ON-POLICY DISTILLATION OF VIDEO AND MULTI-VIEW PRIORS FOR MULTI-VIEW VIDEO GENERATION

Anonymous authors
Paper under double-blind review

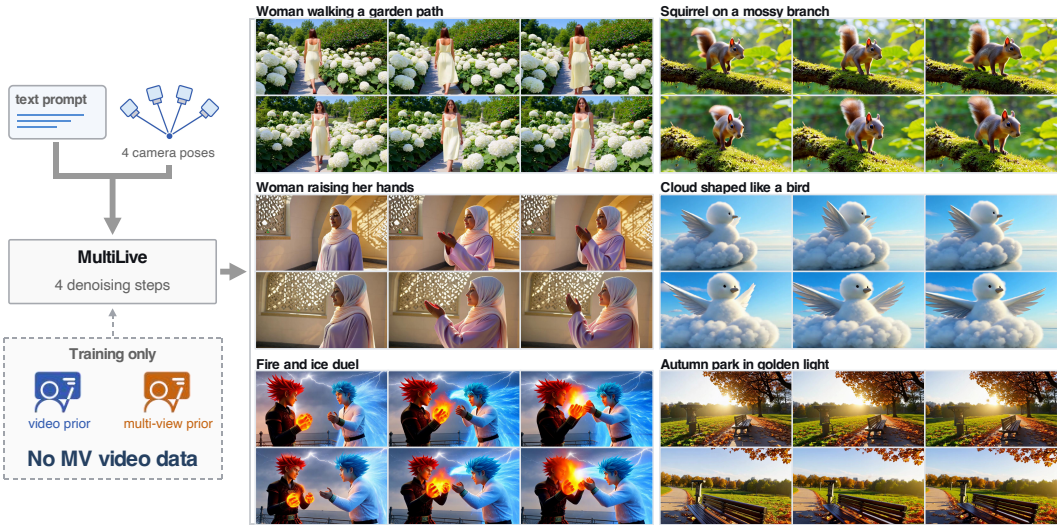


Figure 1: **Multi-view video generation without multi-view video training data.** Six scenes generated by MULTILIVE from text prompts, covering diverse content and styles. Video and multi-view teachers guide motion and geometry only during training. MULTILIVE jointly generates four synchronized views in 4 denoising steps; only two views are shown here due to space constraints.

ABSTRACT

Multi-view video generation aims to synthesize synchronized videos of the same dynamic scene from different viewpoints. Direct generation methods face a shortage of synchronized training data, while combining monocular videos and static multi-view images provides only separate supervision for motion and geometry. Geometry-guided methods rely on estimated depth or 3D proxies, whose errors can compromise cross-view consistency. We propose MULTILIVE, an on-policy dual-teacher distillation framework that combines separately learned video and multi-view priors in a single generator. Our key idea is to concurrently match the temporal and multi-view distributions of the student's own generations to these complementary priors. The video teacher guides motion within each monocular video, while the multi-view teacher guides geometry and appearance across synchronized views. Both distribution-matching objectives update the student using the same generated sample, encouraging the two capabilities to remain compatible as the scene evolves without explicit 3D reconstruction. Extensive experiments demonstrate that MULTILIVE achieves high visual quality, strong cross-view consistency, and coherent temporal dynamics without multi-view video training data, using only 4 denoising steps at inference.

1 INTRODUCTION

Multi-view video, often represented as $\{\mathcal{I}_{i,j} \mid i = 1, \ldots, V,\ j = 1, \ldots, T_{\text{rgb}}\}$, describes synchronized videos of the same dynamic scene from different viewpoints. Here, V denotes the number

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.

of viewpoints, and T_{rgb} denotes the number of RGB frames. This requires both cross-view and temporal consistency. At each time step, the scene's geometry and appearance should be consistent across views. Within each video, motion should be coherent, and objects should maintain a consistent appearance and identity over time. Such consistency is essential for applications such as virtual cinematography, dynamic 3D/4D content creation, and embodied world models (Chen et al., 2026b; Yin et al., 2025; Wu et al., 2025a; Sun et al., 2026a; Savva et al., 2026).

Existing approaches generally fall into two categories. *Joint view–time diffusion models* represent all camera streams as a $T_{\mathrm{rgb}} \times V$ grid and couple them through cross-view attention or synchronization layers, jointly modeling correspondence across cameras and dynamics within each stream (Bai et al., 2024; Xu et al., 2024a; Wang et al., 2025a; Li et al., 2024; Yang et al., 2025; Sun et al., 2026b). However, synchronized multi-view video data are extremely scarce and difficult to acquire for most real-world scenes. Consequently, methods directly fine-tuning pretrained video models on limited synthetic multi-view video datasets often lead to restricted scene diversity. Another branch, *geometry-guided methods*, adopts a cascaded pipeline: it first generates a monocular video $\{\mathcal{I}_{1,j} \mid j = 1, \ldots, T_{\mathrm{rgb}}\}$, estimates its depth to construct a coarse 3D proxy, and then renders or warps the proxy into the target viewpoints for a video diffuser to fill disocclusions and refine appearance (Park et al., 2026; Jeong et al., 2025; Ren et al., 2025; Fiebelman et al., 2026). In this procedure, errors in the estimated depth or 3D proxy can project content to incorrect image locations, while occlusions and non-rigid motion leave newly visible regions without reliable observations, forcing the diffusion model to hallucinate their appearance and weakening cross-view consistency.

To alleviate the scarcity of synchronized multi-view video data, several methods (Bai et al., 2024; Wu et al., 2025b) augment training with multi-view images and monocular videos. Conceptually, this strategy collapses the temporal dimension T_{rgb} and the viewpoint dimension V of a predicted multi-view video, and thereby replaces supervision of a full multi-view video with supervision of unpaired videos and static images. However, we argue that such data augmentation introduces a fundamental training–inference mismatch: during training, the model is optimized on two separately sampled sources that never form a multi-view video, whereas at inference time it is required to generate a full multi-view video in one pass. In our experiments, this approach produces pronounced flickering and geometric distortions, degrading video quality and cross-view consistency.

We introduce MULTILIVE, whose core idea is to replace the flow-matching loss (Lipman et al., 2023) on unpaired videos and static images with a distribution-matching objective, so that both teachers supervise the same generated multi-view video. Specifically, we propose a dual-teacher distribution-matching framework that distills separately learned video priors (e.g., Wan (Wan et al., 2025)) and multi-view priors (e.g., CAT3D (Gao* et al., 2024)) into a unified multi-view video generator. At each training step, the student model samples multi-view videos in an on-policy manner. The video teacher regularizes the temporal dynamics of each individual view, while the multi-view teacher provides supervision on cross-view consistency at each time step. This design enables every generated multi-view video to satisfy both priors simultaneously, even without access to synchronized multi-view video data, thereby avoiding the train–inference mismatch.

To support this dual-teacher learning, we introduce a shared fake-score model that provides the score estimates for the two teacher objectives through two branch-specific forward passes, each paired with its teacher's target distribution. We further train the multi-view model on synchronized multi-view images encoded with the causal Wan VAE at all latent positions, enabling supervision for both image-latent and video-latent student slices. Together, these designs enable temporally coherent and geometrically consistent multi-view video generation without requiring synchronized multi-view videos for training

Extensive experiments demonstrate that MULTILIVE improves visual quality and cross-view synchronization while maintaining temporal coherence. The model exhibits strong generalization across diverse real-world scenes and subjects, producing photorealistic videos with natural motion and strong text–video alignment. These results are achieved without multi-view video training data, with a single distilled generator producing all views jointly in only 4 denoising steps.

Our contributions can be summarized as follows:

- We propose MULTILIVE, which combines separately learned video and multi-view priors to generate multi-view videos without synchronized multi-view video training data.

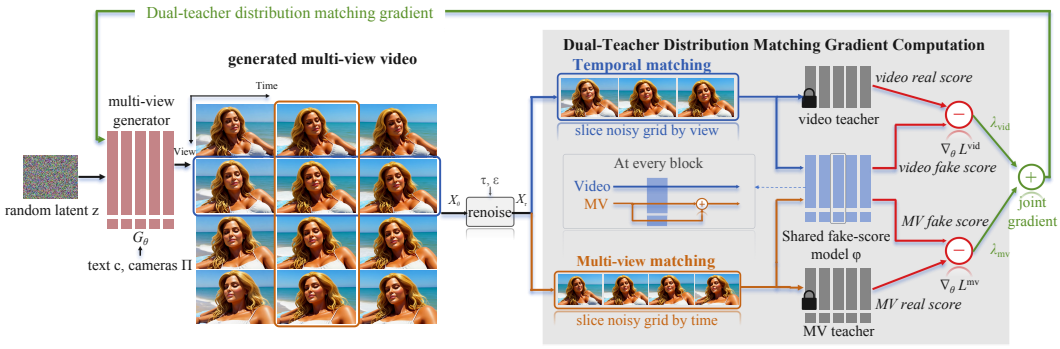


Figure 2: **Overview of MULTILIVE.** A camera-aware generator produces a shared multi-view latent grid. The video and multi-view teachers process per-view temporal trajectories and synchronized view slices, respectively, while a shared fake-score model estimates the student scores.

- We introduce an on-policy dual-teacher distribution-matching objective. The two teachers supervise monocular videos and synchronized multi-view images extracted from the same student-generated sample, encouraging temporal and cross-view consistency.
- Experiments demonstrate improved visual quality and cross-view consistency while maintaining temporal coherence, with all views generated jointly in only 4 denoising steps.

2 METHOD

We aim to enable multi-view video generation through dual-teacher distribution matching along the spatial and temporal dimensions, respectively. To this end, building upon distribution matching distillation (Section 2.1), we propose MULTILIVE, which integrates two complementary distribution priors into one unified generative model (Section 2.2): one for temporal coherence and dynamics, denoted as $p^{\text{vid}}_{\mathsf{T},\tau}(\cdot \mid \mathbf{c})$, and the other for cross-view consistency, denoted as $p^{\text{mv}}_{\mathsf{T},\tau}(\cdot \mid \mathbf{c}, \mathbf{\Pi})$. Here T denotes the teacher model, $\mathbf{c}$ represents the text condition, and $\mathbf{\Pi}$ denotes the camera condition.

2.1 PRELIMINARIES: DISTRIBUTION MATCHING DISTILLATION

Distribution matching distillation (DMD) (Yin et al., 2024b;a) trains a few-step generator by matching the distribution of its noised outputs to that of a frozen diffusion teacher, rather than imitating individual denoising trajectories. Let $\mathbf{c}$ denote the conditioning signal. Given $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$, the student generator G_θ produces $\hat{\mathbf{x}}_0 = G_\theta(\mathbf{z}, \mathbf{c})$, which is perturbed as

$$\mathbf{x}_\tau = \Psi_\tau(\hat{\mathbf{x}}_0, \boldsymbol{\epsilon}) = \alpha_\tau \hat{\mathbf{x}}_0 + \sigma_\tau \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}). \tag{1}$$

For our rectified-flow teachers, $(\alpha_\tau, \sigma_\tau) = (1 - \tau, \tau)$.

For fixed τ and $\mathbf{c}$, let $p_{\theta,\tau}(\cdot \mid \mathbf{c})$ denote the distribution of $\mathbf{x}_\tau$ induced by the current generator and the perturbation noise, and let $p_{\mathsf{T},\tau}(\cdot \mid \mathbf{c})$ denote the corresponding noised distribution of the frozen teacher. Because $\hat{\mathbf{x}}_0$ is sampled from the current generator, DMD is on-policy: the fake-score model and the DMD gradient are evaluated on the student's current outputs. This keeps the distillation signal aligned with the student's evolving distribution and reduces train–inference mismatch. DMD minimizes the reverse KL

$$\mathcal{L}_{\text{DMD}} = \text{KL}\big(p_{\theta,\tau}(\cdot \mid \mathbf{c}) \,\|\, p_{\mathsf{T},\tau}(\cdot \mid \mathbf{c})\big), \tag{2}$$

with τ sampled during training. In the score formulation,

$$\nabla_\theta \mathcal{L}_{\text{DMD}} \simeq \mathbb{E}\left[w_\tau \alpha_\tau \left(s_\phi(\mathbf{x}_\tau, \tau, \mathbf{c}) - s_{\mathsf{T}}(\mathbf{x}_\tau, \tau, \mathbf{c}) \right)^\top \frac{\partial \hat{\mathbf{x}}_0}{\partial \theta} \right], \tag{3}$$

where s_{T} is the score of the frozen teacher distribution, s_ϕ is an online estimate of the score of the current student, and w_τ is a time-dependent DMD weight.

For a rectified-flow velocity prediction v_η, we obtain the denoised prediction and the corresponding score as

$$d_\eta = \mathbf{x}_\tau - \tau v_\eta, \qquad s_\eta(\mathbf{x}_\tau, \tau) = \frac{(1-\tau)d_\eta - \mathbf{x}_\tau}{\tau^2}.$$

The latter is trained on detached student samples with

$$\mathcal{L}_{\text{fake}} = \mathbb{E}\left[\left\|D_\phi(\mathbf{x}_\tau, \tau, \mathbf{c}) - \text{sg}[\hat{\mathbf{x}}_0]\right\|_2^2\right], \tag{4}$$

where $\text{sg}[\cdot]$ denotes stop-gradient. Here D_ϕ is the denoised prediction d_ϕ in the conversion above; teacher velocities are converted to d_T in the same way before mapping to scores. Training alternates between updating the student with the DMD gradient and updating the fake-score model with this denoising loss. In the next section, we apply this construction to the temporal and multi-view readouts of the same student-generated sample.

2.2 Dual-Teacher Distribution Matching over Spatio-Temporal Slices

Shared Student Sample. Following common practice in video generation (Wan et al., 2025), we represent a multi-view video in a compact latent space as $\hat{\mathbf{X}}_0 \equiv \{\hat{\mathbf{x}}_{s,v}\}_{s=1,v=1}^{T,V}$, where s indexes time, v indexes the viewpoint, and each $\hat{\mathbf{x}}_{s,v}$ contains the spatial latent grid at that time and viewpoint. Our goal is to train a generative model G_θ such that $\hat{\mathbf{X}}_0 = G_\theta(\mathbf{z}, \mathbf{c}, \mathbf{\Pi})$ in one on-policy rollout:

$$\hat{\mathbf{X}}_0 = G_\theta(\mathbf{z}, \mathbf{c}, \mathbf{\Pi}), \qquad \mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}),$$

As illustrated in Figure 2, the two teachers inspect different deterministic readouts of this shared sample. This brings the two separately learned priors together to jointly guide the student's own multi-view generations, which constitutes a fundamental distinction between our method and prior approaches that directly use multi-view images or monocular videos for data augmentation, e.g. (Bai et al., 2024).

Temporal and Multi-View Readouts. The generated multi-view video is first partitioned into different slices to accommodate the corresponding teacher priors. Define the video and multi-view readout operators by

$$R_{\text{vid}}(\mathbf{X}) = \{\mathbf{x}_{1:T,v}\}_{v=1}^{V}, \qquad R_{\text{mv}}(\mathbf{X}) = \{\mathbf{x}_{s,1:V}\}_{s=1}^{T}. \tag{5}$$

The first readout contains one temporal trajectory for each viewpoint, whereas the second contains one synchronized set of viewpoints for each latent time position.

Following the noising construction in Eq. equation 1, we perturb the shared rollout once:

$$\hat{\mathbf{X}}_\tau = \Psi_\tau(\hat{\mathbf{X}}_0, \boldsymbol{\epsilon}) = \alpha_\tau \hat{\mathbf{X}}_0 + \sigma_\tau \boldsymbol{\epsilon}, \qquad \boldsymbol{\epsilon} \sim \mathcal{N}(\mathbf{0}, \mathbf{I}).$$

Applying the same readout operators to this noised sample gives

$$\hat{\mathbf{X}}_\tau^{\text{vid}} = R_{\text{vid}}(\hat{\mathbf{X}}_\tau), \qquad \hat{\mathbf{X}}_\tau^{\text{mv}} = R_{\text{mv}}(\hat{\mathbf{X}}_\tau),$$

Branch-aware student scores. In DMD, the score estimator must describe the same noised distribution as the teacher score being matched. The two readouts here induce distributions in different spaces: $\hat{\mathbf{X}}_\tau^{\text{vid}}$ contains per-view temporal trajectories, whereas $\hat{\mathbf{X}}_\tau^{\text{mv}}$ contains synchronized view sets and uses camera conditioning. Their score fields therefore encode different dependencies and cannot be represented by one undifferentiated score (see Section 3.2 for ablation). We estimate the two fields separately, using $s_{\phi,\text{vid}}$ for the video readout and $s_{\phi,\text{mv}}$ for the multi-view readout, and match each estimate only to its corresponding teacher score.

Let $p_{\theta,\tau}^{\text{vid}}$ and $p_{\theta,\tau}^{\text{mv}}$ denote the noised distributions induced by the two branch-level readouts of the on-policy student. The corresponding teacher distributions are $p_{\mathsf{T},\tau}^{\text{vid}}(\cdot \mid \mathbf{c})$ and $p_{\mathsf{T},\tau}^{\text{mv}}(\cdot \mid \mathbf{c}, \mathbf{\Pi})$. We match these distributions with

$$\mathcal{L}_{\text{dual}} = \lambda_{\text{vid}}\,\text{KL}\big(p_{\theta,\tau}^{\text{vid}} \,\|\, p_{\mathsf{T},\tau}^{\text{vid}}\big) + \lambda_{\text{mv}}\,\text{KL}\big(p_{\theta,\tau}^{\text{mv}} \,\|\, p_{\mathsf{T},\tau}^{\text{mv}}\big). \tag{6}$$

Here the expectation over $\tau, \mathbf{c}, \mathbf{\Pi}$ is implicit, and λ_{vid} and λ_{mv} weight the two branches. Equation equation 6 matches the temporal and synchronized multi-view distributions induced by these

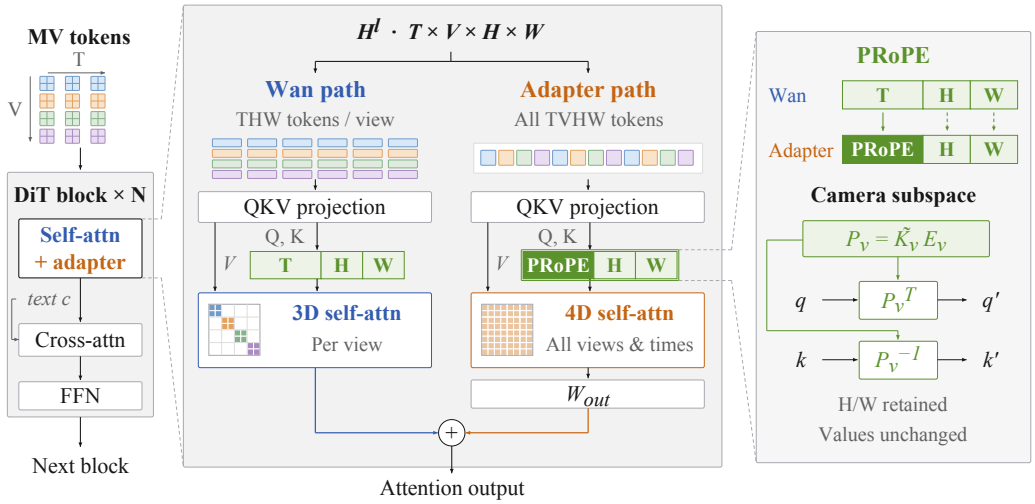


Figure 3: **Geometry-Aware Adapter.** The adapter augments Wan with joint spatio-temporal attention and camera-conditioned PRoPE while retaining Wan's per-view temporal path.

two readouts, rather than directly specifying a full joint distribution over all times and views. Both terms are evaluated on readouts of the same student-generated multi-view latent video and optimized on-policy using samples from the current generator, providing simultaneous feedback for temporal and cross-view compatibility on each generated sample.

In implementation, the corresponding branch-level score residuals are

$$\mathbf{r}_{\text{vid}} = s_{\phi,\text{vid}}(\hat{\mathbf{X}}_\tau^{\text{vid}}, \tau; \mathbf{c}) - s_{\mathsf{T},\text{vid}}(\hat{\mathbf{X}}_\tau^{\text{vid}}, \tau; \mathbf{c}), \tag{7}$$

$$\mathbf{r}_{\text{mv}} = s_{\phi,\text{mv}}(\hat{\mathbf{X}}_\tau^{\text{mv}}, \tau; \mathbf{c}, \mathbf{\Pi}) - s_{\mathsf{T},\text{mv}}(\hat{\mathbf{X}}_\tau^{\text{mv}}, \tau; \mathbf{c}, \mathbf{\Pi}). \tag{8}$$

Each residual is pulled back through its corresponding branch and applied to the same generator parameters:

$$\nabla_\theta \mathcal{L}_{\text{dual}} \simeq \mathbb{E}\left[\lambda_{\text{vid}} w_{\text{vid}} \alpha_\tau \mathbf{r}_{\text{vid}}^\top \frac{\partial \hat{\mathbf{X}}_0^{\text{vid}}}{\partial \theta} + \lambda_{\text{mv}} w_{\text{mv}} \alpha_\tau \mathbf{r}_{\text{mv}}^\top \frac{\partial \hat{\mathbf{X}}_0^{\text{mv}}}{\partial \theta}\right]. \tag{9}$$

Here w_{vid} and w_{mv} are effective branch-specific DMD weights, and α_τ is the signal coefficient of the noise schedule. Both residuals are taken from complementary readouts of the same $\hat{\mathbf{X}}_0$ and back-propagated to the same θ. Thus, each generator update receives simultaneous feedback from both branches, encouraging compatibility between the temporal and cross-view constraints.

Model Updating Mechanism. Training alternates between score-estimator updates and student updates. Given the current rollout and its two noised readouts, we detach the student outputs and update the branch-specific denoising predictors with

$$\mathcal{L}_{\text{fake}}^{\text{vid}} = \mathbb{E}\left[\left\| D_{\phi_{\text{vid}}}(\hat{\mathbf{X}}_\tau^{\text{vid}}, \tau, \mathbf{c}) - \text{sg}[\hat{\mathbf{X}}_0^{\text{vid}}] \right\|_2^2\right], \tag{10}$$

$$\mathcal{L}_{\text{fake}}^{\text{mv}} = \mathbb{E}\left[\left\| D_{\phi_{\text{mv}}}(\hat{\mathbf{X}}_\tau^{\text{mv}}, \tau, \mathbf{c}, \mathbf{\Pi}) - \text{sg}[\hat{\mathbf{X}}_0^{\text{mv}}] \right\|_2^2\right]. \tag{11}$$

The score estimators are updated by minimizing their respective losses while θ is held fixed. We then freeze both score estimators and the two teachers, and update θ with the dual DMD gradient above. After the student update, a fresh on-policy rollout is generated and the same two-step procedure is repeated, so the score estimates continue to track the student's evolving branch distributions. Section 2.3 describes how the two branch forwards are realized by the adapter-augmented network.

2.3 NETWORK ARCHITECTURE

To encode video frames and multi-view observations within a shared latent space for dual-teacher distillation, we adopt a geometry-aware adapter architecture. Specifically, we use a pretrained video diffusion model as the backbone and augment it with a geometry adapter dedicated to multi-view generation.

Geometry-Aware Adapter As shown in Figure 3, the Geometry-Aware Adapter realizes the two teacher roles within one adapter-augmented backbone. The original per-view Wan path preserves temporal modeling, while a camera-conditioned residual path adds cross-view interaction. At attention block ℓ, with features $\mathbf{H}^\ell$, the two execution modes are

$$\begin{aligned} \mathrm{Attn}^\ell_{\mathrm{vid}}(\mathbf{H}^\ell) &= \mathcal{A}^\ell_{\mathrm{Wan}}(\mathbf{H}^\ell), \\ \mathrm{Attn}^\ell_{\mathrm{mv}}(\mathbf{H}^\ell; \mathbf{\Pi}) &= \mathcal{A}^\ell_{\mathrm{Wan}}(\mathbf{H}^\ell) + \mathbf{W}^\ell_{\mathrm{out}} \mathcal{A}^\ell_{\mathrm{adapt}}(\mathbf{H}^\ell; \mathbf{\Pi}). \end{aligned} \tag{12}$$

Here $\mathcal{A}^\ell_{\mathrm{Wan}}$ operates on the per-view temporal layout, whereas $\mathcal{A}^\ell_{\mathrm{adapt}}$ attends over the joint $T \times V \times H \times W$ grid, with residual output projection $\mathbf{W}^\ell_{\mathrm{out}}$. Camera calibration enters only the adapter through PRoPE (Li et al., 2025). The video teacher and its paired score forward use the video-native mode; the multi-view teacher, its paired score forward, and the student use the multi-view-native mode. The two student-score forwards *share backbone parameters* through this adapter architecture while estimating scores separately in their respective readout spaces. PRoPE transformations and initialization are detailed in Appendix C.1.

Causal-Latent Multi-View Teacher The causal Wan VAE represents the first latent position as an image latent and later positions as video latents. A multi-view teacher trained only on synchronized image latents would therefore be mismatched with the later student readouts. We construct teacher inputs at every latent position using the native causal VAE.

For synchronized multi-view images $\mathcal{I} = \{\mathbf{I}_v\}_{v=1}^V$, we repeat each image along the temporal dimension, encode it with the causal VAE, and collect the latent at position q from all views as $\mathbf{Z}_q$, $q \in \{1, \ldots, T\}$. Thus, $\mathbf{Z}_1$ contains image latents, whereas $\mathbf{Z}_q$ for $q > 1$ contains video latents. We train the camera-conditioned adapter on these position-specific multi-view latents with the velocity flow-matching loss

$$\mathcal{L}_{\mathrm{MV}}(\psi) = \mathbb{E}\left[\| F_{\bar{\omega},\psi}(\Psi_\tau(\mathbf{Z}_q, \boldsymbol{\epsilon}), \tau; \mathbf{c}, \mathbf{\Pi}) - (\boldsymbol{\epsilon} - \mathbf{Z}_q) \|_2^2 \right]. \tag{13}$$

Here $\bar{\omega}$ denotes the frozen Wan backbone and ψ the trainable adapter. After training, we freeze $F_{\bar{\omega},\psi}$ as the multi-view teacher T_{mv}. Sampling all latent positions enables it to supervise both the first image-latent position and the later video-latent positions used by the student. Implementation and data-construction details are provided in Appendix C.2.

3 EXPERIMENTS

We first describe our implementation details and evaluation protocol. We then compare MULTI-LIVE with existing methods (Section 3.1), present ablation studies (Section 3.2), and explore an application to action-conditioned world modeling (Section 3.3).

Implementation Details. We build MULTILIVE on Wan2.1-1.3B (Wan et al., 2025). Its backbone serves as the video teacher, with its parameters kept frozen throughout training. To equip it with multi-view generation capability, we augment the backbone with a Geometry-Aware Adapter, as described in Section 2.3. The Geometry-Aware Adapter is trained on DL3DV (Ling et al., 2024) for 10,000 iterations. We then initialize both the student and fake-score model from this teacher and fully fine-tune them for 2,000 distillation iterations, keeping both teachers frozen. Both stages use eight A800 GPUs, with per-GPU batch sizes of four and one, respectively. The distilled student jointly generates all views in four denoising steps. Additional implementation details are provided in the supplementary materials.

Evaluation Setup. We evaluate all methods using the same 100 prompts from MovieGen-Bench (Polyak et al., 2024). For each prompt we randomly sample a four-view camera rig from the

Table 1: **Quantitative Comparison.** Multi-view video generation on 100 prompts, with four 81-frame views at 480P. NFE and latency are for a two-view pass. ReCamMaster and NeoVerse are video-to-video re-rendering methods, whereas SynCamMaster and MULTILIVE jointly generate views from text and camera poses.

	Per-View Quality				Camera Accuracy		View Synchronization			Efficiency	
Method	FID ↓	FVD ↓	CLIP-T ↑	CLIP-F ↑	RotErr ↓	TransErr ↓	Mat. Pix. (K) ↑	FVD-V ↓	CLIP-V ↑	NFE ↓	Lat. (s) ↓
ReCamMaster	108.25	117.91	28.59	98.77	8.72	18.69	477.39	25.83	97.13	50	998.40
NeoVerse	119.43	132.12	26.46	97.12	14.87	16.92	250.03	52.92	88.45	4	86.86
SynCamMaster	131.34	129.18	26.23	99.45	8.22	19.32	213.76	57.97	89.82	50	400.31
MULTILIVE	114.67	111.32	30.67	99.69	13.18	18.21	443.26	24.79	97.91	4	38.12

SynCamVideo validation set. MULTILIVE generates all four views jointly in one pass. Four-view evaluations use 81-frame views at 480P and 16 FPS. Details of camera sampling, pose conversion, and baseline-specific processing are provided in the supplementary materials.

Metrics. Following (Bai et al., 2024), we evaluate per-view quality and cross-view synchronization, and assess camera accuracy. For per-view quality, we evaluate FID (Heusel et al., 2018), FVD (Unterthiner et al., 2019), and CLIP (Radford et al., 2021) similarity through CLIP-T for frame–text alignment and CLIP-F for adjacent-frame consistency. For synchronization, we report Mat. Pix., the number of confidently matched pixels between synchronized views obtained using GIM (Shen et al., 2024), along with FVD-V (Xie et al., 2024; Yao et al., 2025) and CLIP-V (Kuang et al., 2024), which measures CLIP similarity across views at the same timestamp. For camera accuracy, we estimate camera poses using VGGT (Wang et al., 2025b) and report RotErr and TransErr (He et al., 2024). These measure the mean geodesic rotation error and mean angular error in translation direction, respectively, both in degrees. We also include VBench (Huang et al., 2024) scores in Table 2 and evaluate efficiency using the number of denoiser evaluations, denoted NFE, and generation latency, both reported for a two-view pass.

3.1 COMPARISON WITH EXISTING METHODS

Baselines. We compare with SynCamMaster (Bai et al., 2024) (Wan2.1 version), a joint view–time diffusion model that generates all views directly from text. We also include ReCamMaster (Bai et al., 2025) and NeoVerse (Yang et al., 2026) as video re-rendering baselines. Both synthesize videos from new viewpoints given a source video. NeoVerse uses estimated geometry to guide re-rendering. We give both the same randomly chosen MULTILIVE view as source and evaluate only their re-rendered outputs, so that the two re-renderers share content. This is not a matched comparison to text-to-multi-view generation: MULTILIVE produces all views from text and cameras, with no source video.

Quantitative results. As shown in Table 1, MULTILIVE outperforms SynCamMaster on all seven per-view and synchronization metrics: it achieves better per-view quality and stronger view synchronization, with lower FID/FVD, higher CLIP-T/CLIP-F, more matched pixels, lower FVD-V, and higher CLIP-V. Against the re-rendering references, ReCamMaster remains better on FID and matched pixels, consistent with being conditioned on a source view. On camera metrics, VGGT errors remain close to the 50-step methods, with higher rotation error under four-step generation. As shown in Table 2, MULTILIVE improves over SynCamMaster in aesthetic quality, imaging quality, temporal flickering, and motion smoothness, with small decreases in subject and background consistency.

Qualitative results. Figure 4 shows that MULTILIVE maintains consistent subject appearance and coordinated motion across views, as illustrated by the running horse and the tennis player. ReCamMaster and NeoVerse preserve the overall source content but can alter appearance details during re-rendering, such as the player's clothing in the tennis example. MULTILIVE preserves these details across jointly generated views. Compared with SynCamMaster, our method generates more realistic videos and generalizes better across diverse scenes and visual styles.

Efficiency. Because SynCamMaster generates two views per pass, Table 1 reports NFE and latency for a two-view generation. MULTILIVE completes this pass in 38.12 s with four denoising steps, the lowest latency among the compared methods, for up to 26.2× speedup.

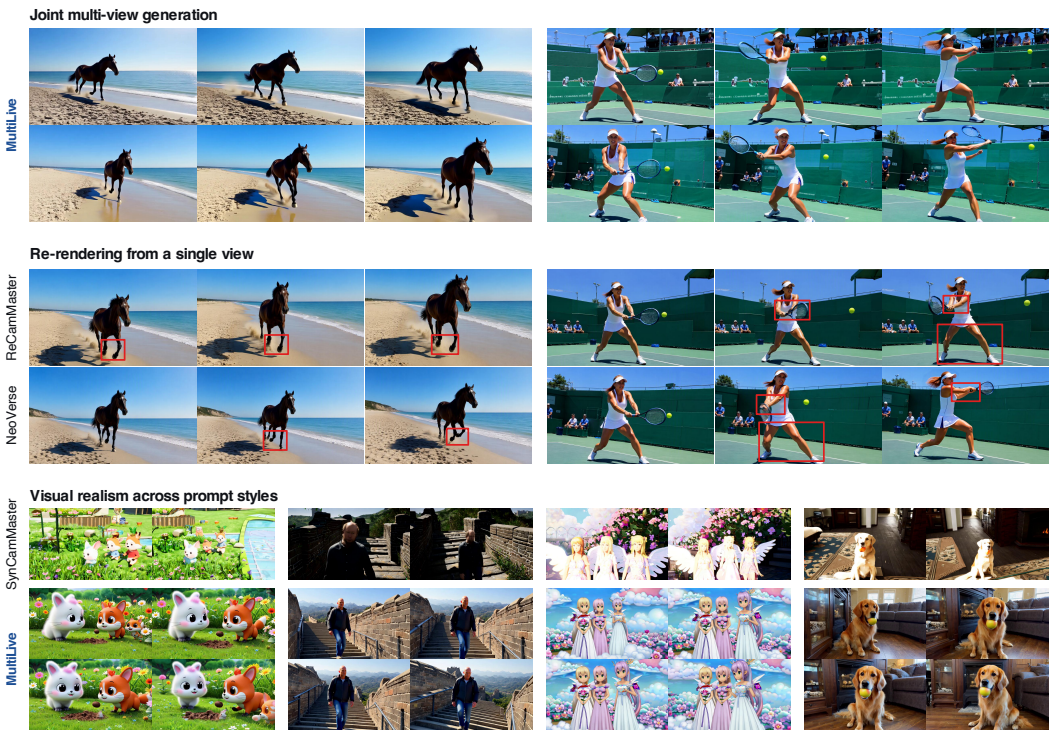


Figure 4: **Qualitative comparison of multi-view video generation.** The top panel shows joint generation by MULTILIVE, the middle panel compares video-to-video re-rendering methods, and the bottom panel illustrates generalization across diverse prompts.

Table 2: **Comparison on VBench metrics.**

Method	Subject Consistency ↑	Background Consistency ↑	Aesthetic Quality ↑	Imaging Quality ↑	Temporal Flickering ↑	Motion Smoothness ↑
ReCamMaster	96.62	95.44	61.93	68.37	96.48	98.56
NeoVerse	91.41	91.77	60.59	66.54	94.09	98.09
SynCamMaster	98.32	98.06	60.85	68.02	98.95	99.23
MULTILIVE	97.68	97.01	63.51	70.29	99.00	99.36

3.2 ABLATION STUDIES

Table 3 examines how the two priors are combined and the designs that support this combination: the multi-view teacher, generator architecture and initialization, and student-score estimation. Full MULTILIVE serves as the common reference.

Training arrangement. Mixed-data SFT on DL3DV images and Koala-36M (Wang et al., 2025c) videos, at comparable compute to Full, produces flickering and geometric distortions and trails Full on all per-view and synchronization metrics. Single-teacher variants share Full's architecture and initialization but retain only one distillation branch. Video-only supervision preserves per-view quality but yields weak synchronization, whereas MV-only supervision favors synchronization over video quality: its FVD-V is slightly better than Full's, but its FVD is substantially worse. Both sequential variants use current student samples and receive supervision from both teachers, but in separate phases where only one teacher's gradient updates the generator at each step. The resulting balance depends on the last teacher: Video → MV yields better synchronization, while MV → Video achieves lower FID and FVD. Full achieves lower FVD and FVD-V than both orders with fewer student-training iterations. These results suggest that applying both constraints to the same generated sample at each step is more effective than accumulating them across separate phases.

Table 3: **Ablation studies.** We evaluate training arrangement, multi-view teacher design and supervision, generator initialization, and student-score estimation relative to MULTILIVE. Mixed-data SFT uses 50 denoise steps; all other variants use 4-step distillation.

		Training	Per-View Quality				Camera Accuracy		View Synchronization		
Group	Variant	Train steps	FID ↓	FVD ↓	CLIP-T ↑	CLIP-F ↑	RotErr ↓	TransErr ↓	Mat. Pix. (K) ↑	FVD-V ↓	CLIP-V ↑
Training arrangement	Mixed-data SFT	20000	143.64	132.55	26.39	97.51	8.13	15.07	197.32	54.92	88.71
	Video teacher only	2000	113.63	114.37	28.12	98.74	24.10	25.53	157.76	69.66	85.88
	MV teacher only	2000	122.71	140.21	28.44	99.02	7.90	11.86	343.93	23.80	96.66
	Video → MV	4000	119.18	126.74	28.41	99.90	9.64	13.27	321.75	30.86	95.54
	MV → Video	4000	115.42	116.81	28.31	98.68	17.85	21.46	233.48	51.37	91.62
Multi-view teacher	Image-only, $s = 1$	2000	110.99	113.42	28.95	99.33	15.32	16.85	244.59	35.48	95.28
	Image-only, all T	2000	138.21	151.08	29.25	99.58	13.64	18.47	394.47	25.83	97.74
	Causal-latent MV teacher, $s = 1$	2000	116.05	115.77	29.13	98.98	16.27	18.85	204.25	35.69	96.23
Generator design and initialization	Same-time attention	2000	115.02	137.61	27.54	96.93	18.73	23.38	423.00	26.25	96.47
	Wan initialization	2000	121.98	127.83	29.85	98.17	19.46	22.15	221.03	37.72	94.52
	Wan initialization	5000	116.56	112.95	29.23	99.19	15.94	19.37	397.62	26.11	97.00
Student-score estimation	Single joint score forward	2000	160.91	185.29	24.93	97.96	18.63	20.82	249.41	36.70	95.71
Full MULTILIVE		2000	114.67	111.32	30.67	99.69	13.18	18.21	443.26	24.79	97.91

Multi-view teacher. We independently vary the teacher construction and supervision positions while retaining video-teacher supervision. Extending image-only supervision from the initial latent position to all 21 positions improves synchronization but degrades visual quality. With the causal-latent multi-view teacher, the same extension improves both quality and synchronization. Over all latent-time positions, the adapted teacher also outperforms the image-only teacher across all reported metrics. These results show that causal-latent adaptation enables broader multi-view supervision to improve synchronization while preserving video quality.

Generator design and initialization. Restricting the adapter's cross-view attention to the same latent time position yields pixel correspondence close to that of Full, but causes severe flickering between frames despite retaining the per-view temporal path. This highlights the importance of direct interaction across both time and views for temporal consistency. Changing only the student initialization from the multi-view teacher to Wan worsens all reported metrics under the same training budget. Longer training substantially narrows the gap, but Full still performs better across all reported metrics with fewer student-training iterations, demonstrating the value of multi-view initialization for efficient student optimization.

Branch-aware student scores. Estimating both objectives from a single joint score forward degrades all metrics, with a pronounced drop in video quality. Both this variant and Full use a shared fake-score model; Full estimates a separate score for each readout distribution. The results support matching each teacher score with the corresponding student score within a shared model.

3.3 APPLICATION

Action-conditioned multi-view world modeling. To extend the framework beyond text conditioning, we apply it to a pretrained TA2V backbone. We retain its action pathway, add the Geometry-Aware Adapter, and distill both teachers into TA2MV. A shared world-frame trajectory is converted into camera-local controls, enabling four synchronized views conditioned on one action trajectory. This extension transfers the same recipe to action-conditioned multi-view world modeling.

4 CONCLUSION

MULTILIVE unifies video and multi-view priors through on-policy dual-teacher matching, without synchronized multi-view video data or explicit 3D reconstruction. It concurrently enforces temporal and cross-view consistency, achieving strong quality and efficient four-view generation in four denoising steps. The same recipe further extends to action-conditioned multi-view world modeling. The few-step budget can still limit camera-control precision on demanding configurations, and some dependencies that couple time and viewpoint remain only indirectly constrained; we discuss these limitations in Appendix H.

AI USE STATEMENT

In this work, we did not use generative AI tools for any required-disclosure task. Specifically, we did not use them to generate synthetic datasets, develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in writing proofs, propose or refine hypotheses, design or provide feedback on methodology or experiments, implement methods, translate, clean or reformat datasets, support qualitative or thematic data analysis, or interpret results. Tasks outside the scope of this work were not applicable.

Additionally, we used generative AI tools for recommended-disclosure tasks, including code debugging, language editing and readability improvements of the manuscript, and scientific figure creation. In particular, GPT-image-2 assisted with the visual composition and rendering of experimental illustrations, including their layout and visual structure. This assistance was limited to visual composition and rendering; the authors determined the scientific content, and GPT-image-2 was not used to generate, modify, analyze, or interpret experimental data or results. We have reviewed all AI-assisted work. In particular, all AI-assisted code was inspected and tested for correctness by the authors, all manuscript edits were reviewed and verified by the authors, and all AI-assisted figures and image assets were checked for accuracy and consistency where applicable. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REFERENCES

Jianhong Bai, Menghan Xia, Xintao Wang, Ziyang Yuan, Xiao Fu, Zuozhu Liu, Haoji Hu, Pengfei Wan, and Di Zhang. Syncammaster: Synchronizing multi-camera video generation from diverse viewpoints. *arXiv preprint arXiv:2412.07760*, 2024.

Jianhong Bai, Menghan Xia, Xiao Fu, Xintao Wang, Lianrui Mu, Jinwen Cao, Zuozhu Liu, Haoji Hu, Xiang Bai, Pengfei Wan, and Di Zhang. Recammaster: Camera-controlled generative rendering from a single video. *arXiv preprint arXiv:2503.11647*, 2025.

Siyi Chen, Shaowei Liu, Yixuan Jia, Zian Wang, Huan Ling, Qing Qu, and Jun Gao. Data-forcing distillation: Restoring diversity and fidelity in few-step video generation. *arXiv preprint arXiv:2606.18478*, 2026a.

Tingxi Chen, Ke Hao, Yabo Chen, Zhengxue Cheng, Rong Xie, Li Song, Haibin Huang, Chi Zhang, and Xuelong Li. Full-4d: Generating full-scope 4d scenes from a single-view video. *arXiv preprint arXiv:2605.25500*, 2026b.

Gal Fiebelman, Hadar Averbuch-Elor, and Sagie Benaim. Mv-forcing: Long multi-view video generation via 4d-grounded spatio-temporal self-forcing. *arXiv preprint arXiv:2607.05376*, 2026.

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

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. Gans trained by a two time-scale update rule converge to a local nash equilibrium, 2018. URL https://arxiv.org/abs/1706.08500.

Zehuan Huang, Haoran Feng, Yang-Tian Sun, Yuan-Chen Guo, Yan-Pei Cao, and Lu Sheng. Animax: Animating the inanimate in 3d with joint video-pose diffusion models. In *Proceedings of the SIGGRAPH Asia 2025 Conference Papers*, pp. 1–13, 2025.

Ziqi Huang, Yinan He, Jiashuo Yu, Fan Zhang, Chenyang Si, Yuming Jiang, Yuanhan Zhang, Tianxing Wu, Qingyang Jin, Nattapol Chanpaisit, et al. Vbench: Comprehensive benchmark suite for video generative models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 21807–21818. IEEE, 2024.

Hyeonho Jeong, Suhyeon Lee, and Jong Chul Ye. Reangle-a-video: 4d video generation as video-to-video translation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 11164–11175. IEEE, 2025.

Zhengfei Kuang, Shengqu Cai, Hao He, Yinghao Xu, Hongsheng Li, Leonidas Guibas, and Gordon Wetzstein. Collaborative video diffusion: Consistent multi-video generation with camera control. In *NeurIPS*, 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 *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6775–6785. IEEE, 2024.

Bing Li, Cheng Zheng, Wenxuan Zhu, Jinjie Mai, Biao Zhang, Peter Wonka, and Bernard Ghanem. Vivid-zoo: Multi-view video generation with diffusion model, 2024.

Ruilong Li, Brent Yi, Junchen Liu, Hang Gao, Yi Ma, and Angjoo Kanazawa. Cameras as relative positional encoding. *Advances in Neural Information Processing Systems*, 2025.

Lu Ling, Yichen Sheng, Zhi Tu, Wentian Zhao, Cheng Xin, Kun Wan, Lantao Yu, Qianyu Guo, Zixun Yu, Yawen Lu, et al. Dl3dv-10k: A large-scale scene dataset for deep learning-based 3d vision. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22160–22169. IEEE, 2024.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matt Le. Flow matching for generative modeling, 2023. URL https://arxiv.org/abs/2210.02747.

Yanzuo Lu, Yuxi Ren, Xin Xia, Shanchuan Lin, Xing Wang, Xuefeng Xiao, Andy J Ma, Xiaohua Xie, and Jian-Huang Lai. Adversarial distribution matching for diffusion distillation towards efficient image and video synthesis. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 16818–16829. IEEE, 2025.

Weili Nie, Julius Berner, Nanye Ma, Chao Liu, Saining Xie, and Arash Vahdat. Transition matching distillation for fast video generation. *arXiv preprint arXiv:2601.09881*, 2026.

Jangho Park, Taesung Kwon, and Jong Chul Ye. Zero4d: Training-free 4d video generation from single video using off-the-shelf video diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4045–4054, 2026.

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

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021. URL https://arxiv.org/abs/2103.00020.

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 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Georgy Savva, Oscar Michel, Daohan Lu, Suppakit Waiwitlikhit, Timothy Meehan, Dhairya Mishra, Srivats Poddar, Jack Lu, and Saining Xie. Solaris: Building a multiplayer video world model in minecraft. *arXiv preprint arXiv:2602.22208*, 2026.

Xuelun Shen, Wei Yin, Matthias Müller, Zijun Li, Kaixuan Wang, Xiaozhi Chen, Cheng Wang, et al. Gim: Learning generalizable image matcher from internet videos. In *International Conference on Learning Representations*, volume 2024, pp. 13182–13197, 2024.

Huiqiang Sun, Zhan Peng, Size Wu, Kun Wang, Kang Liao, Dianyi Wang, Xingyu Zeng, Sheng Jin, Yangguang Li, Zhiguo Cao, Ziwei Liu, and Wei Li. Prisma-world: Camera-controllable multi-agent video world model. *arXiv preprint arXiv:2606.09507*, 2026a.

Yang-Tian Sun, Yi-Hua Huang, Lin Ma, Xiaoyang Lyu, Yan-Pei Cao, and Xiaojuan Qi. Splatter a video: Video gaussian representation for versatile processing. *arXiv preprint arXiv:2406.13870*, 2024.

Yang-Tian Sun, Zehuan Huang, Yifan Niu, Lin Ma, Yan-Pei Cao, Yuewen Ma, and Xiaojuan Qi. Stereo world model: Camera-guided stereo video generation. *arXiv preprint arXiv:2603.17375*, 2026b.

Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphael Marinier, Marcin Michalski, and Sylvain Gelly. Towards accurate generative models of video: A new metric & challenges, 2019. URL https://arxiv.org/abs/1812.01717.

Team Wan, Ang Wang, Baole Ai, Bin Wen, Chaojie Mao, Chen-Wei Xie, Di Chen, Feiwu Yu, Haiming Zhao, Jianxiao Yang, Jianyuan Zeng, Jiayu Wang, Jingfeng Zhang, Jingren Zhou, Jinkai Wang, Jixuan Chen, Kai Zhu, Kang Zhao, Keyu Yan, Lianghua Huang, Mengyang Feng, Ningyi Zhang, Pandeng Li, Pingyu Wu, Ruihang Chu, Ruili Feng, Shiwei Zhang, Siyang Sun, Tao Fang, Tianxing Wang, Tianyi Gui, Tingyu Weng, Tong Shen, Wei Lin, Wei Wang, Wei Wang, Wenmeng Zhou, Wente Wang, Wenting Shen, Wenyuan Yu, Xianzhong Shi, Xiaoming Huang, Xin Xu, Yan Kou, Yangyu Lv, Yifei Li, Yijing Liu, Yiming Wang, Yingya Zhang, Yitong Huang, Yong Li, You Wu, Yu Liu, Yulin Pan, Yun Zheng, Yuntao Hong, Yupeng Shi, Yutong Feng, Zeyinzi Jiang, Zhen Han, Zhi-Fan Wu, and Ziyu Liu. Wan: Open and advanced large-scale video generative models. *arXiv preprint arXiv:2503.20314*, 2025.

Chaoyang Wang, Peiye Zhuang, Tuan Duc Ngo, Willi Menapace, Aliaksandr Siarohin, Michael Vasilkovsky, Ivan Skorokhodov, Sergey Tulyakov, Peter Wonka, and Hsin-Ying Lee. 4real-video: Learning generalizable photo-realistic 4d video diffusion. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 17723–17732. IEEE, 2025a.

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

Qiuheng Wang, Yukai Shi, Jiarong Ou, Rui Chen, Ke Lin, Jiahao Wang, Boyuan Jiang, Haotian Yang, Mingwu Zheng, Xin Tao, et al. Koala-36m: A large-scale video dataset improving consistency between fine-grained conditions and video content. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8428–8437. IEEE, 2025c.

Zhouxia Wang, Ziyang Yuan, Xintao Wang, Yaowei Li, Tianshui Chen, Menghan Xia, Ping Luo, and Ying Shan. Motionctrl: A unified and flexible motion controller for video generation. In *ACM SIGGRAPH 2024 Conference Papers*, pp. 1–11, 2024.

Haoyu Wu, Jiwen Yu, Yingtian Zou, and Xihui Liu. Multiworld: Scalable multi-agent multi-view video world models. *arXiv preprint arXiv:2604.18564*, 2025a.

Qi Wu, Khiem Vuong, Minsik Jeon, Srinivasa Narasimhan, and Deva Ramanan. Novel view synthesis as video completion. *arXiv preprint arXiv:2604.08500*, 2026.

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, 2025b.

Yiming Xie, Chun-Han Yao, Vikram Voleti, Huaizu Jiang, and Varun Jampani. SV4D: Dynamic 3d content generation with multi-frame and multi-view consistency. *arXiv preprint arXiv:2407.17470*, 2024.

Dejia Xu, Yifan Jiang, Chen Huang, Liangchen Song, Thorsten Gernoth, Liangliang Cao, Zhangyang Wang, and Hao Tang. Cavia: Camera-controllable multi-view video diffusion with view-integrated attention. *arXiv preprint arXiv:2410.10774*, 2024a.

Dejia Xu, Weili Nie, Chao Liu, Sifei Liu, Jan Kautz, Zhangyang Wang, and Arash Vahdat. Camco: Camera-controllable 3d-consistent image-to-video generation. *arXiv preprint arXiv:2406.02509*, 2024b.

Yuxue Yang, Lue Fan, Ziqi Shi, Junran Peng, Feng Wang, and Zhaoxiang Zhang. Neoverse: Enhancing 4d world model with in-the-wild monocular videos. *arXiv preprint arXiv:2601.00393*, 2026.

Zeyu Yang, Zijie Pan, Chun Gu, and Li Zhang. Diffusion^2: Dynamic 3d content generation via score composition of video and multi-view diffusion models. In *International Conference on Learning Representations*, volume 2025, pp. 76979–77000, 2025.

Chun-Han Yao, Yiming Xie, Vikram Voleti, Huaizu Jiang, and Varun Jampani. SV4D2.0: Enhancing spatio-temporal consistency in multi-view video diffusion for high-quality 4d generation. *arXiv preprint arXiv:2503.16396*, 2025.

Minghao Yin, Yukang Cao, Songyou Peng, and Kai Han. Splat4d: Diffusion-enhanced 4d gaussian splatting for temporally and spatially consistent content creation. In *Proceedings of the Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers*, pp. 1–10, 2025.

Tianwei Yin, Michaël Gharbi, Taesung Park, Richard Zhang, Eli Shechtman, Fredo Durand, and William T Freeman. Improved distribution matching distillation for fast image synthesis. In *NeurIPS*, 2024a.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Frédo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *CVPR*, 2024b.

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.

Haiyu Zhang, Xinyuan Chen, Yaohui Wang, Xihui Liu, Yunhong Wang, and Yu Qiao. 4diffusion: Multi-view video diffusion model for 4d generation. *Advances in Neural Information Processing Systems*, 37:15272–15295, 2024.

Wenhu Zhang, Kun Cheng, Changyuan Wang, Shiyao Li, Yuechen Zhang, Wenbo Li, Jiajun Zha, Jingyi Zhang, Kang Zhao, and Jiaya Jia. Codmd: Copula-aware distribution matching distillation for fast video generation. *arXiv preprint arXiv:2606.21982*, 2026.

Guangcong Zheng, Teng Li, Rui Jiang, Yehao Lu, Tao Wu, and Xi Li. Cami2v: Camera-controlled image-to-video diffusion model. *arXiv preprint arXiv:2410.15957*, 2024.

Haiyang Zhou, Wangbo Yu, Chaoran Feng, Xunyu Zhou, Yonghong Tian, and Li Yuan. Uniworld-view: Large-baseline view synthesis via video diffusion models. *arXiv preprint arXiv:2608.04701*, 2026.

Yuanzhi Zhu, Hanshu Yan, Huan Yang, Kai Zhang, and Junnan Li. Accelerating video diffusion models via distribution matching. *arXiv preprint arXiv:2412.05899*, 2024.

A Related Work

A.1 Multi-View Video Generation

Recent video diffusion models have extended text-to-video generation from a single camera stream to synchronized camera rigs. One line of work directly models a joint view–time grid, using cross-view attention, synchronized generation windows, or camera-conditioned modules to couple dynamic content across viewpoints (Kuang et al., 2024; Bai et al., 2024; Xu et al., 2024a; Wang et al., 2025a; Li et al., 2024; Wu et al., 2025b; Xie et al., 2024; Yao et al., 2025; Sun et al., 2026b; Zhang et al., 2024; Huang et al., 2025). These methods make cross-view correspondence an explicit part of generation, but they usually depend on synchronized multi-view video, synthetic 4D training data, or mixed training sources that only partially cover the desired joint distribution. A related set of camera-controlled video methods learns to control viewpoint motion or camera trajectories for a single generated video (He et al., 2024; Wang et al., 2024; Xu et al., 2024b; Zheng et al., 2024). These methods provide useful camera-control mechanisms, but they do not directly generate a synchronized set of calibrated videos. MultiLive instead generates all views jointly while avoiding synchronized multi-view video training data: the same on-policy student sample is supervised as per-view videos by a video teacher and as synchronized view sets by a multi-view teacher.

A.2 Video Re-Rendering

Video re-rendering and novel-view video synthesis methods take an observed or generated source video and synthesize target camera views. Some approaches combine off-the-shelf video diffusion models with estimated geometry or 3D proxies to move a video through a target camera path (Park et al., 2026; Jeong et al., 2025; Ren et al., 2025; Bai et al., 2025; Yang et al., 2026; Fiebelman et al., 2026). Recent novel-view synthesis systems further propose geometry priors or video generation priors for sparse-view or large-baseline camera transitions (Kwak et al., 2024; Yu et al., 2024; Sun et al., 2024; Zhou et al., 2026; Wu et al., 2026). These methods are effective when a source video is available, and they can preserve source content while synthesizing new viewpoints. However, their outputs inherit the source-view choice and can be affected by depth, geometry, or occlusion errors; independently re-rendered targets may also lack global coordination across the entire camera rig. MultiLive addresses a different setting: it generates the complete calibrated multi-view video from text and camera poses without privileging any source view.

A.3 Diffusion Distillation and Teacher Composition

Diffusion distillation reduces sampling cost by training a student generator to match a teacher distribution in fewer denoising steps. Distribution matching distillation (DMD) optimizes a distribution-level objective rather than imitating a full teacher trajectory (Yin et al., 2024b;a), and recent work adapts distribution matching, consistency learning, and transition matching to efficient video generation (Zhu et al., 2024; Nie et al., 2026; Lu et al., 2025; Chen et al., 2026a; Zhang et al., 2026). A separate direction combines complementary generative priors, for example by composing video and multi-view diffusion scores for dynamic 3D content generation (Yang et al., 2025). MultiLive differs from test-time score composition and single-teacher video distillation. It trains one camera-aware few-step generator with two branch-matched distribution-matching objectives: a video branch transfers the temporal prior of a pretrained video model (Wan et al., 2025), while a multi-view branch transfers synchronized-view geometry from a camera-conditioned multi-view teacher (Gao* et al., 2024). The shared fake-score model estimates the current student distribution in both readout spaces, allowing the two teachers to supervise different branch-level readouts of the same generated multi-view video.

B Algorithmic Details

Let $\hat{X}_0 \in \mathbb{R}^{B\times C\times(TV)\times H\times W}$ use time-major order (t, v). The branch-level readouts satisfy $\mathcal{R}_{\mathrm{vid}}(X) \in \mathbb{R}^{(BV)\times C\times T\times H\times W}$ and $\mathcal{R}_{\mathrm{mv}}(X) \in \mathbb{R}^{(BT)\times C\times V\times H\times W}$. These operators realize the main-text readouts: $\mathcal{R}_{\mathrm{vid}}(\hat{X}_0) \equiv \hat{\mathbf{X}}_0^{\mathrm{vid}}$ and $\mathcal{R}_{\mathrm{mv}}(\hat{X}_0) \equiv \hat{\mathbf{X}}_0^{\mathrm{mv}}$. The former groups each view into a temporal trajectory, while the latter concatenates the synchronized view sets at all T latent-time positions along the batch axis. In the paper, latent-time indices are one-based; the implementation

uses the equivalent zero-based indices. The teacher and fake-score model process this concatenated batch with shared parameters. For implementation, the model outputs are converted to x_0 space for the DMD and fake-score losses; let d_b and y_b denote the fake-score and teacher outputs in this space. The symbol sg denotes stop-gradient and $\varepsilon = 10^{-6}$ is a numerical stabilizer. During training, the rollout exits at a uniformly sampled step of the four-step SDE schedule. Its prefix is detached, and only the student forward at the sampled exit step carries the generator gradient. Specifically, D_ϕ^{vid} and D_ϕ^{mv} denote branch-matched forwards of the same network: the former uses the video-native per-view temporal layout, while the latter uses the branch-level synchronized-view layout and camera metadata. They share the single parameter set ϕ and produce separate branch predictions.

For $b \in \{\mathrm{vid}, \mathrm{mv}\}$, let $\Delta_b = d_b - y_b$ be the x_0-prediction residual of branch b, and let $\rho_b = (\mathrm{mean}\,|x_0^b - \mathrm{sg}[y_b]| + \varepsilon)^{-1}$. This residual is proportional to the main-text score residual $\mathbf{r}_b$; the factor is absorbed into w_{vid} and w_{mv}. The DMD update uses $g_b = \rho_b\,\mathrm{sg}[\Delta_b]$ and

$$\mathcal{L}_b^{\mathrm{DMD}} = \frac{1}{2}\|x_0^b - \mathrm{sg}[x_0^b - g_b]\|_2^2. \tag{14}$$

The fake-score update uses the resulting x_0 prediction and the loss $\mathcal{L}_b^{x_0} = \frac{1}{2}\|d_b - \mathrm{sg}[x_0^b]\|_2^2$. The implementation averages squared errors over elements within each branch.

Algorithm 1: Dual-teacher multi-view DMD (Yin et al., 2024b;a).

Input: frozen $\mathsf{T}_{\mathrm{vid}}, \mathsf{T}_{\mathrm{mv}}$; trainable generator G_θ and one shared fake-score model D_ϕ; noise kernel Ψ; weights $(\lambda_{\mathrm{vid}}, \lambda_{\mathrm{mv}})$; period K.
Output: G_θ.

$G_\theta, D_\phi \leftarrow \mathrm{copy}(\mathsf{T}_{\mathrm{mv}})$; keep G_θ, D_ϕ trainable; freeze $\mathsf{T}_{\mathrm{vid}}, \mathsf{T}_{\mathrm{mv}}$.
$D_\phi^{\mathrm{vid}}(\cdot) \leftarrow D_\phi(\cdot; \mathrm{video})$ and $D_\phi^{\mathrm{mv}}(\cdot) \leftarrow D_\phi(\cdot; \mathrm{multi\text{-}view})$; both forwards share ϕ.
for $n = 1, 2, \ldots$ **do**
  Sample $(c, \Pi) \sim p_{\mathrm{data}}$, $z, \epsilon \sim \mathcal{N}(0, I)$, $\tau \sim p_\tau$, and $k \sim \mathrm{Unif}\{1, \ldots, 4\}$.
  $\hat{X}_0 \leftarrow \mathrm{Rollout}_k(G_\theta, z, c, \Pi)$; detach the rollout prefix and retain generator gradients only at exit step k.
  $\hat{X}_\tau \leftarrow \Psi_\tau(\hat{X}_0, \epsilon)$; this noised sample is shared by both updates.
  $x_0^{\mathrm{vid}} \leftarrow \mathcal{R}_{\mathrm{vid}}(\hat{X}_0)$; $x_\tau^{\mathrm{vid}} \leftarrow \mathcal{R}_{\mathrm{vid}}(\hat{X}_\tau)$.
  $x_0^{\mathrm{mv}} \leftarrow \mathcal{R}_{\mathrm{mv}}(\hat{X}_0)$; $x_\tau^{\mathrm{mv}} \leftarrow \mathcal{R}_{\mathrm{mv}}(\hat{X}_\tau)$; the readout concatenates all T latent-time positions.
  if $n \bmod K = 0$ **then** *// student update*
    $d_{\mathrm{vid}} \leftarrow \mathrm{sg}[D_\phi^{\mathrm{vid}}(x_\tau^{\mathrm{vid}}, \tau, c)]$; $y_{\mathrm{vid}} \leftarrow \mathrm{sg}[\mathsf{T}_{\mathrm{vid}}(x_\tau^{\mathrm{vid}}, \tau, c)]$.
    $d_{\mathrm{mv}} \leftarrow \mathrm{sg}[D_\phi^{\mathrm{mv}}(x_\tau^{\mathrm{mv}}, \tau, c, \Pi)]$; $y_{\mathrm{mv}} \leftarrow \mathrm{sg}[\mathsf{T}_{\mathrm{mv}}(x_\tau^{\mathrm{mv}}, \tau, c, \Pi)]$.
    $\Delta_{\mathrm{vid}} \leftarrow d_{\mathrm{vid}} - y_{\mathrm{vid}}$; $\Delta_{\mathrm{mv}} \leftarrow d_{\mathrm{mv}} - y_{\mathrm{mv}}$; set $g_b \leftarrow \rho_b\,\mathrm{sg}[\Delta_b]$.
    $\mathcal{L}_\theta \leftarrow \lambda_{\mathrm{vid}}\mathcal{L}_{\mathrm{vid}}^{\mathrm{DMD}} + \lambda_{\mathrm{mv}}\mathcal{L}_{\mathrm{mv}}^{\mathrm{DMD}}$; $\theta \leftarrow \mathrm{AdamStep}(\theta, \nabla_\theta\mathcal{L}_\theta)$.
  else *// fake-score update*
    $\bar{x}_0^b \leftarrow \mathrm{sg}[x_0^b]$ and $\bar{x}_\tau^b \leftarrow \mathrm{sg}[x_\tau^b]$ for $b \in \{\mathrm{vid}, \mathrm{mv}\}$; freeze θ.
    $\mathcal{L}_\phi^{\mathrm{vid}} \leftarrow \frac{1}{2}\|D_\phi^{\mathrm{vid}}(\bar{x}_\tau^{\mathrm{vid}}, \tau, c) - \bar{x}_0^{\mathrm{vid}}\|_2^2$;
    $\mathcal{L}_\phi^{\mathrm{mv}} \leftarrow \frac{1}{2}\|D_\phi^{\mathrm{mv}}(\bar{x}_\tau^{\mathrm{mv}}, \tau, c, \Pi) - \bar{x}_0^{\mathrm{mv}}\|_2^2$; $\phi \leftarrow \mathrm{AdamStep}(\phi, \nabla_\phi(\mathcal{L}_\phi^{\mathrm{vid}} + \mathcal{L}_\phi^{\mathrm{mv}}))$.
  end if
end for

We use $K = 5$, $(\lambda_{\mathrm{vid}}, \lambda_{\mathrm{mv}}) = (1, 1)$, and four SDE steps.

C Adapter and Teacher Construction Details

C.1 Geometry-Aware Adapter Implementation

At block ℓ, features use time-major order (t, v, h, w). The Wan path reorders them into one sequence per camera, applies the original self-attention with its $T/H/W$ rotary encoding, and restores the joint order. The adapter uses dedicated query, key, and value projections over the flattened $T \times V \times H \times W$ grid; its output projection maps the residual back into the backbone feature space.

Let $\mathbf{\Pi} = \{(\mathbf{K}_v, \mathbf{E}_v)\}_{v=1}^V$, where $\mathbf{K}_v \in \mathbb{R}^{3\times3}$ and $\mathbf{E}_v \in \mathbb{R}^{4\times4}$ are intrinsics and homogeneous world-to-camera extrinsics. Following Wan's $T/H/W$ channel partition, PRoPE (Li et al., 2025) replaces the adapter's T-axis rotary block with a camera-transformed block and retains the H/W rotary blocks. For $\mathbf{P}_v = \mathrm{diag}(\mathbf{K}_v, 1)\mathbf{E}_v$, we apply

$$\mathbf{q}_{v,\mathrm{cam}} \leftarrow \mathbf{P}_v^\top \mathbf{q}_{v,\mathrm{cam}}, \qquad \mathbf{k}_{v,\mathrm{cam}} \leftarrow \mathbf{P}_v^{-1}\mathbf{k}_{v,\mathrm{cam}}. \tag{15}$$

Values receive no camera-dependent transformation after projection. The implementation uses the normalized extrinsics described in Appendix D in place of $\mathbf{E}_v$.

At insertion, adapter query, key, and value projections and query/key normalization parameters are copied from the base attention. The residual output projection and its bias are initialized to zero, preserving the backbone function at insertion. Full subsequently loads compatible pretrained adapter parameters through this interface.

C.2 Causal-Latent Teacher Training

Following the camera-conditioned multi-view setup of CAT3D (Gao* et al., 2024), each sample contains synchronized images, a text condition, and calibrated cameras. Each image is repeated along time and encoded by the native causal Wan VAE. At each training step, we sample $q \sim \mathrm{Unif}\{1, \ldots, T\}$ and collect position q from all views to form $\mathbf{Z}_q$; the same position is used across views. We perturb this set as $\mathbf{Z}_\tau^{(q)} = \Psi_\tau(\mathbf{Z}_q, \boldsymbol{\epsilon})$ and train $F_{\bar{\omega},\psi}$ to predict $\boldsymbol{\epsilon} - \mathbf{Z}_q$ using the main-text flow-matching objective. Only the Geometry-Aware Adapter is updated; the Wan backbone, VAE, and text encoder remain frozen. The trained model is then frozen for distillation, where it supervises synchronized student slices at all latent positions. Data sampling and training settings are specified in Appendix D.

D Training and Implementation Details

We build MultiLive on Wan2.1-T2V-1.3B with four camera views. The student packs its latent video in time-major order as $\hat{X}_0 \in \mathbb{R}^{B\times 16\times 84\times 60\times 104}$, corresponding to $T = 21$ latent-time positions and $V = 4$ views at 480×832 pixels. The causal video VAE therefore decodes 21 latent positions into 81 RGB frames. The multi-view teacher is trained on calibrated multi-view images randomly sampled from DL3DV with a camera clip interval of 24: for each sample, the loader randomly chooses a scene, start frame, temporal direction, and four-view camera clip. The four views share the caption associated with the first selected frame. The teacher uses the position-specific latent representation described in the main paper. At each teacher-training iteration, the latent-time position q is sampled uniformly from $\{1, \ldots, 21\}$. Position $q = 1$ is the first-frame latent, while $q > 1$ are later-frame latents; sampling all positions trains the teacher to supervise the complete student sequence. Images are bicubically resized to the target resolution without cropping and their RGB values are mapped to $[-1, 1]$. We use world-to-camera extrinsics and normalize focal lengths and principal points by the corresponding image width and height. Writing each extrinsic as $[R_v \mid t_v]$, we compute $\mathbf{o}_v = -R_v^\top t_v$, $\bar{\mathbf{o}} = V^{-1}\sum_v \mathbf{o}_v$, $\mathbf{o}'_v = (\mathbf{o}_v - \bar{\mathbf{o}})/\max_u \|\mathbf{o}_u - \bar{\mathbf{o}}\|_2$, and $t'_v = -R_v\mathbf{o}'_v$; rotations and intrinsics are unchanged. We use $\mathbf{K}_v \in \mathbb{R}^{3\times 3}$, and the PRoPE camera subspaces $\mathbf{q}_{v,\mathrm{cam}}, \mathbf{k}_{v,\mathrm{cam}} \in \mathbb{R}^4$ are column vectors. The normalized homogeneous matrix used by PRoPE is $E'_v = \begin{bmatrix} R_v & t'_v \\ 0 & 1 \end{bmatrix}$, obtained from the main-text calibrated extrinsics $\mathbf{E}_v$. During distillation, the multi-view branch readout concatenates the synchronized view sets from all 21 positions of the student's causal-latent grid.

The training procedure has two stages. We first train the Geometry-Aware Adapter for 10,000 iterations while keeping the Wan backbone, VAE, and text encoder fixed; the resulting multi-view teacher is then frozen. The video teacher is the pretrained Wan video model. We initialize both the student generator G_θ and the shared fake-score model D_ϕ from the multi-view teacher and distill them for 2,000 iterations. Both stages use eight A800 GPUs, with per-GPU batch sizes of four for teacher training and one for distillation. The shared D_ϕ uses branch-matched forwards for the video and multi-view readouts, while the two teachers remain frozen throughout distillation.

During distillation, one generator update is performed every $K = 5$ iterations and the other four iterations update D_ϕ. We use equal branch weights $(\lambda_{\mathrm{vid}}, \lambda_{\mathrm{mv}}) = (1, 1)$. The generator uses a four-step SDE rollout from pure noise. For the rectified-flow noise kernel, $\Psi_\tau(X, \epsilon) = (1-\tau)X + \tau\epsilon$. Its configured time list is $(1000, 750, 500, 250, 0)$, followed by the shifted-time transform with shift 5 and clipping to $[0.001, 0.999]$. The same renoised sample is shared by both branches. Classifier-free guidance with scale 5 is applied to the two teacher predictions; student and fake-score forwards use conditional predictions. The diffusion time for each DMD/x_0 update is sampled at each training iteration from the shifted-uniform distribution on $[0.001, 0.999]$ with shift 5, and this same τ is used

by both branches; it is distinct from the four-step rollout grid. We use AdamW with learning rate 10^{-5}, betas $(0, 0.999)$, weight decay 0.01, bfloat16 arithmetic, and random seed 42. At inference, the model jointly generates four 81-frame views at 480P and 16 FPS without classifier-free guidance in the student rollout.

E Evaluation Protocol and Baseline Implementation

Evaluation setup. We evaluate all methods on the same 100 prompts sampled from MovieGen-Bench. MultiLive, ReCamMaster, and NeoVerse use one four-view camera rig per prompt, sampled from the SynCamVideo validation set, and are evaluated on four 81-frame views at 480×832 and 16 FPS. SynCamMaster is a native two-view model. For each prompt we independently sample an official supported camera pair from the SynCamVideo validation set, and we report quality and efficiency on that two-view output only. We do not stitch multiple passes into four views: unpaired views would not have been denoised together, which would understate its cross-view consistency. The re-rendering baselines use a source view generated by MultiLive under the same prompt and the same four-view target cameras. Camera poses are stored as world-to-camera matrices, with intrinsics normalized by image width and height, and are then converted to each method's official input interface. The centered-radius translation used by PRoPE (Section D) applies only to MultiLive; it is not applied to the baseline inputs. For ReCamMaster, target poses are expressed relative to the source view (source = identity).

Aggregation across views. Per-view metrics (FID, FVD, CLIP-T, CLIP-F) are computed on each view independently and averaged over the views of that method: four views for MultiLive, ReCamMaster, and NeoVerse, and two views for SynCamMaster. Cross-view metrics (Mat. Pix., FVD-V, CLIP-V) are computed on every unordered pair of views and averaged: $\binom{4}{2} = 6$ pairs for the four-view methods and the single pair for SynCamMaster. NFE counts the denoising steps of one native pass and does not double for classifier-free guidance. Latency is the wall-clock time of a two-view generation, including text encoding, denoising, and VAE decoding, and excluding model loading. For MultiLive this is a two-view forward of the joint generator; for SynCamMaster it is the native two-view pass; for ReCamMaster and NeoVerse it is the time to re-render two views independently, while NFE remains the step count of a single pass.

Camera accuracy. We obtain camera pose estimates from VGGT (Wang et al., 2025b) and evaluate RotErr and TransErr by comparing the estimated and target relative poses.

SynCamMaster. We use the official code release built on Wan2.1-T2V-1.3B. It generates two views jointly in a fixed 81-frame window with the default 50-step denoising schedule and classifier-free guidance scale 5.0.

ReCamMaster. We use the official code and checkpoint. The source video is a randomly selected view from MultiLive's output for the same prompt. Using the same source content for both re-rendering baselines ensures that output differences reflect re-rendering capability rather than variation in source quality. Following MV-Forcing (Fiebelman et al., 2026), we use an adapted checkpoint for the non-overlapping first-frame setting. Target camera poses are expressed relative to the source view (source = identity). The method re-renders each of the four target views independently.

NeoVerse. We use the official 14B implementation with the distilled few-step LoRA. Inference runs four denoising steps; classifier-free guidance is distilled into the model and not applied at inference. All other settings follow the official configuration. The method takes the same source video as ReCamMaster and the four target camera trajectories as input, and re-renders all four target views independently.

F Ablation Study Details

Unless noted, every variant uses the same frozen video teacher, the same frozen multi-view teacher (Geometry-Aware Adapter trained for 10,000 iterations on DL3DV), and the same fake-score topology, noise schedule, and optimizer as MultiLive. We change one factor at a time. Distillation variants use four SDE steps.

Training arrangement. Mixed-data SFT alternates DL3DV four-view images (clip interval 24) and Koala-36M 81-frame videos with equal probability. It trains for 20,000 steps so that wall-clock time on the same eight A800 GPUs matches MULTILIVE distillation, and is evaluated at 50 inference steps. The learning rate is 10^{-5} with linear decay and 500 warmup steps. This isolates how the two priors are combined—mixed-data supervision versus on-policy dual-teacher matching—rather than the optimizer-step count.

Single-teacher variants keep MULTILIVE's architecture and initialization but set the unused branch weight to zero ($\lambda_{\mathrm{mv}} = 0$ or $\lambda_{\mathrm{vid}} = 0$). Sequential variants run one teacher-only phase for 2,000 steps, then the other for 2,000 steps. The shared fake-score model is not reset; only the branch weights change. Even with twice as many student updates, applying the two constraints in separate phases remains worse than matching both teachers on the same sample at each step.

Multi-view teacher. We cross teacher construction with supervision support, keeping video-teacher distillation. The image-only teacher is trained for 10,000 steps on synchronized DL3DV images with a separate image-native encoder, without the causal-latent construction of Section 2.3. At distillation time, student latents are passed in without conversion, so later-frame video latents are scored by an image encoder. We evaluate it at $s = 1$ and at all $T = 21$ positions. A causal-latent teacher restricted to $s = 1$ uses MULTILIVE's multi-view teacher but reads out only the first latent position. Causal-latent supervision of all T positions is MULTILIVE.

Generator design and initialization. Same-time attention replaces the adapter's joint 4D attention over all $TVHW$ tokens with independent 3D attention over VHW tokens at each latent time. The multi-view teacher, student, and fake-score model all use this layout, so the comparison isolates the attention pattern.

Wan initialization changes only the student and fake-score start point: both are initialized from the Wan backbone with function-preserving adapter insertion. The frozen multi-view teacher remains the 10k Geometry-Aware Adapter. We evaluate this at 2,000 and 5,000 steps with the same learning-rate schedule as MULTILIVE.

Student-score estimation. The single-forward variant keeps the shared parameters ϕ but drops the branch-matched readout split. One joint forward over the packed grid produces a single score field, which is then sliced into video and multi-view components. Each branch's DMD loss pairs its teacher with the corresponding slice. The two readout distributions differ, so this shared field is not a score of p_θ^{vid} or p_θ^{mv}.

G APPLICATION DETAILS

For the world-model application, we initialize the multi-view teacher from Wan2.2-TI2V-5B. The teacher is trained on randomly sampled four-view DL3DV image groups with scene-level captions. We freeze the Wan backbone and train the multi-view adapter for 20,000 steps using AdamW with learning rate 10^{-4}, betas $(0.9, 0.95)$, weight decay 10^{-6}, 1,500 warm-up steps, gradient clipping at 1.0, and bfloat16 training on eight A800 GPUs. Unlike Section 2.3, the teacher uses the static-image VAE path, independent noise for the four view slots, and uniform sampling on the native Wan flow-matching timestep grid. Each sample randomly selects a scene and a circular four-frame clip with interval 24; images are resized to 480×832 and paired with a scene-level static caption. Camera centers follow the teacher's centered-radius normalization while rotations and intrinsics are preserved.

We then apply the dual-teacher distillation procedure described in the main paper to obtain TA2MV. The model jointly generates four views with 21 latent-time positions, corresponding to 81 RGB frames at 480×832 and 24 FPS. Latents are packed in time-major order, with joint input shape $[B, 48, 84, 30, 52]$. The action sequence contains one text control per latent-time position. The frozen text encoder produces at most 16 action tokens per position; explicit time IDs select the tokens, which are injected by frame-aligned action cross-attention in transformer blocks 0–14. For a shared world-frame trajectory, we construct one camera-local control sequence per view and route it with the corresponding view tokens. The training loader randomly samples the camera rig together with each clip. In the world-aligned setting, a shared camera trajectory is converted to view-local controls using the sampled rig before quantization, with the resulting controls aligned to the causal $1 + 4$ latent aggregation used by TA2V. No image condition is used.

We use four SDE sampling steps with shifted time list $(1000, 750, 500, 250, 0)$ and shift 5. Teacher predictions use classifier-free guidance with scale 5, while student and fake-score forwards are conditional. We use equal branch weights $(\lambda_{\text{vid}}, \lambda_{\text{mv}}) = (1, 1)$; one student update is followed by four fake-score updates. The student and fake-score models use AdamW with learning rate 10^{-5}, betas $(0, 0.999)$, weight decay 0.01, bfloat16 arithmetic, batch size one per GPU, eight A800 GPUs, and seed 42. Qualitative results are shown in the supplementary video.

H Limitations and Future Work

MultiLive is designed to transfer complementary video and multi-view priors into one efficient generator. This design allows all views to be produced jointly in a few steps, while the limited denoising budget can make camera-control precision depend on the difficulty of the camera configuration. Since the transferred priors are learned from 2D images and videos, the resulting scene representation is shaped by their visual regularities rather than by an explicit 3D/4D reconstruction objective. The two teachers provide complementary supervision for temporal evolution and cross-view agreement. Together, they offer a practical approximation to the broader space–time–view structure, while some dependencies that involve both dimensions remain only indirectly constrained. Because both objectives are applied to the same jointly generated sample, 4D attention can exchange information across time and views, and on-policy distillation repeatedly reinforces their consistency. Future work can allocate the few-step budget adaptively for demanding camera configurations, and can further couple the two complementary priors on the same generated sample as stronger video and multi-view teachers become available, to improve camera control and long-range consistency while retaining the efficiency of joint generation.