REALTIME VIDEO FRAME INTERPOLATION USING ONE-STEP DIFFUSION SAMPLING

Yongrui Ma[1,2], **Shijie Zhao**[1]*, **Mingde Yao**[2,4], **Junlin Li**[1], **Li Zhang**[1], **Xiaohong Liu**[5], **Qi Dou**[3], **Jinwei Gu**[3], **Tianfan Xue**[2,4]*
[1]ByteDance Inc. [2]CUHK MMLab [3]CUHK [4]CPII under InnoHK [5]SJTU

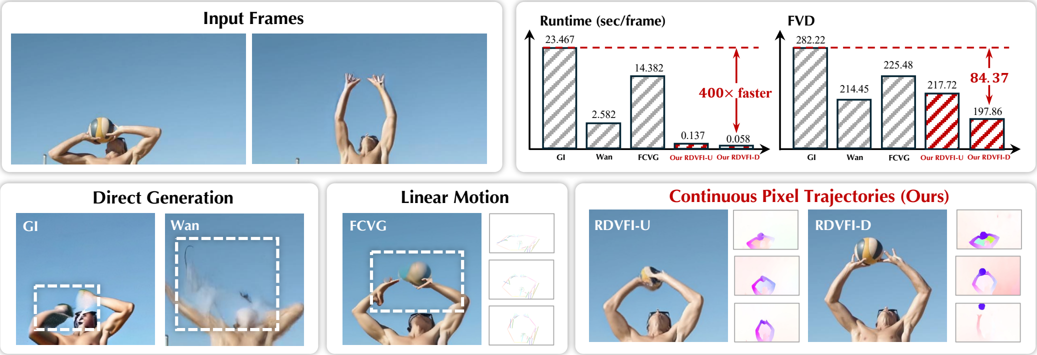

Figure 1: Video frame interpolation results of the proposed Unet-based RDVFI-U and DiT-based RDVFI-D with comparisons to state-of-the-art methods. RDVFI produces the most visually pleasant and the best numeric results with a 400× acceleration, owing to more proper modeling of the intermediate motions using the high-order continuous pixel trajectories, compared with previous direct frame-generation methods, and linear motion controls.

ABSTRACT

Video Frame Interpolation (VFI) involving large, complex motions remains a significant challenge due to the difficulty of modeling diverse pixel trajectories from limited inputs. Traditional methods struggle with low-order approximations, and recent Latent Video Diffusion Models (LVDM) improve it through a conditional generation modeling. Still, current LVDMs often prioritize pixel fidelity over motion coherence in their reconstruction objective, leading to artifacts in extreme motion scenarios. To address this, we propose RDVFI, a novel approach that leverages an LVDM to generate sparse latent keyframes which define high-order, continuous pixel trajectories. The estimated continuous pixel trajectories accurately index pixel movements from inputs to arbitrary timestamps, generating optical flows to warp input pixels into the target frame. By decoupling sequence motion generation from high-resolution rendering, RDVFI operates on a fixed, lower resolution, and fewer diffusion sampling steps, introducing significant efficiency gains. Extensive experiments demonstrate that RDVFI achieves state-of-the-art visual and numerical performance, with over 75% of viewers selecting it as the best method in terms of motion and frame quality compared to leading baselines. Furthermore, RDVFI is the first LVDM-based VFI method to achieve real-time performance (17 FPS at 1024×576), offering a $\times 44$ acceleration over the current state-of-the-art and also robustly handling challenging motions. Please visit the project page for code and more visualization samples.

1 INTRODUCTION

Video Frame Interpolation (VFI) aims to predict intermediate frames between two input ending frames. It serves as a foundational component in diverse fields, including high-frame-rate photogra-

*Corresponding authors.

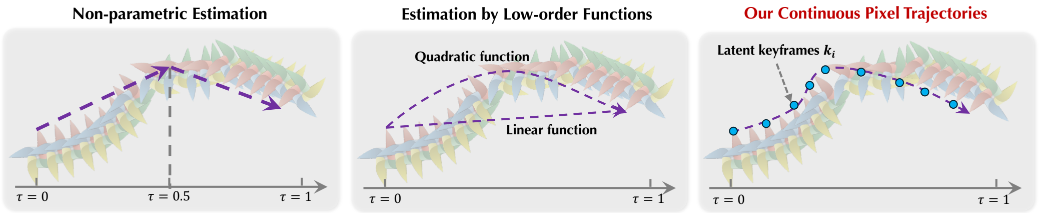

Figure 2: Comparisons between motion estimation methods for VFI. Previous non-parametric methods (Huang et al., 2022; Reda et al., 2022; Li et al., 2023; Danier et al., 2024; Lew et al., 2025) and low-order approximations (linear (Jiang et al., 2018; Bao et al., 2019)/quadratic (Xu et al., 2019; Liu et al., 2020)) struggle to capture large and nonlinear motions like the above rotating windmill. The proposed RDVFI solves continuous pixel trajectories via high-order functions determined by sparse latent keyframes ($\{k_i\}_{i=1}^{N}$) from the LVDM. Therefore, RDVFI can interpolate to more diverse and complex real-world motions that previous methods fail to capture.

phy (Xiang et al., 2020; Gao et al., 2022; Ma et al., 2024), cinematic post-production (Kokaram et al., 2020; Briedis et al., 2021), and scientific visualization (Priessner et al., 2024; Joshi et al., 2025). Current methods often interpolate by estimating optical flows between intermediate and input frames to warp input pixels to each interpolated frame. As illustrated in Fig. 2, standard non-parametric methods (Huang et al., 2022; Reda et al., 2022; Li et al., 2023; Guo et al., 2024; Danier et al., 2024; Lew et al., 2025) predict flows between one middle and input frames directly in a data-driven manner, optionally assisted by an image diffusion model. While straightforward, these approaches struggle to resolve nonlinear, non-rigid motions. Some methods approximate pixel trajectories using rigid motion priors, such as linear (Jiang et al., 2018; Bao et al., 2019) or quadratic (Xu et al., 2019; Liu et al., 2020) functions. However, these low-order approximations lack the ability to model the complex and ambiguous real-world dynamics.

Recently, Latent Video Diffusion Models (LVDM) have revolutionized generative tasks by their ability to produce high-fidelity video content (Blattmann et al., 2023; Guo et al., 2023; Wang et al., 2025). Inspired by this, state-of-the-art approaches also (Zhu et al., 2024; Wang et al., 2024; 2025) reformulate VFI as a conditional generation problem using LVDMs. Still, despite their ability to synthesize plausible content, these models face two critical limitations. First, they rely on standard pixel-reconstruction objectives that prioritize local texture approximation over global motion coherence. This often leads to severe structural artifacts, such as ghosting and object deformation, as seen in Fig. 1. Second, the iterative diffusion sampling required for high-frequency detail imposes a heavy computational burden. Although current distillation techniques can accelerate these methods by reducing the number of sampling steps, they may aggravate the structural artifacts in interpolated frames, especially when dealing with nonlinear and non-rigid motions.

In this paper, we propose RDVFI, a novel VFI framework that formulates large and complex motions as high-order functions, with coefficients determined by an LVDM. Previous methods model pixel movements based solely on input frames, limiting them to linear (Jiang et al., 2018; Bao et al., 2019), quadratic (Xu et al., 2019; Liu et al., 2020), or nonparametric (Huang et al., 2022; Sim et al., 2021; Li et al., 2023; Danier et al., 2024; Lew et al., 2025) formulations, as shown in the first two examples in Fig. 2. Instead, by incorporating an LVDM to generate additional latent keyframes, RDVFI enables high-order pixel trajectories, which capture real-world dynamics more accurately than previous approaches (see Fig. 2), as shown in the third example of Fig. 2. RDVFI then samples the estimated continuous pixel trajectories for optical flows to warp input pixels for interpolation. Because our LVDM primarily focuses on generating accurate intermediate motions that lack high-frequency details, the proposed RDVFI can interpolate to complex motions with superior temporal stability via one-step diffusion sampling, introducing significant efficiency gains in Fig. 1.

RDVFI offers three primary advantages. First, by shifting the LVDM's objective from frame synthesis to motion modeling, we achieve superior temporal coherence and accuracy, as shown in Fig. 1. Second, because the predicted trajectories represent underlying structural motion rather than fine-grained texture, one-step sampling is sufficient to capture the necessary dynamics, significantly reducing inference latency. Third, as motion can be computed from sparse and low-resolution keyframes, RDVFI further accelerates the inference by offloading major computation on a low-resolution and sparse input, and uses a relatively lightweight module for dense-frame and full-resolution computation. This design makes RDVFI extremely efficient for high-resolution input.

Extensive evaluations demonstrate that RDVFI successfully interpolates large-scale motions with over **50**× acceleration of the leading baseline, Wan (Wang et al., 2025). On a single A100 GPU, RDVFI processes 1024×576 resolution at 17 FPS. Our core contributions are:

- We present RDVFI, a VFI framework that solves large, complex motions using high-order continuous pixel trajectories and coefficients determined by a one-step LVDM. To our knowledge, RDVFI is the first work to solve motion in VFI using high-order functions or one-step LVDM.
- We introduce an implicit motion regularization term that **enforces physical and temporal consistency**, allowing for accurate interpolation of complex, non-linear trajectories.
- We demonstrate significant gains in both perceptual quality and computational efficiency, effectively bridging the gap between diffusion-based generative power and practical deployment requirements.

2 Related Work

2.1 Video Frame Interpolation

Non-generative Methods Conventional VFI methods (Jiang et al., 2018; Bao et al., 2019; Xu et al., 2019; Liu et al., 2020; Huang et al., 2022; Reda et al., 2022; Li et al., 2023) primarily rely on regression-based objectives. These are categorized by their underlying motion priors. Early approaches assume pixels follow rigid, predefined trajectories, such as linear (Jiang et al., 2018; Bao et al., 2019) or quadratic (Xu et al., 2019; Liu et al., 2020) models. However, these parametric assumptions fail to capture the stochastic and non-linear nature of real-world dynamics. Subsequent data-driven strategies (Huang et al., 2022; Reda et al., 2022; Li et al., 2023; Guo et al., 2024) predict optical flows between specific timestamps and input frames. While effective at reducing warping artifacts, these methods remain brittle when confronted with the large and complex motions.

Generative-based Methods Diffusion Models (DMs) (Ho et al., 2020) have demonstrated superior performance in solving ill-posed inverse problems for image (Dhariwal & Nichol, 2021; Ramesh et al., 2022; Saharia et al., 2022) and video synthesis (Hong et al., 2023; Blattmann et al., 2023; Yang et al., 2024; Lin et al., 2024; Wang et al., 2025). Recent studies (Danier et al., 2024; Lew et al., 2025; Zhu et al., 2024; Wang et al., 2024; 2025) leverage DMs to improve interpolation fidelity. For instance, LDMVFI (Danier et al., 2024) and MoMo (Lew et al., 2025) use image-based DMs to refine flow estimation. However, they remain optimized for relatively simple motion. More recent state-of-the-art frameworks like GI (Wang et al., 2024), FCVG (Zhu et al., 2024), and Wan (Wang et al., 2025) treat VFI as a conditional generation task using Latent Video Diffusion Models (LVDMs). Despite the progress, these methods suffer from a “pixel-fidelity bias”: their training objectives focus on per-pixel reconstruction rather than explicit motion consistency and accuracy. Consequently, they often produce frames with rich frame details but poor temporal coherence and unnatural motions.

2.2 Enhancing Motion in LVDMs

Several attempts (Zhu et al., 2024; Chefer et al., 2025; Shi et al., 2024) have been made to improve motion modeling within LVDMs. FCVG (Zhu et al., 2024) utilizes linearly interpolated sparse matching to guide motions, yet it remains limited by the reliability of the matching algorithm and the simplicity of its motion model. VideoJAM (Chefer et al., 2025) forces LVDMs to jointly predict frames and flow visualizations; however, this multitask approach often leads to frame degradation due to a distribution mismatch between natural imagery and flow visualizations. Alternatively, Motion-I2V (Shi et al., 2024) decouples flow from frame generation using separate LVDMs. However, it still fails on complex motions because flow estimation degrades with increasing temporal distance, making it impossible to generate accurate ground truth for training. Unlike these methods, RDVFI introduces a robust motion representation by predicting coefficients for high-order continuous pixel trajectories, thereby enabling a more expressive, physically consistent motion space.

2.3 Diffusion Acceleration

Accelerating diffusion inference is critical for practical VFI. Current techniques include rectified flow (Liu et al., 2022; 2023), adversarial training (Lin et al., 2025; Zhang et al., 2024), and score

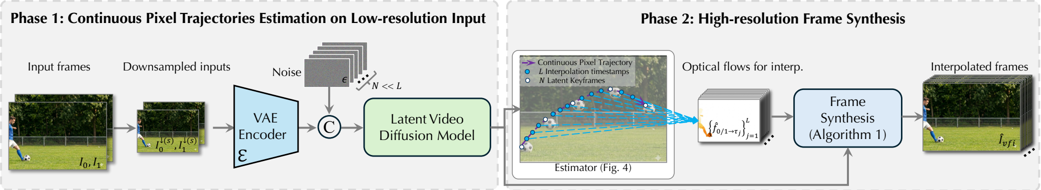
Phase 1: Continuous Pixel Trajectories Estimation on Low-resolution Input
Input frames
I_0, I_1
Downsampled inputs
$I_0^{\downarrow(s)}, I_1^{\downarrow(s)}$
Noise
ϵ
$N << L$
VAE Encoder
$\mathcal{E}$
C
Latent Video Diffusion Model
Phase 2: High-resolution Frame Synthesis
Continuous Pixel Trajectory
L Interpolation timestamps
N Latent Keyframes
Estimator (Fig. 4)
Optical flows for interp.
$\{\hat{f}_{0/1\to\tau_j}\}_{j=1}^{L}$
Frame Synthesis (Algorithm 1)
Interpolated frames
$\hat{I}_{vfi}$

Figure 3: Overall framework of the proposed RDVFI. Our RDVFI interpolates intermediate frames $\hat{I}_{vfi}$ in two phases: continuous pixel trajectories estimation and frame synthesis. **Phase 1** determines the coefficients of high-order continuous pixel trajectories, which index pixel offsets at arbitrary intermediate timestamps, using an LVDM. **Phase 2** samples the estimated trajectories for optical flows that warp input pixels to each interpolation timestamp. RDVFI then fuses the warped frames for interpolated frames with fine-grained details using the frame synthesis module.

distillation (Wang et al., 2023; Yin et al., 2024), to guide a lightweight, fewer-step student network to behave like a robust, multi-step teacher model. While one-step generation has been achieved in text-to-image (Liu et al., 2023; Yin et al., 2024) and video synthesis (Lin et al., 2025; Zhang et al., 2024), these distillation-based methods are difficult to adapt to VFI for two reasons. First, existing conditional frame-generation LVDMs may suffer from fewer diffusion sampling steps, aggravating their artifacts for large and complex motions. Second, the auxiliary networks required for distillation introduce significant training overhead. To this end, RDVFI does not involve distillation techniques. We demonstrate that by re-tasking the LVDM to predict motion trajectories rather than raw pixels, one-step sampling is inherently sufficient to resolve motion ambiguity while maintaining high-fidelity results through warping.

3 Method

We introduce RDVFI, a framework designed to bridge the gap between generative fidelity and motion coherence in VFI. We first outline the overall framework in Section 3.1. This is followed by our continuous pixel trajectory modeling in Section 3.2, the one-step LVDM in Section 3.3, the synthesis network in Section 3.4, and finally our two-stage training strategy in Section 3.5.

3.1 Overall Framework

Given two endpoint frames I_0, I_1, RDVFI synthesizes a sequence of intermediate frames $\{\hat{I}_{\tau_j}\}_{j=1}^{L}$ at arbitrary timestamps $\tau_j \in (0, 1)$. The pipeline operates in two distinct phases: continuous pixel trajectory estimation at a fixed, relatively lower resolution, such as 448×256, and high-resolution frame synthesis (*e.g.*, at 1024×576), as shown in Fig. 3.

In the first phase, a Latent Video Diffusion Model (LVDM) generates a set of sparse latent keyframes $\{\hat{z}_{k_i}\}_{i=1}^{N}$ via one-step sampling. These keyframes serve as temporal anchors to determine the coefficients of continuous pixel trajectories. Because the LVDM operates at a reduced spatio-temporal resolution and models motion instead of texture, it achieves significant computational efficiency. In the second phase, RDVFI samples the estimated trajectories at each interpolation timestamp $\{\tau_j\}_{j=1}^{L}$, and we derive dense optical flows for each intermediate frame. These generated optical flows are used to warp the input frames, following a frame synthesis module to fuse and produce the final high-resolution output $\hat{I}_{vfi}$.

3.2 Continuous Pixel Trajectories

We define continuous pixel trajectories as functions $\{f_{0\to\tau}, f_{1\to\tau}\}$ that map pixel offsets from I_0 or I_1 to any $\tau \in (0, 1)$. To ensure accuracy across large displacements, we decompose complex motions between input frames into small and easy-to-estimate components and solve them iteratively, as shown in Fig. 4. To this end, we first estimate motions across consecutive frames by:

$$f_{k_i \to k_{i+1}} = \phi_1(\hat{z}_{k_i}, \hat{z}_{k_{i+1}}). \tag{1}$$

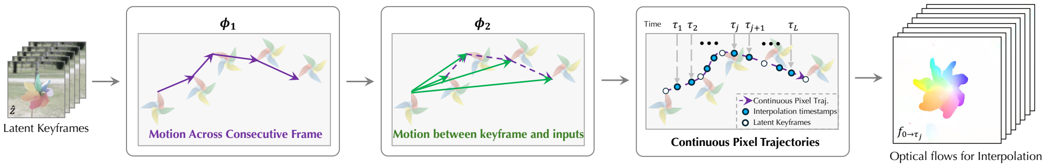
$\hat{z}$
Latent Keyframes
ϕ_1
Motion Across Consecutive Frame
ϕ_2
Motion between keyframe and inputs
Time
τ_1 τ_2 τ_j τ_{j+1} τ_L
Continuous Pixel Traj.
Interpolation timestamps
Latent Keyframes
Continuous Pixel Trajectories
$f_{0\to\tau_j}$
Optical flows for Interpolation

Figure 4: The proposed continuous pixel trajectories estimator. The proposed RDVFI decomposes the large and complex motions between intermediate and input frames into small and simple motions across consecutive frames guided by latent keyframes generated by the LVDM. RDVFI then recursively fuses these motion components to compute optical flows between sparse latent keyframes and input frames. RDVFI fits a high-order continuous pixel trajectory functions using estimated optical flows, which can flexibly sample optical flows to warp input pixels to any intermediate time.

The estimated motion components are fused recursively to compute optical flows between each keyframe and input frames, following:

$$f_{0\to k_{i+1}} = \phi_2(f_{0\to k_i}, f_{k_i\to k_{i+1}}, z_0, \hat{z}_{k_{i+1}}). \tag{2}$$

In this formulation, ϕ_1 and ϕ_2 are convolutional neural networks in the supplementary Section B.2. We estimate the backward continuous pixel trajectories using the same network and weights, but in a reversed direction from $\tau = 1$ to $\tau = 0$. In addition to latent keyframes $\{k_i\}_{i=1}^N$, we also define k_0, k_{N+1} to be encoded latent input frames and calculate the continuous pixel trajectories from $i = 0$ and $i = N + 1$ to $i = N$ and $i = 1$, respectively. This iterative refinement only updates a small and easy-to-estimate pixel offsets to previous flow results, allowing RDVFI to resolve large-scale motions more effectively and efficiently than direct-estimation baselines. Finally, we fit these keyframe optical flows using cubic convolution (Keys, 2003) to provide optical flows for all interpolation.

3.3 One-step Latent Video Diffusion Model

The continuous pixel trajectories introduced above are computed from keyframes, and we use the Latent Video Diffusion Model (LVDM) to generate those keyframes.

To illustrate that, we first revisit the preliminaries of LVDM. LVDMs generate video frames from noise based on input conditions in a low-resolution latent space, following a VAE decoder $\mathcal{D}(\cdot)$ to synthesize high-resolution video frames. In VFI, the LVDM first encodes the two input frames, I_0 and I_1, into the latent frames z_0 and z_1. These latent frames are then used as generation conditions. To learn the generation process, the LVDM degrades ground truth latent frames z with noise $\epsilon^t \sim \mathcal{N}(0, I)$, generating noisy latent frames by $z^t = \alpha^t z + \sigma^t \epsilon^t$. Here, α^t and σ^t are parameters determining how much noise is added. The diffusion timestep $t \in (0, T)$, with $t = T$, corresponds to pure noise. The LVDM uses a denoising network $\mathcal{F}_\theta(z^t; t, z_0, z_1)$ to estimate the added noise. The training objective is:

$$\mathcal{L}_\theta(t) = ||\mathcal{F}_\theta(z^t; t, z_0, z_1) - v^t||, \tag{3}$$

where $v^t = \alpha^t \epsilon^t - \sigma^t z$ is called the velocity of diffusion. After convergence, the LVDM recursively denoises randomly-sampled noise to generate noise-free intermediate latent frames $\hat{z}$. Because these LVDMs directly generate intermediate frames, they require tedious diffusion sampling to capture the fine-grained details of each frame, resulting in a high computation cost that is difficult to accelerate.

Unlike previous methods Zhu et al. (2024); Wang et al. (2024; 2025) that utilize the multi-step LVDM to generate all interpolated frames, our RDVFI only uses one-step LVDM to generate a few keyframes in a continuous pixel trajectory, resulting in significant acceleration. Our design is based on the fact that input frames in VFI have already provided the most desired intermediate appearances, and the core task is determining how to extract pixels from these input frames to generate intermediate frames. Because our LVDM only determines motions, it does not require dense diffusion sampling for fine-grained frame details, where one-step diffusion sampling can generate reasonable results. Furthermore, because the number of keyframes N is often much smaller than the number of interpolated frames L, and generated motions can be flexibly resized to different resolutions, our LVDM can sample at a **fixed and relatively smaller** spatial-temporal frame resolution. This yields further acceleration and facilitates diffusion training. Please find the supplementary Section B.1 for the network details.

Algorithm 1: RDVFI Frame synthesis.

```
Require: Input frames $I_0$ and $I_1$, interpolation time $\tau = \{\tau_j\}_{j=1}^L$, estimated continuous pixel
    trajectories $\{f_{0\to\tau}, f_{1\to\tau}\}$
Ensure: Interpolated frames $\{\hat{I}_j\}_{j=1}^L$
for j=1, 2, ..., L do
    Sample optical flows $\{f_{0\to\tau_j}, f_{1\to\tau_j}\}$ from $\{f_{0\to\tau}, f_{1\to\tau}\}$
    Warp input frames $I_{0\to\tau_j} = Warp(I_0, f_{0\to\tau_j}), I_{1\to\tau_j} = Warp(I_1, f_{1\to\tau_j})$
    Find the nearest keyframe timestep, satisfying $k_i < \tau_j < k_{i+1}, k_0 = 0, k_{l+1} = 1$
    Sample optical flows $\{f_{\tau_i\to\tau_j}, f_{\tau_{i+1}\to\tau_j}\}$ from $\{f_{0\to\tau}, f_{1\to\tau}\}$
    Warp latent keyframes $z_{k_i\to\tau_j} = Warp(z_{k_i}, f_{k_i\to\tau_j}), z_{k_{i+1}\to\tau_j} = Warp(z_{k_{i+1}}, f_{k_{i+1}\to\tau_j})$
    $\hat{I}_j = \phi^s(I_0, I_1, I_{0\to\tau_j}, I_{1\to\tau_j}, f_{0\to\tau_j}, f_{1\to\tau_j}, z_{k_i\to\tau_j}, z_{k_{i+1}\to\tau_j})$
end for
```

3.4 Frame Synthesis

Given the estimated pixel trajectory introduced above, RDVFI finally generates all high-resolution dense intermediate frames. As shown in Algorithm 1, RDVFI first samples the estimated continuous pixel trajectories $\{f_{0\to\tau}, f_{1\to\tau}\}$ at different interpolation times $\tau = \{\tau_j\}_{j=1}^L$ to generate optical flows $\{f_{0\to\tau_j} f_{1\to\tau_j}\}_{j=1}^L$. RDVFI then warps input frames I_0, I_1 with sampled optical flows, producing bi-directional warped frames $\{I_{0\to\tau_j}, I_{1\to\tau_j}\}$. Unlike previous non-generative methods that directly fuse the warped frames, we further incorporate latent keyframes generated through one-step LVDM sampling to provide appearance changes that cannot be modeled by optical flows, such as dynamic textures, lightning changes, and occlusions. For this target, we sample optical flows $\{f_{k_i\to\tau_j}, f_{k_{i+1}\to\tau_j}\}$ between the latent keyframes $z_{k_i}, z_{k_{i+1}}$ that satisfy $k_i \leq \tau_j \leq k_{i+1}$. We further warp these latent frames to time $\tau = \tau_j$, producing warped latent frames $\{z_{k_i\to\tau_j}, z_{k_{i+1}\to\tau_j}\}$. RDVFI then fuses warped frames $\{I_{0\to\tau_j}, I_{1\to\tau_j}\}$ and latent keyframes with a neural network $\phi^s(\cdot)$ to produce the interpolated frames. Please find the supplementary Section B.3 for network details.

3.5 Training Process

As shown in Fig. 8, we train our RDVFI in two stages: motion-guided latent-to-frame generation and one-step diffusion training.

Motion-guided Decoding RDVFI trains the continuous pixel trajectories estimator to predict accurate optical flows for interpolation using given latent keyframes. Due to the lack of accurate ground truth, RDVFI learns this process in an unsupervised manner with the loss on interpolated frames. Existing research (Xue et al., 2019) has demonstrated that these task-oriented optical flows can enhance downstream tasks and have been widely used in recent VFI methods (Jiang et al., 2018; Bao et al., 2019; Xu et al., 2019; Liu et al., 2020; Huang et al., 2022; Reda et al., 2022; Li et al., 2023; Danier et al., 2024). Specifically, RDVFI first selects N keyframes $I^{\downarrow(t)}$ by frame skipping and then downsamples selected keyframes to a relatively low resolution $I^{\downarrow(s,t)}$. RDVFI uses a motion-guided decoding network to synthesize high-resolution frames $\hat{I}_{rec}$ from encoded ground truth latent keyframes z. The training objective is formulated as follows:

$$\mathcal{L}_{rec}(\hat{I}_{rec}, I) = w_1 L_{lpips}(I, \hat{I}_{rec}) + w_2 ||I - \hat{I}_{rec}||_2, \tag{4}$$

In this formulation, w_1 and w_2 denote the loss weights, which are set to 0.2 and 1, respectively.

One-step Diffusion Training In this stage, RDVFI trains the LVDM to generate sparse latent keyframes for interpolation. To achieve this, the continuous pixel trajectories estimator and synthesis network are frozen, and only the parameters in the LVDM are updated. Rather than requiring the LVDM to generate noise-free latent keyframes directly from pure noise, we train the LVDM with randomly sampled $t \in (0, T)$ for higher stability. We linearly increase the diffusion timestep t by adjusting the lower bound for random sampling according to the number of training steps. In addition to the original pixel reconstruction loss in the latent space (Equation 3), the reconstruction loss on the interpolated frames (Equation 4) is also employed, resulting in the following:

$$\mathcal{L}_{vfi}(I, \hat{I}_{vfi}) = \lambda_1 \mathcal{L}_\theta(t) + \lambda_2 \mathcal{L}_{rec}(\hat{I}_{vfi}, I). \tag{5}$$

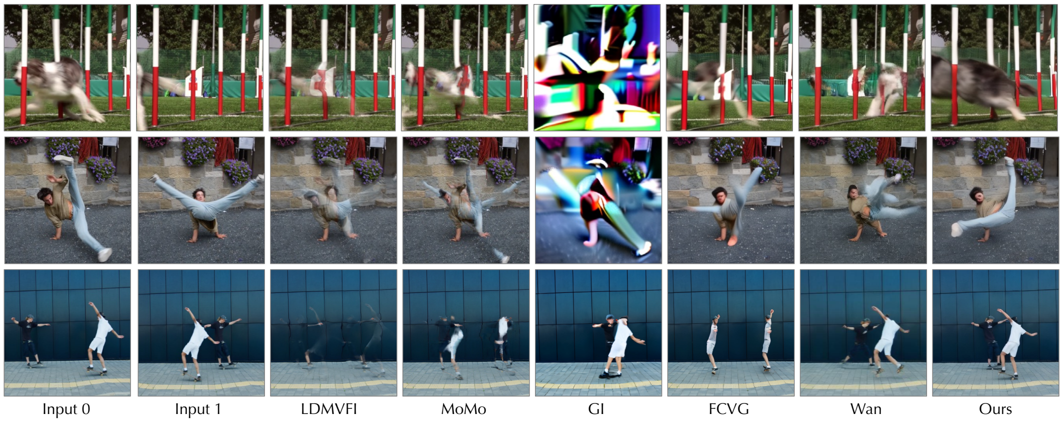

Figure 5: Visual comparison results on DAVIS-7 (Jain et al., 2024) (the top row), FCVG (Zhu et al., 2024) (the middle row), and our self-collected evaluation dataset (the bottom row). Our RDVFI-D method consistently outperforms existing diffusion-based VFI methods with less blurring, ghosting effects, color shifts, and fractions.

In this context, λ_1 and λ_2 are coefficients that balance each loss term, set to 1.0 and 0.5, respectively. Since our motion-guided decoding process involves continuous pixel trajectories estimation, the reconstruction loss term can serve as an implicit motion regularization term in the diffusion training and is essential for generating high-quality frames.

4 Experimental Results

4.1 Implementation Details

To verify that our RDVFI can adopt different pretrained LVDMs, we trained two versions: the Unet-based RDVFI-U, which builds upon 1.5B SVD (Blattmann et al., 2023), and the DiT-based RDVFI-D based on Wan2.1-Fun-1.3B-InP (Wang et al., 2025). Both of them use seven keyframes. During the training, we randomly resize images to one of four resolutions: $(448 \times 256, 576 \times 320, 1024 \times 576, 1280 \times 720)$. The LVDM and the continuous pixel trajectories estimator operate at the 448×256 resolution by resizing input images. The motion-guided decoding module then generates intermediate frames at the input resolutions with up-scaled optical flows. We use a fixed learning rate of 2e-5 for all experiments. We adopt eight A100-80G GPUs and a total batch size of 8. We train for 400K iterations in the first stage and 800K in the second. Please find the supplementary Section C for more details.

4.2 Benchmarking

Selected Methods and Setting We compare the proposed RDVFI with several state-of-the-art methods, including conventional non-generative ones, such as RIFE (Huang et al., 2022), FILM (Reda et al., 2022), and AMT (Li et al., 2023), as well as the diffusion-based ones, including LDMVFI (Danier et al., 2024), MoMo (Lew et al., 2025), GI (Wang et al., 2024), FCVG (Zhu et al., 2024), and Wan 1.3B InP (Wang et al., 2025). Because methods may require different training strategies for optimal performance, we utilize the released weights without further tuning, as the authors have already best optimized them. We evaluate Wan (Wang et al., 2025) with blank text input. We do not compare with any distillation-based VFI method because current methods still focus on solving large and complex motions, and we have not found a baseline method that specifically involves VFI distillation.

Results We report comparison results between the proposed RDVFI and baseline methods in Table 1, Fig. 6, and Fig. 5. The results show that our RDVFI outperforms existing baseline methods across the reported benchmark datasets. Our DiT-based method RDVFI-D outperforms the SVD-based version, RDVFI-U, benefiting from a more advanced backbone network. Image diffusion-based interpolation methods (Danier et al., 2024; Lew et al., 2025) cannot create complex motions between intermediate and input frames, thus producing severe ghosting effects. Directly generating

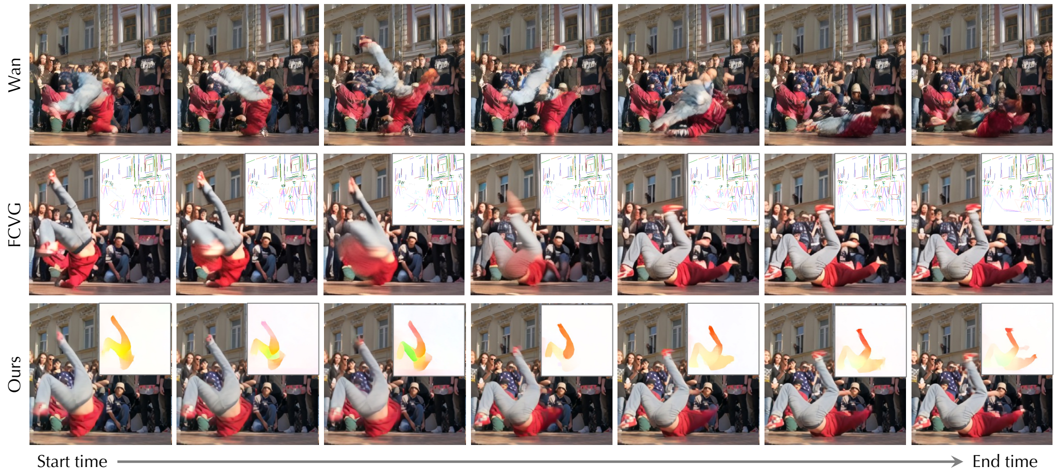

Figure 6: Sequence interpolation comparison. Our RDVFI-D can correctly estimate the continuous pixel trajectories between input frames from diffusion outputs and thus interpolate with correct motion and superior visual quality, compared with the direct generation method Wan (Wang et al., 2025) and the linearly-controlled FCVG (Zhu et al., 2024).

Table 1: Numeric comparison results on three benchmark datasets :DAVIS-7 (Jain et al., 2024), FCVG (Zhu et al., 2024) and our Pixels evaluation set. The best and the second-best results are highlighted by **bold** and underline. The proposed RDVFI outperforms existing baseline methods on most metrics.

Methods	DAVIS-7 (Jain et al., 2024)			FCVG (Zhu et al., 2024)			Pixels		
	LPIPS↓	FID↓	FVD↓	LPIPS↓	FID↓	FVD↓	LPIPS↓	FID↓	FVD↓
Non-Generative Methods									
AMT (Li et al., 2023)	0.254	34.65	234.50	0.224	44.74	375.00	0.361	41.36	378.86
RIFE (Huang et al., 2022)	0.258	23.98	240.04	0.247	39.01	366.14	0.278	31.10	207.82
FILM (Reda et al., 2022)	0.271	30.16	214.80	0.241	39.82	279.08	0.251	27.06	158.68
GIMM-VFI (Guo et al., 2024)	0.276	24.10	245.02	0.231	38.71	292.08	0.269	29.21	192.37
Image Diffusion-based Methods									
LDMVFI (Danier et al., 2024)	0.276	22.10	245.02	0.228	37.74	371.49	0.287	31.36	211.70
MoMo (Lew et al., 2025)	0.268	23.67	240.09	0.207	33.59	261.37	0.269	27.31	230.19
Video Diffusion-based Methods									
TRF (Feng et al., 2024)	0.270	29.12	230.12	0.331	45.37	305.88	0.301	35.31	279.65
GI (Wang et al., 2024)	0.267	27.71	211.47	0.334	43.08	282.22	0.273	33.27	251.38
FCVG (Zhu et al., 2024)	0.266	25.96	207.17	0.266	31.24	225.48	0.257	24.51	137.57
Wan (Wang et al., 2025)	0.323	26.97	248.14	0.223	28.52	214.45	0.261	22.48	131.22
Our RDVFI-U	0.260	23.65	201.49	0.220	27.03	217.72	0.261	23.71	129.33
Our RDVFI-D	**0.251**	**21.17**	**189.37**	**0.201**	**19.98**	**197.86**	**0.253**	**19.48**	**119.21**

intermediate frames with Latent Video Diffusion Models (LVDM) may introduce severe degradations (Wang et al., 2024), due to the mismatched interpolation results from separate forward and backward interpolation from two image-to-video SVD (Blattmann et al., 2023) models; FCVG (Zhu et al., 2024) attempts to mitigate such misalignment by involving linear motion controls, however, sacrificing motion generation ability of VDMs and utilizing them as shaders. As shown in Fig. 6, FCVG (Zhu et al., 2024) produces severe degradations when the matching-based linear motion controller does not distinguish the falling boy. Sharing the same backbone, however, our RDVFI-D significantly outperforms Wan (Wang et al., 2025). Our motion-guided decoding pipeline generates intermediate frames by warping, which forces the diffusion model to create latents that can accurately restore motions across frames. Thus, our interpolation is smoother and more stable, producing the fewest artifacts compared to other LVDM-based solutions. Please find our supplementary video demo and Section D for more dynamic and static visual comparisons.

4.3 Efficiency Comparisons

We report detailed efficiency metrics in Table 2. As we can observe, our RDVFI is the fastest diffusion-based interpolation method, which can interpolate at real time (17 FPS) at the resolution of 1024 × 576. Although a non-generative baseline method, RIFE (Huang et al., 2022), is slightly

Table 2: We compare the inference efficiency between our RDVFI and existing interpolation baseline methods, including AMT (Li et al., 2023), RIFE (Huang et al., 2022), FILM (Reda et al., 2022), LDMVFI (Danier et al., 2024), MoMo (Lew et al., 2025), GI (Wang et al., 2024), FCVG (Zhu et al., 2024), and Wan (Wang et al., 2025), on the FCVG (Zhu et al., 2024) dataset for ×24 interpolation at the 1024×576 resolution. We **bold** and underline the best and the second-best results, respectively. Our RDVFI is the fastest and most memory-efficient diffusion-based interpolation method. Although RIFE (Huang et al., 2022) is slightly faster than our RDVFI, our RDVFI outperforms it with a clear margin for challenging motions.

Method	GPU Mem. (GB)	Runtime (sec.)	FID↓	FVD↓
Non-Generative Methods				
AMT Li et al. (2023)	13.5	0.210	44.74	375.00
RIFE Huang et al. (2022)	**1.4**	**0.025**	39.01	366.14
FILM Reda et al. (2022)	8.0	0.830	39.82	279.08
Image Diffusion-based Methods				
LDMVFI Danier et al. (2024)	21.7	1.563	37.74	371.49
MoMo Lew et al. (2025)	<u>3.9</u>	0.157	33.59	261.37
Video Diffusion-based Methods				
GI Wang et al. (2024)	23.5	29.613	45.37	282.22
FCVG Zhu et al. (2024)	27.6	14.381	31.24	225.48
Wan Wang et al. (2025)	18.0	2.579	28.52	<u>214.45</u>
Our RDVFI-U	14.2	0.137	<u>27.03</u>	217.72
Our RDVFI-D	13.1	<u>0.057</u>	**19.98**	**197.86**

Table 3: Ablation study on different resolutions (3a) and network designs (3b).

(a) Because we perform the diffusion sampling at fixed and relatively smaller resolution than the interpolated frame, the runtime of the algorithm does not change a lot with the resolution increase.

Resolution	Runtime (ms)	
	One-step Wan	RDVFI-D
576 × 320	103.3	55.3
1024 × 576	267.7	58.01
1280 × 720	314.3	63.49

(b) Our RDVFI-D consistently outperforms all ablation experiments, showing the best interpolation accuracy in terms of both reconstruction and perceptual metrics.

Setting	LPIPS↓	FID↓	FVD↓
Direct Warping	0.287	42.33	327.11
L loss	0.269	33.29	281.37
L+P-L2 loss	0.251	29.37	267.42
RDVFI-D	0.224	23.71	209.38

faster than our RDVFI, it cannot deal with large complex motions and introduces severe visual degradations. In contrast, our RDVFI can efficiently and effectively interpolate large complex motions, outperforming RIFE with a clear margin, as shown in Table 1, Fig. 5, and Fig. 6.

4.4 Ablation Study

Motion-guided Decoding Pipeline We compare the runtime of one-step Wan (Wang et al., 2025), which directly reduces the sampling step of the pretrained Wan model, and our RDVFI-D. These two models share the same diffusion backbone and sampling step with two core differences. First, Wan's resolution changes with interpolated frames, whereas our RDVFI-D operates at a fixed, relatively lower resolution. Second, Wan generates high-resolution frames with the VAE decoder and our RDVFI-D scales up the estimated optical flows and warps high-resolution inputs for interpolation. In addition, we claim that iterative motion fusion for continuous pixel trajectories estimation is one of our key contributions to accurate motion estimation. Thus, we degrade our motion-guided decoding pipeline by removing iterative flow fusion and estimating pixel offsets between each keyframe and the input frames from scratch via regression. We define this setting as "Direct Warping".

Training Strategy To validate our latent-pixel training strategy, we degrade our training objective by removing the pixel-space perceptual loss only, formatting"L+P-L2 loss", and all pixel-wise loss, defined as "L loss".

As shown in Table 3a and Table 3b, our RDVFI-D consistently outperforms all degraded versions on efficiency or accuracy. Benefiting from our motion-guided decoding, RDVFI-D can interpolate at different resolutions at the same diffusion sampling cost, resulting in a significant efficiency gain compared with traditional diffusion networks, as shown in Table 3a. Table 3b also shows the effectiveness of our flow fusion strategy and training objective design. For fair comparison, we fine-tune

Table 4: Human evaluation results. Most viewers select our RDVFI-D as the best-performed interpolation method when compared with existing diffusion-based state-of-the-art methods in terms of both generated motion, frame details, and overall quality. Our RDVFI-D surpasses the second-best method, Wan (Wang et al., 2025), for 60% more.

Direction	LDMVFI	MoMo	GI	FCVG	Wan	Ours
Motion	0.9%	0.8%	1.1%	4.1%	16.2%	**76.9%**
Details	0.3%	0.9%	1.5%	9.2%	13.7%	**74.4%**
Overall	0.5%	0.6%	1.0%	8.0%	14.2%	**75.7%**

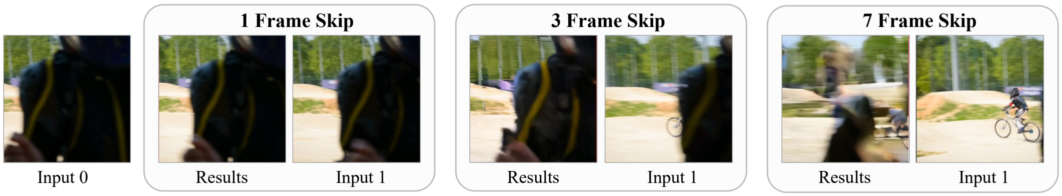

Figure 7: Our RDVFI can robustly interpolate when input frames have consistent objects and enough overlapping regions, such as the 1-frame and 3-frame skip cases. However, when the overlapped regions become minimal, as the ending input frame is produced by 7 frames skipped, the algorithm cannot generate coherent and accurate intermediate optical flows to warp input pixels for interpolation, resulting in severe degradations.

each ablation experiment and report our results at the same iterations. Please find Section A for the ablation study on the keyframe number.

4.5 Human Evaluation

To better illustrate the visual gains of the proposed RDVFI over existing solutions, we also include a human evaluation study involving 25 participants. We form a questionnaire with ten groups of videos, each group contains six video clips from the proposed RDVFI-D and five state-of-the-art methods, including Wan (Wang et al., 2025), FCVG (Zhu et al., 2024), GI (Wang et al., 2024), MoMo (Lew et al., 2025), and LDMVFI (Danier et al., 2024). The respondents are expected to select the best video clip from each group based on the generated motion quality, frame details, and overall quality. As shown in the Table 4, the proposed RDVFI-D significantly outperforms all state-of-the-art methods, with more than 75% of respondents selecting our RDVFI-D as the best-performing method, surpassing the second-highest method, Wan (Wang et al., 2025), by around 5 times.

4.6 Failure Case and Limitations

Our method warps input frames with generated intermediate optical flows for interpolation, thus requiring the input frames to contain consistent objects and enough overlapping regions. As shown in Fig. 7, with the same starting frame, our method can robustly interpolate with the ending input frame produced by 1 frame and 3 frames skipped. However, when the overlapped regions become minimal, even with a changed main object in the 7-frame-skipped situation, the algorithm struggles to generate coherent and accurate optical flows to move pixels from the inputs to each intermediate frame, resulting in severe degradations through interpolation.

5 Conclusions

In this paper, we propose the first real-time video diffusion-based Video Frame Interpolation (VFI) pipeline that can run 17FPS at 1024 × 576 resolution with even superior interpolation quality than current multi-step solutions. Our work is advancing diffusion-based VFI to more practical and challenging scenarios, such as super slow motion and video compression. This work also hopes to suggest that solving ambiguous components with diffusion models, rather than end-to-end generation, may lead to superior accuracy and efficiency.

Acknowledgements

This work is partially supported by the RGC Early Career Scheme (ECS) No. 24209224. This work was done at Yongrui Ma's internship at the Bytedance Inc.

References

Wenbo Bao, Wei-Sheng Lai, Chao Ma, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. Depth-aware video frame interpolation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 3703–3712, 2019.

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

Karlis Martins Briedis, Abdelaziz Djelouah, Mark Meyer, Ian McGonigal, Markus Gross, and Christopher Schroers. Neural frame interpolation for rendered content. *ACM Trans. Graph.*, 40(6):1–13, 2021.

Hila Chefer, Uriel Singer, Amit Zohar, Yuval Kirstain, Adam Polyak, Yaniv Taigman, Lior Wolf, and Shelly Sheynin. Videojam: Joint appearance-motion representations for enhanced motion generation in video models. *arXiv preprint arXiv:2502.02492*, 2025.

Duolikun Danier, Fan Zhang, and David Bull. Ldmvfi: Video frame interpolation with latent diffusion models. In *AAAI*, volume 38, pp. 1472–1480, 2024.

Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. *Adv. Neural Inform. Process. Syst.*, 34:8780–8794, 2021.

Haiwen Feng, Zheng Ding, Zhihao Xia, Simon Niklaus, Victoria Abrevaya, Michael J Black, and Xuaner Zhang. Explorative inbetweening of time and space. In *Eur. Conf. Comput. Vis.*, pp. 378–395. Springer, 2024.

Yue Gao, Siqi Li, Yipeng Li, Yandong Guo, and Qionghai Dai. Superfast: 200× video frame interpolation via event camera. *IEEE Trans. Pattern Anal. Mach. Intell.*, 45(6):7764–7780, 2022.

Songwei Ge, Aniruddha Mahapatra, Gaurav Parmar, Jun-Yan Zhu, and Jia-Bin Huang. On the content bias in fréchet video distance. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 7277–7288, 2024.

Yuwei Guo, Ceyuan Yang, Anyi Rao, Zhengyang Liang, Yaohui Wang, Yu Qiao, Maneesh Agrawala, Dahua Lin, and Bo Dai. Animatediff: Animate your personalized text-to-image diffusion models without specific tuning. *arXiv preprint arXiv:2307.04725*, 2023.

Zujin Guo, Wei Li, and Chen Change Loy. Generalizable implicit motion modeling for video frame interpolation. *Adv. Neural Inform. Process. Syst.*, 37:63747–63770, 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. *Adv. Neural Inform. Process. Syst.*, 30, 2017.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Adv. Neural Inform. Process. Syst.*, 33:6840–6851, 2020.

Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. In *Int. Conf. Learn. Represent.*, 2023.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *Int. Conf. Learn. Represent.*, 1(2):3, 2022.

Zhewei Huang, Tianyuan Zhang, Wen Heng, Boxin Shi, and Shuchang Zhou. Real-time intermediate flow estimation for video frame interpolation. In *Eur. Conf. Comput. Vis.*, pp. 624–642. Springer, 2022.

Siddhant Jain, Daniel Watson, Eric Tabellion, Ben Poole, Janne Kontkanen, et al. Video interpolation with diffusion models. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 7341–7351, 2024.

Huaizu Jiang, Deqing Sun, Varun Jampani, Ming-Hsuan Yang, Erik Learned-Miller, and Jan Kautz. Super slomo: High quality estimation of multiple intermediate frames for video interpolation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 9000–9008, 2018.

Saurabh Joshi, André Forjaz, Kyu Sang Han, Yu Shen, Vasco Queiroga, Florin A Selaru, Marie Gérard, Daniel Xenes, Jordan Matelsky, Brock Wester, et al. Interpolai: deep learning-based optical flow interpolation and restoration of biomedical images for improved 3d tissue mapping. *Nature Methods*, pp. 1–12, 2025.

Robert Keys. Cubic convolution interpolation for digital image processing. *IEEE transactions on acoustics, speech, and signal processing*, 29(6):1153–1160, 2003.

Anil Kokaram, Davinder Singh, Simon Robinson, Damien Kelly, Bill Collis, and Kim Libreri. Motion-based frame interpolation for film and television effects. *IET Computer Vision*, 14(6): 323–338, 2020.

Jaihyun Lew, Jooyoung Choi, Chaehun Shin, Dahuin Jung, and Sungroh Yoon. Disentangled motion modeling for video frame interpolation. In *AAAI*, volume 39, pp. 4607–4615, 2025.

Zhen Li, Zuo-Liang Zhu, Ling-Hao Han, Qibin Hou, Chun-Le Guo, and Ming-Ming Cheng. Amt: All-pairs multi-field transforms for efficient frame interpolation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 9801–9810, 2023.

Bin Lin, Yunyang Ge, Xinhua Cheng, Zongjian Li, Bin Zhu, Shaodong Wang, Xianyi He, Yang Ye, Shenghai Yuan, Liuhan Chen, et al. Open-sora plan: Open-source large video generation model. *arXiv preprint arXiv:2412.00131*, 2024.

Shanchuan Lin, Xin Xia, Yuxi Ren, Ceyuan Yang, Xuefeng Xiao, and Lu Jiang. Diffusion adversarial post-training for one-step video generation. In *Int. Conf. Learn. Represent.*, 2025.

Xingchao Liu, Chengyue Gong, and Qiang Liu. Flow straight and fast: Learning to generate and transfer data with rectified flow. *arXiv preprint arXiv:2209.03003*, 2022.

Xingchao Liu, Xiwen Zhang, Jianzhu Ma, Jian Peng, et al. Instaflow: One step is enough for high-quality diffusion-based text-to-image generation. In *Int. Conf. Learn. Represent.*, 2023.

Yihao Liu, Liangbin Xie, Li Siyao, Wenxiu Sun, Yu Qiao, and Chao Dong. Enhanced quadratic video interpolation. In *Eur. Conf. Comput. Vis. Workshop*, pp. 41–56. Springer, 2020.

Yongrui Ma, Shi Guo, Yutian Chen, Tianfan Xue, and Jinwei Gu. Timelens-xl: Real-time event-based video frame interpolation with large motion. In *Eur. Conf. Comput. Vis.*, pp. 178–194. Springer, 2024.

Jordi Pont-Tuset, Federico Perazzi, Sergi Caelles, Pablo Arbeláez, Alex Sorkine-Hornung, and Luc Van Gool. The 2017 davis challenge on video object segmentation. *arXiv preprint arXiv:1704.00675*, 2017.

Martin Priessner, David CA Gaboriau, Arlo Sheridan, Tchern Lenn, Carlos Garzon-Coral, Alexander R Dunn, Jonathan R Chubb, Aidan M Tousley, Robbie G Majzner, Uri Manor, et al. Content-aware frame interpolation (cafi): deep learning-based temporal super-resolution for fast bioimaging. *Nature Methods*, 21(2):322–330, 2024.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3, 2022.

Fitsum Reda, Janne Kontkanen, Eric Tabellion, Deqing Sun, Caroline Pantofaru, and Brian Curless. Film: Frame interpolation for large motion. In *Eur. Conf. Comput. Vis.*, pp. 250–266. Springer, 2022.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Adv. Neural Inform. Process. Syst.*, 35:36479–36494, 2022.

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

Hyeonjun Sim, Jihyong Oh, and Munchurl Kim. Xvfi: extreme video frame interpolation. In *Int. Conf. Comput. Vis.*, pp. 14489–14498, 2021.

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

Xiaojuan Wang, Boyang Zhou, Brian Curless, Ira Kemelmacher-Shlizerman, Aleksander Holynski, and Steven M Seitz. Generative inbetweening: Adapting image-to-video models for keyframe interpolation. *Int. Conf. Learn. Represent.*, 2024.

Zhengyi Wang, Cheng Lu, Yikai Wang, Fan Bao, Chongxuan Li, Hang Su, and Jun Zhu. Prolificdreamer: High-fidelity and diverse text-to-3d generation with variational score distillation. *Adv. Neural Inform. Process. Syst.*, 36:8406–8441, 2023.

Xiaoyu Xiang, Yapeng Tian, Yulun Zhang, Yun Fu, Jan P Allebach, and Chenliang Xu. Zooming slow-mo: Fast and accurate one-stage space-time video super-resolution. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 3370–3379, 2020.

Xiangyu Xu, Li Siyao, Wenxiu Sun, Qian Yin, and Ming-Hsuan Yang. Quadratic video interpolation. *Adv. Neural Inform. Process. Syst.*, 32, 2019.

Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *Int. J. Comput. Vis.*, 127(8):1106–1125, 2019.

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. Cogvideox: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024.

Tianwei Yin, Michaël Gharbi, Richard Zhang, Eli Shechtman, Fredo Durand, William T Freeman, and Taesung Park. One-step diffusion with distribution matching distillation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pp. 6613–6623, 2024.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conf. Comput. Vis. Pattern Recog.*, 2018.

Zhixing Zhang, Yanyu Li, Yushu Wu, Anil Kag, Ivan Skorokhodov, Willi Menapace, Aliaksandr Siarohin, Junli Cao, Dimitris Metaxas, Sergey Tulyakov, et al. Sf-v: Single forward video generation model. *Adv. Neural Inform. Process. Syst.*, 37:103599–103618, 2024.

Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *arXiv preprint arXiv:1805.09817*, 2018.

Tianyi Zhu, Dongwei Ren, Qilong Wang, Xiaohe Wu, and Wangmeng Zuo. Generative inbetweening through frame-wise conditions-driven video generation. *arXiv preprint arXiv:2412.11755*, 2024.