in camera control. We validate the effectiveness of our approach across multiple datasets, consistently surpassing the baseline methods. Furthermore, by leveraging the generalization capabilities of pretrained video diffusion models, our method exhibits strong zero-shot performance on out-of-distribution scenarios.

Our main contributions are summarized as follows:

- We show that, under the challenging sparse inputs setting, leveraging a pretrained video interpolation model will provide stronger 3D consistency for novel view synthesis.

- We introduce a diffusion initialization strategy using a latent derived from the projected point cloud, enabling more effective adaptation from interpolation to novel view synthesis.
- We propose a camera-pose conditioning mechanism that jointly encodes projected point clouds and camera poses via a *predict–update* architecture, handling uncertainty in geometric priors to achieve robust geometry and precise camera control.
- Extensive experiments on multiple datasets demonstrate that our method consistently outperforms existing baselines and achieves strong zero-shot generalization.

2 Related Work

Regression-based Methods: With the rapid progress of 3D neural rendering [29, 44], feedforward approaches have gained increasing attention due to their generality and fast inference. Recent methods [5, 8, 12, 20, 22, 28, 34, 45, 52, 67, 69, 73, 74, 80] directly infer 3D scene structures from input images and regress rendering parameters for novel view synthesis. However, their performance degrades significantly with sparse inputs and large viewpoint changes. This limitation mainly arises from the lack of generative priors, which hinders accurate 3D representation and the inference of unseen regions. AnySplat [25] alleviates this issue by incorporating the foundation model VGGT [59] for Gaussian positioning, but it still cannot hallucinate unseen regions. LVSM [26] adopts a data-driven design with minimal 3D inductive biases, yet remains challenged by large viewpoint variations. To improve generative capability, later works [9, 42, 63, 66] introduce generative models as refinement modules for scene completion or Gaussian deblurring. Nevertheless, these approaches depend on noisy renderings as input, leading to performance degradation when the initial Gaussian reconstruction is severely corrupted.

Diffusion-based Methods with Camera Control: Another line of research focuses on leveraging diffusion models for novel view synthesis. Recent works [1, 2, 18, 21, 32, 35, 62, 70, 78] introduce camera pose conditioning into video diffusion models, allowing them to synthesize viewpoint-varying sequences. These methods provide video diffusion models with controllable generative power but still suffer from limited camera controllability and the absence of explicit 3D priors, leading to inconsistent content generation and imperfect adherence to target camera trajectories. To mitigate this problem, several approaches [4, 6, 23, 39, 48, 71, 76] estimate depth from images and use the corresponding projections as 3D priors. While this improves 3D consistency, performance remains sensitive to inaccuracies in the projected point clouds. IDCNet [41] jointly models depth and RGB spaces to inject more precise 3D information, but it requires large-scale depth video supervision, which limits its scalability. In our work, we investigate how confidence learning can be leveraged to better utilize noisy 3D priors.

Furthermore, recent studies explore utilizing video diffusion backbones as generative priors for Gaussian Splatting, thereby enhancing the generative ca-

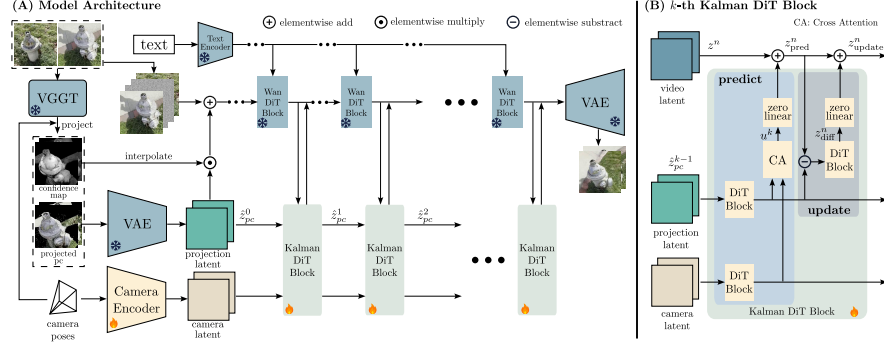


Fig. 2: The **ConfCtrl** framework. (A) *Model Architecture*. Our framework builds upon the Wan2.1-Interpolation model [56]. We introduce two key designs to enhance precise camera-following capability: (1) noise initialization using a confidence-weighted projected point cloud latent, and (2) predict-update camera conditioning mechanism that balance the predicted state with the projected point cloud as a noisy measurement. (B) *Kalman DiT Block*. It consists of two submodules: predict and update.

pability of feedforward approaches [15, 16, 36, 50]. However, adapting information from a video diffusion backbone to a Gaussian parameter decoder remains challenging since purely pixel-space supervision without explicit 3D constraints often leads to suboptimal results.

Confidence Learning in 3D Reconstruction: An emerging direction in 3D reconstruction is the modeling of uncertainty. Klasson *et al.* [31] analyze the origins and types of uncertainty in GS [29] and NeRF [44]. Several works improve reconstruction quality by introducing uncertainty estimation in Gaussian parameters [17, 49, 64], uncertainty-aware regularization [30, 42], or modeling an uncertainty field [11, 55]. For diffusion-based methods [3, 40, 57, 58], uncertainty-aware losses are typically employed to balance training, assigning higher weights to regions with greater confidence. In our work, we explore confidence learning with 3D foundation models, where confidence maps typically serve as auxiliary supervision during pretraining [33, 59, 60] or as state update rules [7]. We introduce a novel mechanism that assigns weights as confidence values derived from VGGT during noise initialization. Furthermore, inspired by the Kalman filter’s predict–update scheme for handling uncertainty, we design a module that fuses features with noisy measurements via learned residual corrections, enabling adaptive refinement of the representation.

3 Preliminary

Adaptive Correction for Uncertainty in Kalman Filter: The Kalman filter is a recursive algorithm that manages uncertainty in state estimation through a *predict–update* loop, explicitly weighting predictions and measurements according to their respective confidences. First, the state $\hat{x}$ is projected forward in the

prediction step:

$$\hat{x}_{\text{pred}} = f(\hat{x}, u), \quad (1)$$

where u denotes the control input. The state is then corrected by integrating a new measurement y in the update step:

$$\hat{x}_{\text{update}} = \hat{x}_{\text{pred}} + \Delta(\hat{x}_{\text{pred}}, y), \quad (2)$$

where Δ denotes the Kalman gain-weighted residual between the prediction and the measurement. This iterative process fuses predictions with noisy measurements to improve state estimation.

Motivated by the predict-update principle, we introduce a *predict-update* structure as a per-block feature refinement module within our conditional control architecture (Sec. 4.2, Fig. 2(B)). Unlike classical Kalman filters, it does not perform recursive Bayesian state estimation across diffusion steps, as diffusion [19] violates linear system assumptions. Instead, it serves as an architectural mechanism to integrate control signals and geometric observations. Each block contains a *prediction* submodule that generates features conditioned on the control input u (e.g., target camera parameters), and an *update* submodule that refines them using projected 3D point clouds as noisy measurements. This refinement improves geometric consistency and adherence to the target camera pose.

Video Interpolation Conditioned on the First and Last Frames: Video interpolation synthesizes intermediate frames between start and end frames while preserving temporal and spatial consistency. Given $\mathbf{I}_0$ and $\mathbf{I}_T$, the goal is to generate $\{\mathbf{I}_t\}_{t=1}^{T-1}$ forming a smooth transition. Modern methods [56] learn motion and appearance transformations conditioned on the two endpoints, providing strong priors for camera control and view synthesis (Figure 1(A)).

4 Method

Our framework (Fig. 2) builds on a pretrained video interpolation model. To handle uncertainty in 3D priors, we introduce two components: a confidence-aware point cloud latent for noise initialization (Sec. 4.1) and a predict-update conditioning mechanism for robust camera integration (Sec. 4.2). The model is trained with a rectified flow objective (Sec. 4.3).

4.1 Initialization Beyond Pure Noise

Existing methods typically employ a ControlNet-like [75] mechanism to condition the projected point cloud for camera control [4, 48]. The projected point cloud is obtained from large-scale 3D foundation models [33, 59, 60, 68] and serves as a geometric prior. While these approaches perform well when the estimated point cloud is sufficiently accurate, their performance often degrades with sparse input views under large viewpoint changes.

In contrast, we propose to integrate this 3D prior into the *initial distribution* for the rectified flow model, rather than initializing from pure Gaussian noise. To address inaccuracies in the point cloud, we introduce a *point-wise confidence map* from VGGT that quantifies the reliability of each estimated point. This confidence map is projected into the target camera space to obtain a *frame-wise confidence* that reflects the uncertainty for each target view. Formally, given the projected point cloud latent $\hat{z}_{pc}^0$ and its corresponding confidence weights $\mathbf{w}$, we compute the confidence-aware initialization as:

$$z_0 = \lambda_1 \cdot (\mathbf{w} \odot \hat{z}_{pc}^0) + \lambda_2 \cdot \epsilon, \quad (3)$$

where $\odot$ denotes element-wise multiplication, $\epsilon \sim \mathcal{N}(0, I)$ represents Gaussian noise, and $\lambda_1, \lambda_2 \in [0, 1]$ control the balance between the confidence-aware latent and random noise. In our experiment, we select $\lambda_1 = 1, \lambda_2 = 1$. To ensure compatibility in temporal channel and spatial resolution, we apply linear interpolation to resize the projected point cloud latent before computing the weighted sum. This confidence-guided initialization provides a more reliable initial distribution, improving adaptation from video interpolation to novel view synthesis.

4.2 Predict-Update Camera Conditioning

Existing methods for camera conditioning typically combine projected point clouds as a 3D geometric prior with camera embeddings (e.g., Plücker coordinates) to ensure pose controllability [4, 48]. A common practice is to simply concatenate these two representations, which does not explicitly account for the inherent uncertainty and noise (e.g., occlusion, warping) present in projected point clouds.

To address this, we propose a *predict-update* architecture that adaptively fuses geometric priors with camera control signals using learned residual corrections. Motivated by the predict-update paradigm in Kalman Filter (Sec. 3), the process operates within each Kalman DiT block (see Figure 2 (B)) and consists of two submodules: *prediction* and *update*. The prediction stage generates an initial feature conditioned solely on the target camera pose, while the update stage refines it by integrating the predicted feature with the noisy projected point cloud, effectively weighting the geometric observation according to its learned reliability.

Prediction. For the k -th Kalman DiT block, we treat the target camera pose as a control input u^k , which guides the video latent z^n from the n -th Wan DiT block toward the desired viewpoint. The predicted latent z_{pred}^n is defined as:

$$z_{\text{pred}}^n = z^n + \text{zero-linear}(u^k) \triangleq f(z^n, u^k), \quad (4)$$

analogous to the state transition in Kalman filtering (Eq. 1). The control input u^k is implemented via a cross-attention mechanism between a camera latent (encoding the target camera Plücker embeddings) and the projection latent of the point cloud (which encodes the 3D prior and known camera poses). The

resulting feature is then added back into z^n through a zero-initialized linear layer, ensuring stable residual integration while preserving the original latent information. This submodule produces a predicted feature that combines the geometric prior with the intended camera motion. Alternative designs for this stage are evaluated in Table 4(f-g).

Update. In the *update* submodule of the k -th Kalman DiT block, the projected point cloud latent $\hat{z}_{pc}^{k-1}$ (i.e., $\hat{z}_{pc}^0$ after passing through $k-1$ Kalman DiT blocks) is treated as a noisy measurement. To update the predicted latent z_{pred}^n with measurement $\hat{z}_{pc}^{k-1}$, we learn a correction residual

$$z_{\text{diff}}^n = \text{Diff}(z_{\text{pred}}^n - \hat{z}_{pc}^{k-1}) \quad (5)$$

using additional DiT blocks [46]. These blocks, connected via zero-initialized linear layers, output the residual based on the discrepancy between the prediction and the measurement. The final refined latent is then

$$z_{\text{update}}^n = z_{\text{pred}}^n + \text{zero-linear}(z_{\text{diff}}^n) \triangleq z_{\text{pred}}^n + \Delta(z_{\text{pred}}^n, \hat{z}_{pc}^{k-1}) \quad (6)$$

analogous to the measurement update in Kalman Filter (Eq. 2). This residual-based fusion modulates the influence of geometric observations through the learned correction residual, attenuating inconsistent measurements and improving geometric consistency and camera controllability.

4.3 Training Objective

We adopt the standard *rectified flow loss* [38, 43] as our training objective, but replace the initial distribution with our proposed *confidence-aware latent projection*. This modification provides a more stable latent initialization to follow camera trajectories. Formally, given a latent variable z_t at time t , we define its evolution under the rectified flow as

$$z_t = (1 - t) z_0 + t z_1, \quad (7)$$

where z_0 is sampled from the confidence-aware latent prior (Eq. 3), and z_1 represents the target latent derived from the ground-truth video representation. The model learns a velocity field $v_\theta(z_t, t)$ that predicts the instantaneous flow between z_0 and z_1 . The rectified flow loss is then given by

$$\mathcal{L}_{\text{RF}} = \|v_\theta(z_t, t) - (z_1 - z_0)\|_2^2, \quad (8)$$

where the difference $z_{\text{target}} = (z_1 - z_0)$ acts as the supervision signal for the flow direction.

To improve training stability and maintain fine-grained structural consistency when adapting video interpolation for novel view synthesis under noisy or jerky camera poses, we introduce a *latent gradient regularization* term that enforces alignment of spatial gradients in the latent space:

$$\mathcal{L}_{\text{grad}} = \|\nabla_x v_\theta - \nabla_x z_{\text{target}}\|_1 + \|\nabla_y v_\theta - \nabla_y z_{\text{target}}\|_1. \quad (9)$$

The overall training objective is thus defined as

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{RF}} + \lambda_{\text{grad}} \mathcal{L}_{\text{grad}}, \quad (10)$$

with $\lambda_{\text{grad}} = 0.05$ in our experiments. This regularization encourages the preservation of high-frequency details and enforces local spatial consistency, slightly reducing noise and flicker artifacts under rapid viewpoint changes while improving the model’s ability to accurately follow the camera trajectory (Table 4(e)).

5 Experimental Results

5.1 Setup

Dataset. We train and evaluate our method on three datasets respectively: CO3D-Hydrant [47], CO3D-Teddybear [47], and DL3DV [37], all rendered at a resolution of 256×256 pixels due to limited computational resources. Specifically, we use only 250, 450, and 200 sequences from each dataset for training, and the remaining sequences for testing.

Metric. We report the PSNR, LPIPS, and SSIM metrics to assess the quality of novel view synthesis, and *translation error* (m) and *rotation error* (radian) to evaluate camera controllability. The predicted camera poses are obtained using VGGT [59], which estimates the camera trajectory from the generated video with respect to the first input image. The translation error is computed as the mean Euclidean distance between the predicted and ground-truth camera positions:

$$E_t = \frac{1}{N} \sum_{i=1}^N \|\mathbf{t}_i^{\text{pred}} - \mathbf{t}_i^{\text{gt}}\|_2, \quad (11)$$

where $\mathbf{t}_i^{\text{pred}}$ and $\mathbf{t}_i^{\text{gt}}$ denote the predicted and ground-truth camera translations, respectively. The rotation error measures the angular difference between predicted and ground-truth camera orientations:

$$E_r = \frac{1}{N} \sum_{i=1}^N \arccos \left(\frac{\text{trace}(\mathbf{R}_i^{\text{pred}} \mathbf{R}_i^{\text{gt}^\top}) - 1}{2} \right), \quad (12)$$

where $\mathbf{R}_i^{\text{pred}}$ and $\mathbf{R}_i^{\text{gt}}$ represent the predicted and ground-truth rotation matrices.

Implementation Details. Our implementation is built upon the 1.3B Wan-2.1-InP model [56]. We insert 10 Kalman DiT blocks uniformly into the architecture to incorporate camera conditioning. For video diffusion, we use sequences of 33 frames. The model is trained for 20k steps with a learning rate of 10^{-5} until convergence. Due to limited computational resources and leveraging the strong prior of the interpolation model, we adopt a batch size of 1 and train on a single GPU, with a memory footprint of approximately 40 GB. Training for 20k steps takes about 24 hours on a single GPU, while inference with 50 diffusion steps takes approximately 10 seconds per scene.

Method	Cam PC	CO3D-Hydrant					CO3D-Teddybear					DL3DV				
		LPIPS ↓	PSNR ↑	SSIM ↑	E _t ↓	E _r ↓	LPIPS ↓	PSNR ↑	SSIM ↑	E _t ↓	E _r ↓	LPIPS ↓	PSNR ↑	SSIM ↑	E _t ↓	E _r ↓
NopoSplat [69]	✓	0.561	14.23	0.246	0.212	0.151	0.553	14.95	0.418	0.397	0.321	0.475	14.58	0.377	0.505	0.377
DepthSplat [67]	✓	0.537	14.22	0.254	0.464	0.373	0.503	14.97	0.373	0.438	0.369	0.478	14.05	0.352	0.513	0.423
MvSplat [8]	✓	0.710	12.94	0.228	1.372	1.110	0.629	14.96	0.398	0.889	0.729	0.530	13.69	0.373	0.532	0.401
PixelSplat [5]	✓	0.537	14.81	0.317	0.266	0.309	0.560	15.88	<u>0.462</u>	0.637	0.528	0.476	15.56	<u>0.424</u>	0.401	0.322
AnySplat [25]	✓	0.368	15.36	<u>0.322</u>	0.201	0.193	0.395	16.44	0.454	0.345	0.238	0.303	16.13	0.408	0.221	0.187
LVSM [26]	✓	0.435	14.25	0.288	0.282	0.254	0.415	15.26	0.407	0.395	0.386	0.326	15.73	0.374	0.279	0.245
CameraCtrl [18]	✓	0.415	14.51	0.285	0.260	0.194	0.461	14.95	0.394	0.716	0.565	0.343	15.25	0.347	0.270	0.222
ViewCrafter [71]	✓	0.413	14.61	0.291	0.249	0.181	0.408	15.94	0.419	0.361	0.271	0.324	15.40	0.370	0.264	0.201
SeVA [78]	✓	0.408	14.56	0.232	0.302	0.284	0.409	16.04	0.431	0.343	0.302	0.319	15.80	0.373	0.253	0.198
Gen3R [21]	✓	<u>0.341</u>	15.32	0.321	<u>0.189</u>	<u>0.144</u>	<u>0.368</u>	<u>16.92</u>	0.452	<u>0.234</u>	<u>0.159</u>	0.296	<u>16.34</u>	0.391	0.225	0.160
Uni3C [4]	✓	0.350	<u>15.41</u>	<u>0.322</u>	0.219	0.167	0.382	16.45	0.438	0.271	0.198	<u>0.306</u>	15.99	0.395	<u>0.213</u>	<u>0.159</u>
ConfCtrl (Ours)	✓	0.339	15.54	0.328	0.143	0.103	0.354	17.27	0.480	0.210	0.155	0.287	16.49	0.427	0.195	0.149

Table 1: Comparison of methods on three datasets. **Best** and **second best** results are highlighted. For input camera condition, Cam denotes camera parameters, and PC denotes projected point cloud.

	CO3D-Hydrant							DL3DV		
	TCE ↓	Subject ↑	Background ↑	Motion ↑	FID ↓	FD-DINOv2 ↓	FVD ↓	FID ↓	FD-DINOv2 ↓	FVD ↓
CameraCtrl [18]	0.164	0.857	0.921	0.931	127.82	221.23	146.26	93.36	263.29	125.90
ViewCrafter [71]	0.167	0.820	0.916	0.943	129.32	208.24	148.02	97.83	261.16	128.01
Uni3C [4]	0.148	0.842	0.932	0.945	124.53	202.31	140.98	89.21	233.07	118.64
Gen3R [21]	0.142	0.823	0.935	0.936	115.85	203.47	125.05	86.53	255.63	105.52
Ours	0.140	0.865	0.939	0.948	109.08	181.76	123.86	85.66	231.29	113.46

Table 2: Quantitative comparison on CO3D-Hydrant and DL3DV. Inter-frame consistency is evaluated using TCE and VBench metrics (Subject, Background, Motion), while generative quality is measured by FID, FD-DINOv2, and FVD.

Baselines. We compare our method with representative regression- and diffusion-based approaches trained on the same data. The regression baselines include PixelSplat [5], MvSplat [8], DepthSplat [67], NopoSplat [69], AnySplat [25], and LVSM [26]; the diffusion baselines include CameraCtrl [18], View-Crafter [71], and Uni3C [4]. Diffusion baselines are retrained in a video diffusion setting conditioned on the first and last frames, using the same base model as ours for fairness. For Uni3C and ViewCrafter, we use the same VGGT-projected point clouds as input. We additionally compare with large-scale pretrained diffusion models, SeVA [78] and Gen3R [21]. More details are provided in supplementary.

5.2 Comparison with Existing Methods

Quantitative results in Table 1 demonstrate that our method consistently outperforms all baseline approaches across all evaluation metrics. The improvements in reconstruction scores highlight the robustness of our approach, even under sparse input views and large viewpoint variations. Moreover, the reduced translation and rotation errors indicate enhanced controllability and precise camera-pose alignment, confirming that our model not only generates visually accurate results but also adheres more faithfully to the specified geometric constraints.

Additionally, we evaluate inter-frame consistency using Temporal Consistency Error (TCE), computed via optical flow-based frame warping, together with complementary VBench [24] metrics, including subject consistency, background consistency, and motion smoothness on CO3D-Hydrant dataset (Table 2).

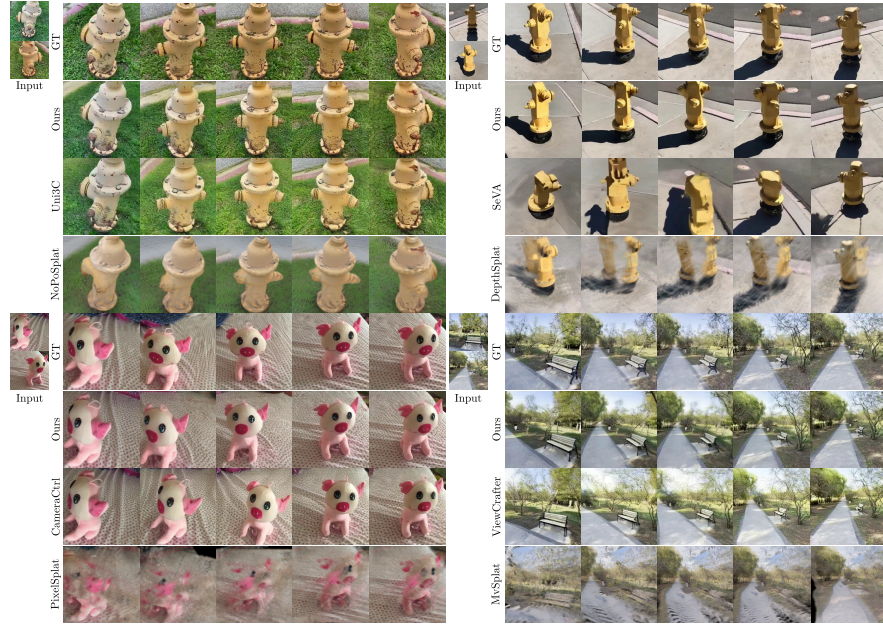


Fig. 3: Qualitative comparison with baseline methods on the CO3D-Hydrant, CO3D-Teddybear, and DL3DV datasets. The two input images exhibit a large viewpoint change, demonstrating the robustness of our method under challenging view variations.

To further assess generative capability, we additionally compare our method with diffusion-based approaches using three widely adopted generative quality metrics [53]: FID, FDDINOv2, and FVD. As summarized in Table 2, our method consistently outperforms existing approaches across the majority of evaluated metrics, demonstrating both superior inter-frame consistency and higher generative quality.

As shown in Figure 3, our method further produces novel views with finer details and sharper visual quality compared to regression-based approaches, which often suffer from inaccurate Gaussian parameter estimation. In contrast to diffusion-based methods, our approach exhibits superior consistency with the specified target camera pose, achieving geometrically accurate and structurally coherent results across views.

5.3 Cross Dataset Performance

Benefiting from the large-scale pretraining of video diffusion models, our method, along with other video-diffusion-based approaches, exhibits strong zero-shot generalization across diverse datasets compared to regression-based methods under large viewpoint changes. To rigorously evaluate cross-dataset performance, we conduct experiments on representative datasets including RealEstate10k [79],

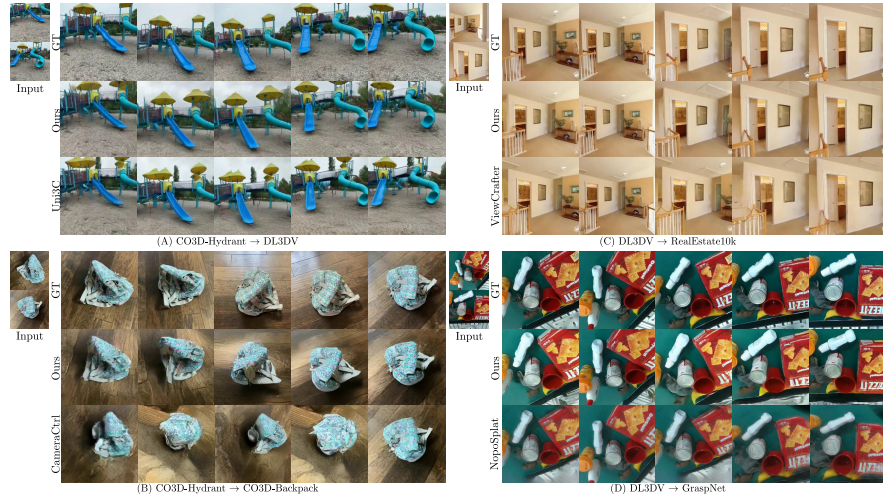


Fig. 4: Zero-shot results. Qualitative comparison on out-of-distribution datasets, demonstrating our strong generalization ability.

Method	CO3D-Hydrant → CO3D-Teddybear					CO3D-Hydrant → DL3DV					DL3DV → GraspNet				
	LPIPS ↓	PSNR ↑	SSIM ↑	E _t ↓	E _r ↓	LPIPS ↓	PSNR ↑	SSIM ↑	E _t ↓	E _r ↓	LPIPS ↓	PSNR ↑	SSIM ↑	E _t ↓	E _r ↓
NopoSplat [69]	0.656	14.13	0.293	1.125	0.949	0.687	10.01	0.172	1.099	0.878	0.659	10.87	0.242	0.222	0.157
DepthSplat [67]	0.675	12.48	0.290	1.218	1.032	0.659	10.65	0.201	1.079	0.854	0.621	10.02	0.189	0.202	0.143
MvSplat [8]	0.724	7.74	0.119	1.445	1.214	0.715	10.26	0.202	1.120	0.876	0.670	10.48	0.257	0.357	0.255
PixelSplat [5]	0.641	13.93	0.402	0.691	0.550	0.643	12.82	0.293	0.701	0.529	0.576	14.42	0.405	0.219	0.156
CameraCtrl [18]	0.460	15.24	0.392	0.560	0.433	0.408	14.21	0.311	0.392	0.309	0.490	11.77	0.369	0.220	0.156
ViewCrafter [71]	<u>0.423</u>	<u>16.03</u>	<u>0.408</u>	0.359	0.261	0.352	15.29	0.354	<u>0.296</u>	<u>0.234</u>	0.272	14.37	0.476	0.180	0.127
Uni3C [4]	0.439	15.26	0.405	<u>0.347</u>	<u>0.256</u>	<u>0.350</u>	<u>15.30</u>	<u>0.368</u>	0.317	0.249	<u>0.238</u>	<u>15.05</u>	<u>0.499</u>	<u>0.128</u>	<u>0.091</u>
ConfCtrl (Ours)	0.386	16.15	0.416	0.323	0.236	0.336	15.33	0.373	0.285	0.220	0.229	15.23	0.508	0.099	0.071

Table 3: Cross-dataset generalization results on CO3D, DL3DV, and GraspNet benchmarks. Best and second best results are highlighted.

GraspNet [13], CO3D [47], and DL3DV [37], which cover a wide range of scene types and motion distributions. Some quantitative comparisons are summarized in Table 3. Our approach achieves the highest overall performance across all datasets, demonstrating superior robustness under significant domain shifts. These results indicate that the learned geometric priors and pose-conditioned generation framework effectively transfer to unseen scenarios without fine-tuning.

As illustrated in Figure 4, our method produces visually coherent and geometrically accurate novel views on out-of-distribution datasets. The generated results preserve fine scene details and maintain consistent spatial layouts across varying camera poses, demonstrating strong generalization of both appearance and geometry. These qualitative examples further confirm that our model can effectively handle diverse scene structures and camera trajectories without dataset-specific adaptation.

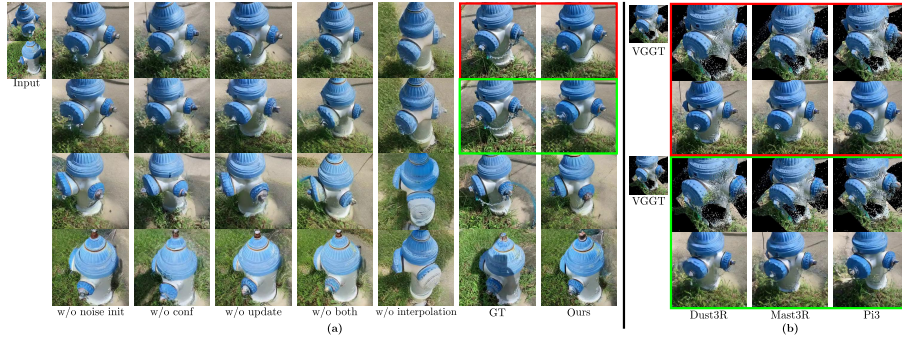


Fig. 5: Qualitative Ablation. (a) Visual comparison of different model configurations showing the effect of each component. (b) Ablation across 3D foundation models. Our results and the ground truth (GT) are boxed in matching colors in (a).

5.4 Ablation Study

Noise Initialization. First, we remove the proposed confidence-weighted noise initialization and adopt the standard rectified flow starting from pure Gaussian noise. As shown in Table 4(a) and Fig. 5(a), both novel-view quality and camera control accuracy degrade significantly, with noticeable distortions and warping artifacts.

Second, we evaluate the role of the confidence map by assigning a uniform weight of 1.0. Results in Table 4(b) show that this naive setting further reduces image quality compared to (a), although it yields slightly better camera controllability.

Kalman DiT Block. We remove the update submodule in each Kalman DiT block, making the model behave similarly to ControlNet [75], conditioned only on the camera pose and projected point cloud. Without this correction learning, the model struggles to balance the noisy 3D prior with the target camera pose, leading to noticeable degradation in both video quality and camera control accuracy (Fig. 5(a)), confirming that this update effectively refines the generation.

We further remove both the update submodule and the confidence-aware noise initialization. This variant yields the worst camera controllability among all settings, performing worse than (a), (b), and (c). The results indicate that Kalman DiT and confidence-weighted initialization are complementary, providing distinct benefits whose combination yields stronger performance.

Gradient Regulation Loss. Removing the gradient loss as a regularization term in Eq. 10 slightly degrades visual quality and camera control accuracy. While the gain is modest, this loss stabilizes training and improves spatiotemporal consistency.

Camera Pose Condition in Prediction Submodule. In this experiment, we remove the cross-attention module that uses the projected point cloud latent in the prediction step of Kalman DiT, leaving the model conditioned only on the

camera pose. The corresponding results are reported in Table 4(f). The observed performance drop confirms the importance of integrating both the camera pose and projected point cloud as complementary signals. While the simple addition of these features provides a minor improvement, it remains notably inferior to the cross-attention formulation (Table 4 (g)).

Interpolation Model. We ablate the initialization choice by starting from a pretrained camera-guided video diffusion model instead of an interpolation model. As shown in Table 4(h) and Fig. 5(a), the substantial performance decline verifies that the interpolation model is crucial for preserving geometric consistency in novel view synthesis.

Model Parameter Number. Our Kalman DiT block comprises three DiT blocks. To verify that improvements stem from the architecture rather than parameter scaling, we build a deeper variant of design (c) with 15 layers to match our model’s DiT count. Table 4(i) shows that naive scaling under the same training budget degrades performance, confirming the effectiveness of our design.

3D Foundation Model. To investigate the influence of different geometry foundation models, we replace VGGT in our methods with Dust3R [60], Mast3R [33], and Pi3 [61], respectively, providing projected point cloud and corresponding confidence map. Stronger geometry priors improve performance across all backbones while maintaining qualitative robustness (Table 4(j-l), Fig. 5(b)), confirming the effectiveness of our design for precise camera control in video diffusion conditioned on noisy projected point clouds.

Other architecture. MMDiT and Ctrl-xs [10, 72] enable bidirectional information flow between pose and geometry signals. We respectively replace our Kalman DiT block with an MMDiT block and a Ctrl-xs block using the same inputs: the camera-pose latent and the projected point-cloud latent. As shown in Table 4(m,n), our method outperforms MMDiT and Ctrl-xs, highlighting the advantage of our design. We attribute this gain to the update submodule, which refines latents using geometric observations alleviating uncertainty.

6 Conclusion

In this paper, we propose ConfCtrl, a framework built on a pretrained video interpolation model. ConfCtrl (1) uses a confidence-weighted point-cloud projection latent to initialize denoising, and (2) employs a predict-update camera conditioning mechanism to enable precise camera control and improve visual quality under noisy point-cloud conditions. Extensive experiments show that our approach outperforms both regression- and diffusion-based baselines across multiple benchmarks. On out-of-distribution datasets, it also demonstrates strong zero-shot performance.

Limitations: Despite these advantages, our model remains constrained by the limitations of the current VAE used in the video diffusion paradigm. Existing VAEs are primarily designed to generate temporally consistent, smooth con-

	LPIPS ↓	PSNR ↑	SSIM ↑	E _t ↓	E _r ↓
(a) w/o noise init	0.347	15.44	0.326	0.174	0.128
(b) w/o conf	0.366	14.75	0.296	0.164	0.117
(c) w/o update	0.386	14.65	0.305	0.194	0.139
(d) w/o both	0.362	14.85	0.304	0.213	0.155
(e) w/o L_{grad}	0.339	15.49	0.322	0.152	0.110
(f) camera pose only	0.370	14.90	0.312	0.192	0.141
(g) camera pose + proj.	0.363	15.06	0.313	0.174	0.124
(h) w/o interpolation	0.491	13.45	0.193	0.457	0.362
(i) param number	0.382	14.56	0.306	0.202	0.145
(j) VGGT → Dust3R	0.369	14.99	0.299	0.241	0.196
(k) VGGT → Mast3R	0.356	15.07	0.314	0.207	0.161
(l) VGGT → Pi3	0.333	15.56	0.326	0.146	0.107
(m) Kalman-DiT → MMDiT	0.366	15.01	0.314	0.194	0.140
(n) Kalman-DiT → Ctrl-xs	0.406	14.63	0.289	0.197	0.149
Ours	0.339	15.54	0.328	0.143	0.103

Table 4: Ablation study of model components on CO3D-Hydrant dataset.

tent between adjacent frames, making them less suitable for novel view synthesis involving abrupt camera motion or large positional changes. Future work may explore optimizing the VAE architecture for this setting or eliminating the VAE [51, 77] from novel view generation.

Acknowledgements

We thank the reviewers for their valuable comments and suggestions that helped improve this paper. We are grateful to our colleagues and collaborators for their insightful discussions and feedback throughout this work.

References

1. Bahmani, S., Skorokhodov, I., Qian, G., Siarohin, A., Menapace, W., Tagliasacchi, A., Lindell, D.B., Tulyakov, S.: Ac3d: Analyzing and improving 3d camera control in video diffusion transformers (2025), <https://arxiv.org/abs/2411.18673>
2. Bai, J., Xia, M., Fu, X., Wang, X., Mu, L., Cao, J., Liu, Z., Hu, H., Bai, X., Wan, P., Zhang, D.: Recammaster: Camera-controlled generative rendering from a single video (2025), <https://arxiv.org/abs/2503.11647>
3. Bose, S., Dutta, A., Nag, S., Zhang, J., Li, J., Karydis, K., Chowdhury, A.K.R.: Uncertainty-aware diffusion guided refinement of 3d scenes (2025), <https://arxiv.org/abs/2503.15742>
4. Cao, C., Zhou, J., Li, S., Liang, J., Yu, C., Wang, F., Xue, X., Fu, Y.: Uni3c: Unifying precisely 3d-enhanced camera and human motion controls for video generation. arXiv preprint arXiv:2504.14899 (2025)

5. Charatan, D., Li, S.L., Tagliasacchi, A., Sitzmann, V.: 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)
6. Chen, L., Zhou, Z., Zhao, M., Wang, Y., Zhang, G., Huang, W., Sun, H., Wen, J.R., Li, C.: Flexworld: Progressively expanding 3d scenes for flexible-view synthesis (2025), <https://arxiv.org/abs/2503.13265>
7. Chen, X., Chen, Y., Xiu, Y., Geiger, A., Chen, A.: Ttt3r: 3d reconstruction as test-time training. arXiv preprint arXiv:2509.26645 (2025)
8. Chen, Y., Xu, H., Zheng, C., Zhuang, B., Pollefeys, M., Geiger, A., Cham, T.J., Cai, J.: MVSplat: Efficient 3D Gaussian Splatting from Sparse Multi-view Images, p. 370–386. Springer Nature Switzerland (Oct 2024). https://doi.org/10.1007/978-3-031-72664-4_21, http://dx.doi.org/10.1007/978-3-031-72664-4_21
9. Chen, Y., Zheng, C., Xu, H., Zhuang, B., Vedaldi, A., Cham, T.J., Cai, J.: Mvsplat360: Feed-forward 360 scene synthesis from sparse views (2024), <https://arxiv.org/abs/2411.04924>
10. Esser, P., Kulal, S., Blattmann, A., Entezari, R., Müller, J., Saini, H., Levi, Y., Lorenz, D., Sauer, A., Boesel, F., et al.: Scaling rectified flow transformers for high-resolution image synthesis. In: Forty-first international conference on machine learning (2024)
11. Ewen, P., Chen, H., Isaacson, S., Wilson, J., Skinner, K.A., Vasudevan, R.: These magic moments: Differentiable uncertainty quantification of radiance field models (2025), <https://arxiv.org/abs/2503.14665>
12. Fan, Z., Cong, W., Wen, K., Wang, K., Zhang, J., Ding, X., Xu, D., Ivanovic, B., Pavone, M., Pavlakos, G., et al.: Instantsplat: Sparse-view gaussian splatting in seconds. arXiv preprint arXiv:2403.20309 (2024)
13. Fang, H.S., Gou, M., Wang, C., Lu, C.: Robust grasping across diverse sensor qualities: The graspnet-1billion dataset. The International Journal of Robotics Research (2023)
14. Gao, R., Holynski, A., Henzler, P., Brussee, A., Martin-Brualla, R., Srinivasan, P., Barron, J.T., Poole, B.: Cat3d: Create anything in 3d with multi-view diffusion models. arXiv preprint arXiv:2405.10314 (2024)
15. Go, H., Park, B., Jang, J., Kim, J.Y., Kwon, S., Kim, C.: Splatflow: Multi-view rectified flow model for 3d gaussian splatting synthesis. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 21524–21536 (2025)
16. Go, H., Park, B., Nam, H., Kim, B.H., Chung, H., Kim, C.: Videorfisplat: Direct scene-level text-to-3d gaussian splatting generation with flexible pose and multi-view joint modeling. arXiv preprint arXiv:2503.15855 (2025)
17. Han, C., Dumery, C.: View-dependent uncertainty estimation of 3d gaussian splatting (2025), <https://arxiv.org/abs/2504.07370>
18. He, H., Xu, Y., Guo, Y., Wetzstein, G., Dai, B., Li, H., Yang, C.: Cameractrl: Enabling camera control for text-to-video generation (2025), <https://arxiv.org/abs/2404.02101>
19. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020)
20. Hong, Y., Zhang, K., Gu, J., Bi, S., Zhou, Y., Liu, D., Liu, F., Sunkavalli, K., Bui, T., Tan, H.: Lrm: Large reconstruction model for single image to 3d. arXiv preprint arXiv:2311.04400 (2023)
21. Huang, J., Yang, Y., Yang, B., Ma, L., Ma, Y., Liao, Y.: Gen3r: 3d scene generation meets feed-forward reconstruction. arXiv preprint arXiv:2601.04090 (2026)

22. Huang, R., Mikolajczyk, K.: No pose at all: Self-supervised pose-free 3d gaussian splatting from sparse views. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 27947–27957 (2025)
23. Huang, T., Zheng, W., Wang, T., Liu, Y., Wang, Z., Wu, J., Jiang, J., Li, H., Lau, R.W., Zuo, W., et al.: Voyager: Long-range and world-consistent video diffusion for explorable 3d scene generation. arXiv preprint arXiv:2506.04225 (2025)
24. Huang, Z., He, Y., Yu, J., Zhang, F., Si, C., Jiang, Y., Zhang, Y., Wu, T., Jin, Q., Chanpaisit, N., et al.: Vbench: Comprehensive benchmark suite for video generative models. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 21807–21818 (2024)
25. Jiang, L., Mao, Y., Xu, L., Lu, T., Ren, K., Jin, Y., Xu, X., Yu, M., Pang, J., Zhao, F., Lin, D., Dai, B.: Anysplat: Feed-forward 3d gaussian splatting from unconstrained views (2025), <https://arxiv.org/abs/2505.23716>
26. Jin, H., Jiang, H., Tan, H., Zhang, K., Bi, S., Zhang, T., Luan, F., Snavely, N., Xu, Z.: Lvsm: A large view synthesis model with minimal 3d inductive bias (2025), <https://arxiv.org/abs/2410.17242>
27. Kalman, R.E.: A new approach to linear filtering and prediction problems (1960)
28. Kang, G., Yoo, J., Park, J., Nam, S., Im, H., Shin, S., Kim, S., Park, E.: Selfsplat: Pose-free and 3d prior-free generalizable 3d gaussian splatting. In: Proceedings of the Computer Vision and Pattern Recognition Conference. pp. 22012–22022 (2025)
29. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering (2023), <https://arxiv.org/abs/2308.04079>
30. Kim, M., Lim, J., Han, B.: 4d gaussian splatting in the wild with uncertainty-aware regularization (2024), <https://arxiv.org/abs/2411.08879>
31. Klasson, M., Mereu, R., Kannala, J., Solin, A.: Sources of uncertainty in 3d scene reconstruction (2024), <https://arxiv.org/abs/2409.06407>
32. Kong, X., Watson, D., Strümpfer, Y., Niemeyer, M., Tombari, F.: Causnvs: Autoregressive multi-view diffusion for flexible 3d novel view synthesis. arXiv preprint arXiv:2509.06579 (2025)
33. Leroy, V., Cabon, Y., Revaud, J.: Grounding image matching in 3d with mast3r (2024), <https://arxiv.org/abs/2406.09756>
34. Li, J., Tan, H., Zhang, K., Xu, Z., Luan, F., Xu, Y., Hong, Y., Sunkavalli, K., Shakhnarovich, G., Bi, S.: Instant3d: Fast text-to-3d with sparse-view generation and large reconstruction model. arXiv preprint arXiv:2311.06214 (2023)
35. Li, T., Zheng, G., Jiang, R., Zhan, S., Wu, T., Lu, Y., Lin, Y., Deng, C., Xiong, Y., Chen, M., Cheng, L., Li, X.: Realcam-i2v: Real-world image-to-video generation with interactive complex camera control (2025), <https://arxiv.org/abs/2502.10059>
36. Liang, H., Cao, J., Goel, V., Qian, G., Korolev, S., Terzopoulos, D., Plataniotis, K.N., Tulyakov, S., Ren, J.: Wonderland: Navigating 3d scenes from a single image (2025), <https://arxiv.org/abs/2412.12091>
37. Ling, L., Sheng, Y., Tu, Z., Zhao, W., Xin, C., Wan, K., Yu, L., Guo, Q., Yu, Z., Lu, Y., Li, X., Sun, X., Ashok, R., Mukherjee, A., Kang, H., Kong, X., Hua, G., Zhang, T., Benes, B., Bera, A.: D13dv-10k: A large-scale scene dataset for deep learning-based 3d vision (2023), <https://arxiv.org/abs/2312.16256>
38. Lipman, Y., Chen, R.T., Ben-Hamu, H., Nickel, M., Le, M.: Flow matching for generative modeling. arXiv preprint arXiv:2210.02747 (2022)
39. Liu, F., Sun, W., Wang, H., Wang, Y., Sun, H., Ye, J., Zhang, J., Duan, Y.: Reconx: Reconstruct any scene from sparse views with video diffusion model. arXiv preprint arXiv:2408.16767 (2024)

40. Liu, J., Wang, G., Ye, W., Jiang, C., Han, J., Liu, Z., Zhang, G., Du, D., Wang, H.: Diffflow3d: Toward robust uncertainty-aware scene flow estimation with diffusion model (2024), <https://arxiv.org/abs/2311.17456>
41. Liu, L., Li, W., Zhang, D., Wang, S., Jiao, S.: Idcnet: Guided video diffusion for metric-consistent rgbd scene generation with precise camera control (2025), <https://arxiv.org/abs/2508.04147>
42. Liu, X., Zhou, C., Huang, S.: 3dgs-enhancer: Enhancing unbounded 3d gaussian splatting with view-consistent 2d diffusion priors (2024), <https://arxiv.org/abs/2410.16266>
43. Liu, X., Gong, C., Liu, Q.: Flow straight and fast: Learning to generate and transfer data with rectified flow. arXiv preprint arXiv:2209.03003 (2022)
44. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. *Communications of the ACM* **65**(1), 99–106 (2021)
45. Min, Z., Luo, Y., Sun, J., Yang, Y.: Epipolar-free 3d gaussian splatting for generalizable novel view synthesis (2024), <https://arxiv.org/abs/2410.22817>
46. Peebles, W., Xie, S.: Scalable diffusion models with transformers. In: *Proceedings of the IEEE/CVF international conference on computer vision*. pp. 4195–4205 (2023)
47. Reizenstein, J., Shapovalov, R., Henzler, P., Sbordon, L., Labatut, P., Novotny, D.: Common objects in 3d: Large-scale learning and evaluation of real-life 3d category reconstruction. In: *Proceedings of the IEEE/CVF international conference on computer vision*. pp. 10901–10911 (2021)
48. Ren, X., Shen, T., Huang, J., Ling, H., Lu, Y., Nimier-David, M., Müller, T., Keller, A., Fidler, S., Gao, J.: 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)
49. Savant, L., Valsesia, D., Magli, E.: Modeling uncertainty for gaussian splatting (2024), <https://arxiv.org/abs/2403.18476>
50. Schwarz, K., Mueller, N., Kotschieder, P.: Generative gaussian splatting: Generating 3d scenes with video diffusion priors. arXiv preprint arXiv:2503.13272 (2025)
51. Shi, M., Wang, H., Zheng, W., Yuan, Z., Wu, X., Wang, X., Wan, P., Zhou, J., Lu, J.: Latent diffusion model without variational autoencoder. arXiv preprint arXiv:2510.15301 (2025)
52. Smart, B., Zheng, C., Laina, I., Prisacariu, V.A.: Splatt3r: Zero-shot gaussian splatting from uncalibrated image pairs (2024), <https://arxiv.org/abs/2408.13912>
53. Stein, G., Cresswell, J., Hosseinzadeh, R., Sui, Y., Ross, B., Vilecroze, V., Liu, Z., Caterini, A.L., Taylor, E., Loaiza-Ganem, G.: Exposing flaws of generative model evaluation metrics and their unfair treatment of diffusion models. *Advances in Neural Information Processing Systems* **36**, 3732–3784 (2023)
54. Sun, W., Chen, S., Liu, F., Chen, Z., Duan, Y., Zhang, J., Wang, Y.: Dimensionx: Create any 3d and 4d scenes from a single image with controllable video diffusion. arXiv preprint arXiv:2411.04928 (2024)
55. Tan, Z., Chen, X., Zhang, J., Feng, L., Hu, D.: Uncertainty-aware normal-guided gaussian splatting for surface reconstruction from sparse image sequences (2025), <https://arxiv.org/abs/2503.11172>
56. Wan, T., Wang, A., Ai, B., Wen, B., Mao, C., Xie, C.W., Chen, D., Yu, F., Zhao, H., Yang, J., et al.: Wan: Open and advanced large-scale video generative models. arXiv preprint arXiv:2503.20314 (2025)

57. Wang, F., Xu, Q., Ong, Y.S., Pollefeys, M.: Lightweight and accurate multi-view stereo with confidence-aware diffusion model. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2025)
58. Wang, J., Zheng, Z., Xu, W., Liu, P.: Rigi: Rectifying image-to-3d generation inconsistency via uncertainty-aware learning (2024), <https://arxiv.org/abs/2411.18866>
59. Wang, J., Chen, M., Karaev, N., Vedaldi, A., Rupprecht, C., Novotny, D.: Vggt: Visual geometry grounded transformer. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. pp. 5294–5306 (2025)
60. Wang, S., Leroy, V., Cabon, Y., Chidlovskii, B., Revaud, J.: Dust3r: Geometric 3d vision made easy (2024), <https://arxiv.org/abs/2312.14132>
61. Wang, Y., Zhou, J., Zhu, H., Chang, W., Zhou, Y., Li, Z., Chen, J., Pang, J., Shen, C., He, T.: π^3 : Permutation-equivariant visual geometry learning (2025), <https://arxiv.org/abs/2507.13347>
62. Wang, Z., Yuan, Z., Wang, X., Chen, T., Xia, M., Luo, P., Shan, Y.: Motionctrl: A unified and flexible motion controller for video generation (2024), <https://arxiv.org/abs/2312.03641>
63. Wewer, C., Raj, K., Ilg, E., Schiele, B., Lenssen, J.E.: latentsplat: Autoencoding variational gaussians for fast generalizable 3d reconstruction (2024), <https://arxiv.org/abs/2403.16292>
64. Wilson, J., Almeida, M., Sun, M., Mahajan, S., Ghaffari, M., Ewen, P., Ghahemalizadeh, O., Kuo, C.H., Sen, A.: Modeling uncertainty in 3d gaussian splatting through continuous semantic splatting (2024), <https://arxiv.org/abs/2411.02547>
65. Wu, R., Gao, R., Poole, B., Trevithick, A., Zheng, C., Barron, J.T., Holynski, A.: Cat4d: Create anything in 4d with multi-view video diffusion models. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. pp. 26057–26068 (2025)
66. Wu, S., Xu, C., Huang, B., Geiger, A., Chen, A.: Genfusion: Closing the loop between reconstruction and generation via videos. In: *Proceedings of the Computer Vision and Pattern Recognition Conference*. pp. 6078–6088 (2025)
67. Xu, H., Peng, S., Wang, F., Blum, H., Barath, D., Geiger, A., Pollefeys, M.: Depth-splat: Connecting gaussian splatting and depth (2025), <https://arxiv.org/abs/2410.13862>
68. Yang, L., Kang, B., Huang, Z., Zhao, Z., Xu, X., Feng, J., Zhao, H.: Depth anything v2. *Advances in Neural Information Processing Systems* **37**, 21875–21911 (2024)
69. Ye, B., Liu, S., Xu, H., Li, X., Pollefeys, M., Yang, M.H., Peng, S.: No pose, no problem: Surprisingly simple 3d gaussian splats from sparse unposed images. *arXiv preprint arXiv:2410.24207* (2024)
70. YU, M., Hu, W., Xing, J., Shan, Y.: Trajectorycrafter: Redirecting camera trajectory for monocular videos via diffusion models (2025), <https://arxiv.org/abs/2503.05638>
71. Yu, W., Xing, J., Yuan, L., Hu, W., Li, X., Huang, Z., Gao, X., Wong, T.T., Shan, Y., Tian, Y.: Viewcrafter: Taming video diffusion models for high-fidelity novel view synthesis (2024), <https://arxiv.org/abs/2409.02048>
72. Zavadski, D., Feiden, J.F., Rother, C.: Controlnet-xs: Rethinking the control of text-to-image diffusion models as feedback-control systems. In: *European Conference on Computer Vision*. pp. 343–362. Springer (2024)
73. Zhang, C., Zou, Y., Li, Z., Yi, M., Wang, H.: Transplat: Generalizable 3d gaussian splatting from sparse multi-view images with transformers (2024), <https://arxiv.org/abs/2408.13770>

74. Zhang, K., Bi, S., Tan, H., Xiangli, Y., Zhao, N., Sunkavalli, K., Xu, Z.: Gs-lrm: Large reconstruction model for 3d gaussian splatting. In: European Conference on Computer Vision. pp. 1–19. Springer (2024)
75. Zhang, L., Rao, A., Agrawala, M.: Adding conditional control to text-to-image diffusion models. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3836–3847 (2023)
76. Zhang, S., Xu, H., Guo, S., Xie, Z., Bao, H., Xu, W., Zou, C.: Spatialcrafter: Unleashing the imagination of video diffusion models for scene reconstruction from limited observations. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 27794–27805 (2025)
77. Zheng, B., Ma, N., Tong, S., Xie, S.: Diffusion transformers with representation autoencoders. arXiv preprint arXiv:2510.11690 (2025)
78. Zhou, J.J., Gao, H., Voleti, V., Vasishta, A., Yao, C.H., Boss, M., Torr, P., Rupprecht, C., Jampani, V.: Stable virtual camera: Generative view synthesis with diffusion models. arXiv preprint arXiv:2503.14489 (2025)
79. Zhou, T., Tucker, R., Flynn, J., Fyffe, G., Snavely, N.: Stereo magnification: Learning view synthesis using multiplane images. arXiv preprint arXiv:1805.09817 (2018)
80. Ziwen, C., Tan, H., Zhang, K., Bi, S., Luan, F., Hong, Y., Fuxin, L., Xu, Z.: Long-lrm: Long-sequence large reconstruction model for wide-coverage gaussian splats. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 4349–4359 (2025)