PANORAMA-AWARE DIFFUSION TRANSFORMER FOR 360° VIDEO FRAME INTERPOLATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Panoramic Video Frame Interpolation (VFI) poses unique challenges due to the equirectangular projection (ERP), which leads to latitude-dependent spatial distortion and a discontinuous longitude boundary. Furthermore, the presence of large and nonlinear motion in ERP panoramas complicates correspondence estimation and temporally coherent synthesis. In this work, we present PADiT, the first **P**anorama-**A**ware **Di**ffusion **T**ransformer-based framework specifically designed for 360° video frame interpolation. PADiT introduces new DiT blocks that blend **M**otion perception and **T**emporal-attention (MT-DiT), where each transformer block employs multi-head temporal attention to aggregate information from the input ERP frame pair and the latent representation of the intermediate frame. We also integrate flow-derived motion embeddings with diffusion timesteps and frame differences, allowing for the modulation of the denoising blocks through adaptive layer normalization. To enhance geometric and motion modeling for ERP panoramas, we implement spherical position encoding, flow-aware attention, and panorama-aware pyramid feature fusion. These components help mitigate issues associated with ERP, such as positional bias, unreliable motion correspondences, and the misalignment of cross-scale features. Experiments on several widely used panoramic benchmarks, including WEB360, 360VDS, and FlowScape, demonstrate that the proposed PADiT outperforms state-of-the-art methods in panoramic VFI. Specifically, it reduces the FloLPIPS and LPIPS metrics by up to 9.33% and 8.44%, respectively.

1 INTRODUCTION

Video frame interpolation (VFI) aims to synthesize high-fidelity intermediate frames from two or more consecutive input frames(Jin et al., 2023; Jiang et al., 2018; Huang et al., 2022; Bao et al., 2019; Lei et al., 2024; Seo et al., 2025). This process is essential for improving video smoothness and facilitating various downstream applications, such as slow-motion generation, frame-rate conversion, and video enhancement. Recently, diffusion models have extended VFI beyond the traditional flow-guided warping approach by directly modeling conditional frame synthesis. Representative methods in this area include LDMVFI(Danier et al., 2024), CBBD(Lyu et al., 2024), and EDEN(Zhang et al., 2025). These methods typically compress frames into latent representations, incorporate the context of input frames during the diffusion process, and then decode the resulting latent into an intermediate frame(Song et al., 2021; Ho et al., 2020).

However, panoramic videos are typically stored in an equirectangular projection (ERP) format, which has unique characteristics such as latitude-dependent sampling distortion and a circular longitude topology that are not adequately addressed by traditional planar VFI pipelines (Jain et al., 2024). Applying diffusion-based VFI methods to panoramic video presents several challenges. Firstly, standard 2D positional encoding(Peebles & Xie, 2023) does not account for latitude-dependent geometry and the circular nature of longitudes. Moreover, camera rotation and cross-boundary object motion can create significant nonlinear displacements in an equirectangular image, especially in high-latitude regions where the sampling distortion is most severe (Shi et al., 2023). This makes optical flow estimation and cross-frame feature alignment unreliable (Ranjan & Black, 2017; Liu et al., 2022). Lastly, using planar multi-scale fusion may propagate these geometric misalignments across different resolutions.

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.

To address the aforementioned challenges, we present PADiT, a Panorama-Aware Diffusion Transformer designed for 360° VFI. PADiT jointly models ERP geometry and spatial-varying motion deformation by using panorama-aware latent representations, motion-conditioning, and temporal attention aggregation. It incorporates spherical position encoding to provide ERP-derived positional cues, thereby enhancing the modeling of regions near the poles and the left-right boundaries in an ERP image. Additionally, PADiT includes a Motion-conditioning and Temporal attention Diffusion Transformer (MT-DiT) that combines optical flow, interpolation timestamp, and frame-difference conditions. Instead of jointly optimizing the tokenizer and the MT-DiT, we employ an improved two-stage progressive optimization strategy that collaborates with an adapted representation-alignment (REPA) objective after pretraining the tokenizer. This training paradigm helps prevent early competition between the reconstruction and diffusion objectives, allowing for their mutual adaptation.

The contributions of this work can be summarized in four aspects: **(i)** To the best of our knowledge, PADiT is the first diffusion Transformer framework specifically designed for 360° video frame interpolation. Our proposed MT-DiT backbone integrates motion-aware embeddings with multi-head temporal attention to achieve effective temporal information aggregation. **(ii)** We propose an optical flow-aware attention mechanism that incorporates confidence-weighted motion correspondences and explicitly establishes cross-longitude interactions between boundary tokens. **(iii)** We construct a panorama-aware feature representation combining spherical positional encoding and pyramid fusion to mitigate ERP distortion and cross-scale feature misalignment. **(iv)** Experiments on the WEB360, 360VDS, and FlowScape benchmarks demonstrate that our method achieves the lowest LPIPS and FloLPIPS scores compared to other representative and widely-adopted VFI methods, demonstrating significant improvements in perceptual quality.

2 Related Work

2.1 Video Frame Interpolation

Recent VFI methods primarily rely on deep learning, and can be categorized into three main types: kernel-based, flow-guided, and generative-based approaches.

Kernel-based methods. These methods predict local convolution kernels or sampling offsets, or synthesize frames directly from spatio-temporal volumes without explicit motion estimation, to produce intermediate frames(Lee et al., 2020; Niklaus et al., 2017b; Kalluri et al., 2023; Niklaus et al., 2017a; Cheng & Chen, 2020; 2022). They have evolved from using adaptive and separable convolution(Niklaus et al., 2017b; Chollet, 2017; Szegedy et al., 2015) to employing deformable convolution (Dai et al., 2017; Zhu et al., 2019). Representative examples include SepConv (Niklaus et al., 2017b), AdaCoF (Lee et al., 2020), EDSC (Cheng & Chen, 2022), CDFI (Ding et al., 2021), and MSEConv(Ding et al., 2024). Although these approaches are efficient, their limited local receptive fields may hinder performance in scenarios involving significant motion and occlusion.

Flow-guided methods. These methods estimate inter-frame correspondences and synthesize intermediate frames through warping operations (Huang et al., 2022; Seo et al., 2025; Ma et al., 2026; Jin et al., 2023; Liu et al., 2024). Early deep optical-flow estimation was introduced by FlowNet (Dosovitskiy et al., 2015), while estimators such as RAFT provide stronger motion priors for feature alignment (Teed & Deng, 2020). Representative VFI models include AMT (Li et al., 2023), UPR-Net (Jin et al., 2023), MOMO (Lew et al., 2025), SGM-VFI (Liu et al., 2024), VFIMamba (Zhang et al., 2024), and PerVFI (Wu et al., 2024). Bidirectional consistency can suppress unreliable flow estimates (Li et al., 2023), while attention and Transformer mechanisms improve long-range feature matching (Lu et al., 2022; Zhang et al., 2023).

Generative-based methods. These approaches formulate VFI as conditional synthesis and are better suited to complex motion and ambiguous regions (Song et al., 2021; Rombach et al., 2022; Ho et al., 2020). Existing works include GAN-based interpolation (Tran & Yang, 2022), Transformer architectures (Vaswani et al., 2017), diffusion transformers (Peebles & Xie, 2023; Lu et al., 2024), and diffusion-based methods such as LDMVFI(Danier et al., 2024), CBBD(Lyu et al., 2024), EDEN (Zhang et al., 2025), ViBiDSampler (Yang et al., 2025), MPD (Jeon et al., 2026), and ArbInterp(Zhang et al., 2026), all of which target planar video and leave ERP geometry unaddressed.

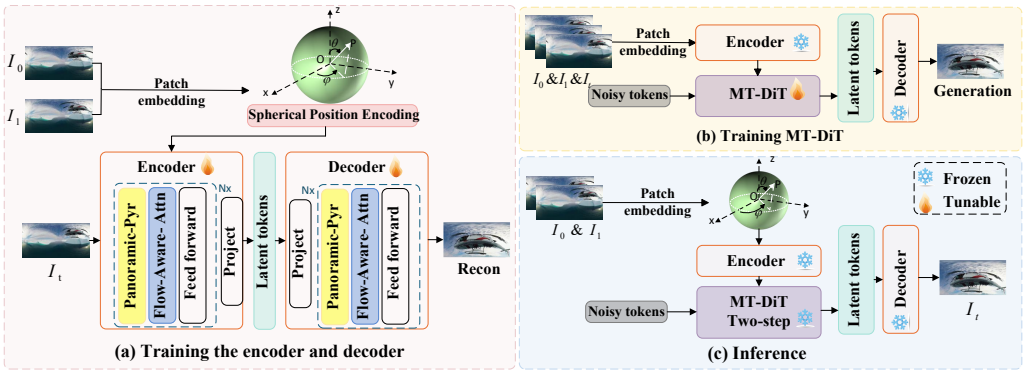

Figure 1: Overview of the PADiT framework. (a) We train a tokenizer through panoramic frame reconstruction to obtain optimal latent tokens; (b) We train the MT-DiT with an improved two-stage progressive optimization strategy; (c) During inference, a two-step denoising process generates intermediate latent tokens, which are then decoded into the interpolated frame.

2.2 Geometric adaptation in panoramic video processing

Panoramic videos are commonly represented in ERP format, whose spatially varying distortion and longitude-boundary continuity pose unique challenges to video processing (Yoon et al., 2022). Existing geometry-adaptive studies mainly address static tasks, such as super-resolution(An et al., 2026), semantic segmentation (Zhang et al., 2022), and denoising (Fermanian et al., 2023; Xu et al., 2022), by incorporating spherical geometric constraints, mapping ERP images to spherical representations (Eder et al., 2020), or designing panorama-specific feature extraction (Su & Grauman, 2017; Tateno et al., 2018). However, the two-dimensional positional encoding used in diffusion transformers cannot represent ERP geometry, and existing panoramic encodings have been studied mainly for static classification and segmentation (Coors et al., 2018). Even VFI-specific positional designs, such as the timestamp-aware RoPE of ArbInterp(Zhang et al., 2026), address only the temporal axis. The DM-360VFI (Li et al., 2025) extends diffusion-based VFI to 360° video, and explicit spherical-coordinate positional encoding and diffusion transformer-based VFI for 360° video remain underexplored. In contrast, we introduce a panoramic VFI based on the DiT.

3 Approach

3.1 Network Architecture

We aim to enhance diffusion-based panoramic (VFI) by utilizing panoramic-aware latent representations, motion-conditioning embeddings, tokenizer pretraining, and an improved two-stage progressive diffusion optimization process. The proposed framework, referred to as **PADiT**, is built on a Motion-conditioning Temporal attention Diffusion Transformer (MT-DiT) backbone and includes three key panorama-aware modules: spherical position encoding, panorama-aware pyramid feature fusion, and flow-aware attention. The overall architecture is shown in Figure 1.

3.1.1 Spherical position encoding

Conventional positional encodings for Transformers(Vaswani et al., 2017) do not explicitly represent the latitude-dependent geometry and circular longitude boundary of ERP video. Inspired by latent representations for ERP images(Coors et al., 2018), we construct positional features from the spherical coordinates associated with each ERP pixel.

Formally, let I_0 and I_1 be the input ERP frame pairs, each with a width of W and a height of H. For any pixel location (u, v) in the ERP frame, we can calculate its corresponding spherical coordinates using the formulas: $\theta = \frac{v}{H} \times \pi$ and $\phi = \frac{u}{W} \times 2\pi - \pi$. Here $\theta \in [0, \pi]$ is the pitch angle (latitude), and $\phi \in [-\pi, \pi]$ is the azimuth angle (longitude). We have converted the coordinate system from the

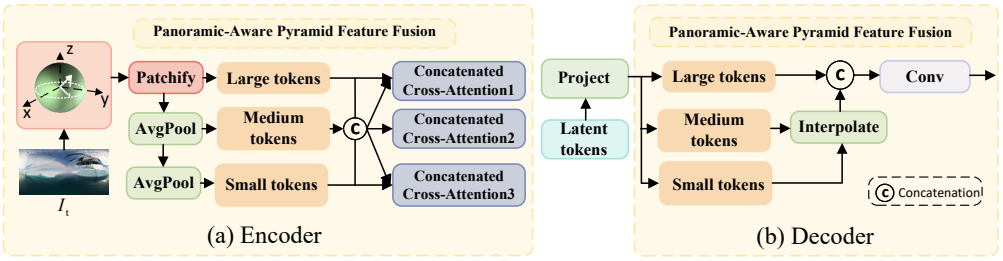

Figure 2: Architectural details of the pyramid feature fusion module. (a) Encoder and (b) Decoder.

traditional two-dimensional grid index to spherical coordinates. While retaining the dual-frequency position encoding mechanism, we constructed high-dimensional position features based on these geometric representations. For instance, regarding θ, the positional encoding formula is as follows:

$$PE(\theta, 2k) = \sin\left(\frac{\theta}{10000^{\frac{2k}{d_{\mathrm{m}}}}}\right), \qquad PE(\theta, 2k+1) = \cos\left(\frac{\theta}{10000^{\frac{2k}{d_{\mathrm{m}}}}}\right) \tag{1}$$

where d_{m} denotes the total dimension of the positional encoding, which is set to 768 to align with the hidden dimension of the DiT backbone, and k is the dimension index. The encodings for latitude θ and longitude ϕ are concatenated along the channel dimension to create $PE(\theta, \phi) \in \mathbb{R}^{d_{\mathrm{m}}}$, which is then added to the corresponding ERP visual feature. This construction injects the ERP-derived angular coordinates into the token representation.

3.1.2 Panorama-Aware Pyramid Feature Fusion

Planar pyramid feature fusion does not explicitly account for latitude-dependent ERP sampling, which can lead to misalignment across different scales. We therefore develop a panorama-aware pyramid feature fusion module that maps multi-scale features to spherical coordinates and uses self-attention mechanisms to enable within- and cross-scale interactions.

As illustrated in Figure 2,our fusion module employs an integrated encoder-decoder structure to extract multi-scale features and facilitate cross-scale interactions. During the encoding stage, the input $H \times W$ ERP image is paired with its corresponding spherical coordinates. A spherical-perception patchification then divides the image along its latitude and longitude, flattening it into a token sequence while preserving adjacency between patches at the left-right boundaries. Through average pooling, we construct a multi-scale token pyramid with lengths of N, $N/4$, and $N/16$. Tokens at each scale are processed by designated transformer blocks. Interactions between adjacent scales are achieved by forming the query from the tokens of the current scale, while the keys and values are derived from the concatenation of tokens from both the current and adjacent scales.

In the decoding stage, we use stepwise upsampling and feature aggregation to reconstruct the fused multi-scale tokens into the original resolutions of the image. Starting from the coarse-resolution latent representation, we apply bilinear interpolation to upsample it to the finer resolution of the adjacent branch. This result is concatenated with the fused token at the current scale and then processed by a lightweight convolution for aggregation and channel adjustment. This process is repeated until we restore the original token length, after which the features are reshaped to match the original ERP size.

3.1.3 Flow-Aware Attention

ERP distortion can degrade optical flow estimation, especially near the poles and at the longitude boundaries. To address this challenge, we introduce a confidence-weighted attention mechanism that integrates motion-aligned input features into the panoramic representation.

As shown in Figure 3(a), we first use RAFT-Lite(Teed & Deng, 2020) to estimate the bidirectional flows $\mathbf{F}_{0\to1}$ and $\mathbf{F}_{1\to0}$ from the two ERP frames I_0 and I_1. For a target timestamp $t \in (0, 1)$, we obtain the intermediate flows using the standard bidirectional flow projection: $\mathbf{F}_{t\to0} = -t(1 -$

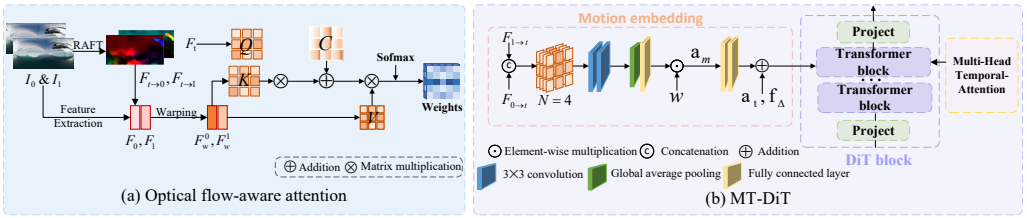

Figure 3: Architectural details of the proposed modules. (a) Optical flow-aware attention and (b) MT-DiT, which incorporates motion-aware embedding and multi-head temporal attention.

$t)\mathbf{F}_{0\to1} + t^2\mathbf{F}_{1\to0}$ and $\mathbf{F}_{t\to1} = (1-t)^2\mathbf{F}_{0\to1} - t(1-t)\mathbf{F}_{1\to0}$. Since these flows are derived solely from I_0, I_1, and t, the projection does not require access to the ground-truth intermediate frame I_t during inference. We then employ bilinear sampling to backward-warp the input feature maps $\mathcal{F}_0$ and $\mathcal{F}_1$ toward the target timestamp, producing $\mathcal{F}_w^0 = \mathcal{W}(\mathcal{F}_0, \mathbf{F}_{t\to0})$ and $\mathcal{F}_w^1 = \mathcal{W}(\mathcal{F}_1, \mathbf{F}_{t\to1})$, where $\mathcal{W}(\cdot,\cdot)$ denotes bilinear warping. To mitigate unreliable correspondences in highly distorted or occluded regions, we compute a smoothness-based flow confidence map:

$$\boldsymbol{C} = \exp\left(-0.5\left\|\nabla\mathbf{F}_{t\to0}\right\|_2 - 0.5\left\|\nabla\mathbf{F}_{t\to1}\right\|_2\right), \tag{2}$$

where ∇ signifies the spatial gradient operator. Regions with smooth flow yield confidence values closer to 1, whereas abrupt flow variations are assigned lower confidence values. At each denoising step, the current target latent features form the query $\boldsymbol{Q}$, while the aligned features $\mathcal{F}_w^0$ and $\mathcal{F}_w^1$, along with the current target-latent features, serve as the keys $\boldsymbol{K}$ and values $\boldsymbol{V}$. We incorporate the confidence map as a logarithmic bias in the attention scores:

$$\mathrm{Attn}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \mathrm{Softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^\top}{\sqrt{d_k}} + \log(\boldsymbol{C})\right)\boldsymbol{V}, \tag{3}$$

where d_k denotes the dimension of the key vectors.

3.1.4 MT-DiT

As shown in Figure 3(b), MT-DiT combines Motion-aware embedding with multi-head Temporal attention. The motion-aware conditioning modulates each DiT block, while multi-head temporal attention aggregates information from the two input frames and the latent representation of the intermediate frame during the denoising process. Existing diffusion models often integrate motion information through feature concatenation(Jain et al., 2024) or global conditioning(Huang et al., 2024). In contrast, we design a motion-aware embedding module that extracts a global motion condition from bidirectional optical flow and combines it with interpolation timestamps and frame-difference information.

We concatenate the bidirectional optical flow fields $\mathbf{F}_{0\to t}$ and $\mathbf{F}_{1\to t}$ into a four-channel input and process it with a two-layer convolutional network using 3×3 kernels and ReLU activations. A learnable channel-weight vector $\boldsymbol{w} \in \mathbb{R}^N$, where $N = 26$ is the number of feature channels, reweights the extracted motion features:

$$\boldsymbol{M} = \boldsymbol{w} \odot \mathrm{Conv2d}\left(\mathrm{Cat}(\mathbf{F}_{0\to t}, \mathbf{F}_{1\to t})\right), \tag{4}$$

where $\odot$ represents element-wise multiplication and $\mathrm{Cat}(\cdot)$ denotes channel concatenation. Next, global average pooling compresses the weighted motion feature map $\boldsymbol{M}$ into a column vector $\mathbf{a}_\mathrm{m} \in \mathbb{R}^N$. A two-layer fully connected network then projects this vector to the 768-dimensional DiT hidden space. The result is added to the interpolation-time embedding $\mathbf{a}_t \in \mathbb{R}^{768}$ and frame-difference embedding $\mathbf{f}_\Delta \in \mathbb{R}^{768}$, and the fused condition is injected into each DiT block through adaptive layer normalization.

3.2 Training Strategy

Our training pipeline includes two phases: tokenizer pretraining and a two-stage progressive optimization. First, we pretrain the tokenizer specifically for panoramic frame reconstruction. After this,

we freeze the pretrained tokenizer and proceed to train the MT-DiT model using panoramic datasets. Finally, we unfreeze the tokenizer and jointly fine-tune it alongside MT-DiT, using the combined supervision that incorporates diffusion, reconstruction, and representation-alignment losses.

Tokenizer Pretraining. The tokenizer, consisting of a transformer-based encoder and decoder, is first pretrained for panoramic frame reconstruction. Given a ground-truth intermediate frame, the encoder transforms it into a one-dimensional latent-token sequence, while the decoder reconstructs the frame from these tokens. For tokenizer pretraining, our training objective combines a pixel-level ℓ_1 Loss (Isola et al., 2017), a perceptual loss (Zhang et al., 2018), a patch-based adversarial loss (Esser et al., 2021), and a KL-divergence loss (Rombach et al., 2022) as follows:

$$\mathcal{L}_{\text{rec}} = \lambda_1 \mathcal{L}_1 + \lambda_p \mathcal{L}_p + \lambda_G \mathcal{L}_G + \lambda_{\text{kl}} \mathcal{L}_{\text{kl}}, \tag{5}$$

where λ_1, λ_p, λ_G, and λ_{kl} are the trade-off parameters for each loss. This pretraining phase establishes a stable latent representation that serves as a foundation for the subsequent diffusion training.

Two-stage Progressive Optimization. After pretraining the tokenizer, we proceed with a two-stage progressive optimization. In Stage 1, all tokenizer parameters are frozen, and only the MT-DiT model is trained on panoramic datasets. For MT-DiT training, we adopt the standard diffusion noise-prediction loss as the training objective(Ho et al., 2020):

$$\mathcal{L}_{\text{diff}} = \mathbb{E}_{I_0, I_1, t, \epsilon} \left[\|\epsilon - \epsilon_\theta(z_t, I_0, I_1, t)\|_2^2 \right], \tag{6}$$

where I_0 and I_1 are the input frame pairs, z_t is the noisy latent representation of the target frame at diffusion timestep t, ϵ is the sampled noise, and ϵ_θ is the noise predicted by MT-DiT. The fixed tokenizer ensures a stable latent space, allowing MT-DiT to adapt to the distribution of training data without being impacted by simultaneous changes in the latent representation.

In Stage 2, we unfreeze the tokenizer and jointly fine-tune both the tokenizer and the MT-DiT model. To enhance the semantic consistency of intermediate MT-DiT representations, we introduce a REPresentation Alignment (REPA) (Yu et al., 2025). Specifically, a pretrained DINOv2 encoder extracts the target representation $\mathbf{y}_t$ from the ground-truth intermediate frame I_t, while a trainable projection layer maps the selected MT-DiT output $\mathbf{h}_t$ to the hidden state $\tilde{\mathbf{h}}_t$. REPA achieves representation alignment through the following optimization objective:

$$\mathcal{L}_{\text{REPA}} = 1 - \text{sim}(\mathbf{y}_t, \tilde{\mathbf{h}}_t), \tag{7}$$

where $\text{sim}(\cdot)$ represents cosine similarity. The total supervision for Stage 2 training is given by:

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{diff}} + \lambda \mathcal{L}_{\text{REPA}} + \eta \mathcal{L}_{\text{reg}}, \tag{8}$$

where λ and η control the contributions of representation alignment and tokenizer reconstruction, respectively. This progressive approach prevents unstable joint optimization from the outset while allowing the tokenizer and MT-DiT to adapt to each other during the final training phase.

4 Experiments

4.1 Experimental Setup

Datasets. We construct the training data from WEB360 (Wang et al., 2024), which contains 2,114 panoramic video-text pairs. We resize the ERP frames to 512×256 and form 33 triplets per video, yielding 69,762 triplets in total. For evaluation, we additionally use 360VDS (Baniya et al., 2024), whose source paper reports 590 videos at 480×360. Because the spherical position encoding and the cross-longitude attention of PADiT assume a complete equirectangular parameterisation—the spherical coordinates must span the full latitude range and the left and right image borders must be physically adjacent—we resample every frame to 480×240 so that the 2:1 aspect ratio of ERP is restored while the full $360^\circ \times 180^\circ$ field of view is preserved and no region of the sphere is cropped. This yields seven triplets per video and 4,130 test triplets. FlowScape (Shi et al., 2023) contains 6,400 panoramic frames with ground-truth optical flow and pixel-level semantic annotations; we likewise resample the frames to 480×240 and construct 700 triplets for evaluation.

Evaluation Metrics. We report PSNR to measure fidelity, LPIPS (Zhang et al., 2018) for perceptual similarity, and FloLPIPS (Danier et al., 2022) for motion-aware interpolation quality. FloLPIPS

Table 1: Quantitative evaluation on three challenging panoramic benchmarks. The best results among all methods are highlighted in red, and the second best in blue. Runtime (RT) is measured on an NVIDIA GeForce RTX 3090 at 480×240 and *includes* optical-flow estimation and tokenizer encoding/decoding, averaged over the test set.

	WEB360				360VDS			FlowScape		
Method	PSNR↑	FloLPIPS↓	LPIPS↓	RT(s)↓	FloLPIPS↓	LPIPS↓	RT(s)↓	FloLPIPS↓	LPIPS↓	RT(s)↓
EDEN (Zhang et al., 2025)	34.406	0.0420	0.0223	1.850	0.0256	0.0151	1.046	0.0415	0.0268	1.617
CBBD (Lyu et al., 2024)	37.042	0.0308	0.0154	1.910	0.0298	0.0148	1.282	0.0405	0.0192	1.806
LDMVFI (Danier et al., 2024)	35.837	0.0449	0.0214	5.607	0.0264	0.0139	4.005	0.0422	0.0239	6.832
MOMO (Lew et al., 2025)	36.809	0.0326	0.0170	0.774	0.0239	0.0129	0.453	0.0268	0.0133	0.669
AMT (Li et al., 2023)	37.310	0.0533	0.0256	0.433	0.0292	0.0165	0.378	0.0824	0.0530	0.405
Ours	37.646	0.0280	0.0141	0.978	0.0236	0.0125	0.475	0.0243	0.0129	0.908

adjusts the perceptual distance based on optical-flow endpoint error along the interpolated trajectory, making it particularly informative in the context of the large, non-linear motions seen in ERP panoramas. Additionally, we provide wall-clock runtime measurements to compare practical efficiency.

Implementation Details. The training process utilized the AdamW optimizer, with parameters set to $\beta_1 = 0.9$, $\beta_2 = 0.99$, and a weight decay of 1×10^{-4}. These settings help reduce model overfitting and enhance the generalization ability for panoramic data. The tokenizer is first pretrained for panoramic frame reconstruction using the loss function described in (5). Following this pretraining, we carry out a two-stage progressive optimization. In Stage 1, the tokenizer remains frozen while only the MT-DiT is trained. During this stage, cosine annealing is employed to decrease the learning rate from 1.0×10^{-4} to 1.0×10^{-8}. In Stage 2, the tokenizer is unfrozen and is jointly fine-tuned with MT-DiT. The learning rate in this stage is decayed from 1.0×10^{-5} to 1.25×10^{-8}. The weights λ_1, λ_p, λ_G, and λ_{kl} in (5) are set to 1.0, 1.0, 0.5, and 1.0×10^{-6}, respectively. Additionally, the weights λ and η in (8) are set to 0.5 and 1.0, respectively. The tokenizer uses eight transformer blocks in the encoder and decoder, while the MT-DiT incorporates 12 blocks. All blocks have a hidden dimension of 768.

4.2 Comparisons with prior works

Baselines. We compare against representative VFI methods, including optical flow-based AMT (Li et al., 2023), MOMO (Lew et al., 2025), and diffusion-based EDEN (Zhang et al., 2025), LDMVFI (Danier et al., 2024), and CBBD (Lyu et al., 2024). For a fair comparison, all competing methods are retrained on the same panoramic training data using their official implementations, with full ERP frames directly used as planar inputs at the same resolution.

Quantitative Results. As shown in Table 1, PADiT demonstrates superior performance across the WEB360, 360VDS, and FlowScape benchmarks. On the WEB360 benchmark, PADiT achieves the best PSNR, LPIPS, and FloLPIPS scores, lowering the LPIPS by 8.4% (0.0154→0.0141) and reducing the FloLPIPS by 9.1% (0.0308→0.0280) compared to the strongest baseline, CBBD. Similarly, On the 360VDS and FlowScape, PADiT records the best LPIPS and FloLPIPS results among all compared methods: LPIPS improves by 3.1% and 3.0%, while FloLPIPS improves by 1.3% and 9.3%, relative to MOMO, the leading baseline for both benchmarks. The consistent gains across all three panoramic benchmarks, particularly in FloLPIPS, demonstrate that MT-DiT more effectively exploits cross-frame dependencies during denoising than generic diffusion-based baselines.

Runtime Comparisons. As reported in Table 1, our PADiT is the fastest diffusion-based interpolation method, running $1.9\times$ and $5.7\times$ faster than EDEN and LDMVFI on the WEB360, respectively. Additionally, It runs $2.2\times$, $2.7\times$, and $8.4\times$ faster than EDEN, CBBD, and LDMVFI on the 360VDS benchmark, while remaining within $1.3\times$ of the fastest method overall, MOMO, and matching its perceptual quality. The slight gap compared to AMT and MOMO is attributed to the additional RAFT-Lite flow estimation and tokenizer encoding and decoding, which can be amortized across consecutive frames.

Qualitative Results. Figure 4 provides a qualitative comparison of PADiT with other VFI methods on the WEB360 benchmark. In the helicopter example, CBBD and LDMVFI produce blurriness around the rotating propeller, whereas our result retains more distinct blade structures. In the snow-



Figure 4: Visual comparison of different methods on the WEB360 benchmark.



Figure 5: Visual comparison of different methods on the 360VDS.

sports example, the displayed AMT and EDEN results contain edge ghosting near the moving arm, while our result shows a clearer contour.

Figure 5 compares PADiT with other VFI techniques on the 360VDS. In the outdoor skiing example, both CBBD and LDMVFI produce blurred ski structures and artifacts near the ski poles. Meanwhile, AMT and EDEN show edge ghosting near the skier and equipment. In contrast, our method maintains sharper ski contours and reduces visible artifacts in these regions. Figure 6 presents the qualitative results on the FlowScape benchmark. Our method retains more of the red car's hood texture and preserves the door-frame structure more closely to the ground truth than the displayed comparison results.

4.3 Ablation Study

To evaluate the effectiveness of the proposed structure, we conducted ablation studies on the WEB360, 360VDS, and FlowScape datasets. Unless stated otherwise, each ablation starts from the full model and removes one component at a time while keeping all other components, the training schedule, and the inference protocol the same. The corresponding cumulative variant in the supple-

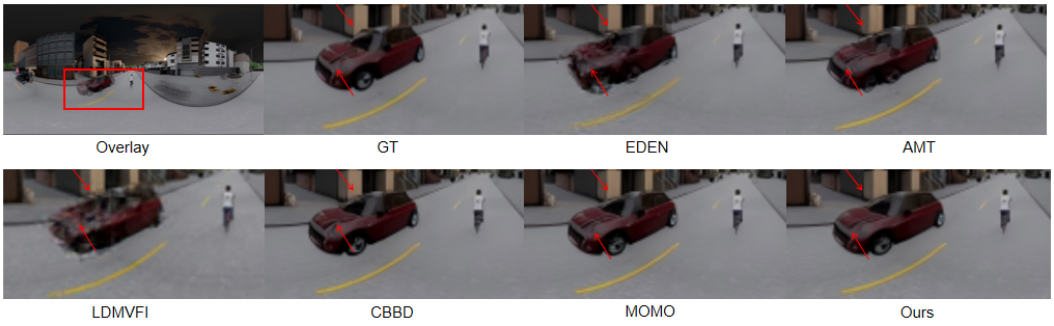

Figure 6: Visual comparison of different methods on the FlowScape benchmark.

Table 2: Ablation study on the SPE.

		WEB360		360VDS		FlowScape	
		LPIPS↓	FloLPIPS↓	LPIPS↓	FloLPIPS↓	LPIPS↓	FloLPIPS↓
SPE	×	0.0158	0.0325	0.0131	0.0246	0.0137	0.0290
SPE	✓	**0.0141**	**0.0280**	**0.0125**	**0.0236**	**0.0129**	**0.0243**

Table 3: Ablation study on the PA-PFF.

		WEB360		360VDS		FlowScape	
		LPIPS↓	FloLPIPS↓	LPIPS↓	FloLPIPS↓	LPIPS↓	FloLPIPS↓
PA-PFF	×	0.0150	0.0298	0.0146	0.0256	0.0192	0.0283
PA-PFF	✓	**0.0141**	**0.0280**	**0.0125**	**0.0236**	**0.0129**	**0.0243**

mentary material (see Section A) differs from these single-component removals and is therefore not directly comparable on a row-by-row basis.

Spherical Position Encoding. We begin by replacing Spherical Position Encoding (SPE) with planar positional encoding while keeping the remaining model components unchanged. As shown in Table 2, SPE reduces LPIPS and FloLPIPS across all three datasets under the specified conditions.

Panorama-Aware Pyramid Feature Fusion. Next, we substitute the Panorama-Aware Pyramid Feature Fusion (PA-PFF) module with a conventional pyramid fusion module. As Table 3 shows, the panorama-aware variant consistently achieves lower LPIPS and FloLPIPS scores on all three datasets.

Table 4: Ablation study on the flow-aware attention.

		WEB360		360VDS		FlowScape	
		LPIPS↓	FloLPIPS↓	LPIPS↓	FloLPIPS↓	LPIPS↓	FloLPIPS↓
FAA	×	0.0172	0.0328	0.0154	0.0240	0.0145	0.0283
FAA	✓	**0.0141**	**0.0280**	**0.0125**	**0.0236**	**0.0129**	**0.0243**

Table 5: Ablation study on the motion-aware embedding.

		WEB360		360VDS		FlowScape	
		LPIPS↓	FloLPIPS↓	LPIPS↓	FloLPIPS↓	LPIPS↓	FloLPIPS↓
MEM	×	0.0167	0.0348	0.0152	0.0295	0.0135	0.0259
MEM	✓	**0.0141**	**0.0280**	**0.0125**	**0.0236**	**0.0129**	**0.0243**

Optical Flow-Aware Attention. We then replace optical Flow-Aware Attention (FAA) with temporal attention while keeping the remaining model components fixed. As shown in Table 4, the flow-aware variant obtains lower LPIPS and FloLPIPS scores on all three datasets.

Motion-Aware Embedding. Finally, we remove the motion-aware embedding (MEM) from the final model. As Table 5 indicates, this removal leads to an increase in LPIPS and FloLPIPS scores across all three datasets, demonstrating the effectiveness of the MEM.

5 Conclusion

We propose PADiT, the first panorama-aware diffusion transformer specifically designed for 360° video frame interpolation. Our model incorporates several advanced features: spherical positional encoding, panorama-aware pyramid feature fusion, optical flow-aware attention, and motion conditioning. Moreover, we introduce an improved two-stage progressive optimization training strategy to address instability arising from jointly training the tokenizer and the MT-DiT. This strategy reduces interference among competing training objectives, prevents drift in the latent space, and allows both the tokenizer and MT-DiT to evolve synergistically. Experiments over multiple widely adopted 360° video datasets show that PADiTachieves lower LPIPS and FloLPIPS scores than current state-of-the-art methods and produces superior synthesized frames.

AI USE STATEMENT

Generative AI tools were used to assist with language editing. All AI-assisted changes were reviewed by the authors, who take responsibility for the final text, claims, and presentation.

REPRODUCIBILITY STATEMENT

The model architecture and objectives are described in Section 3, and the datasets, implementation settings, evaluation protocol, and component ablations are reported in Section 4. Additional implementation material can be provided as anonymous supplementary material for review.

REFERENCES

Hongyu An, Xinfeng Zhang, Shijie Zhao, Li Zhang, and Ruiqin Xiong. Spatio-temporal distortion aware omnidirectional video super-resolution. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(4):2309–2317, 2026.

Arbind Agrahari Baniya, Tsz-Kwan Lee, Peter W. Eklund, and Sunil Aryal. Omnidirectional video super-resolution using deep learning. *IEEE Transactions on Multimedia*, 26:540–554, 2024.

Wenbo Bao, Wei-Sheng Lai, Chao Ma, et al. Depth-aware video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3703–3712, 2019.

Xianhang Cheng and Zhenzhong Chen. Video frame interpolation via deformable separable convolution. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 10607–10614, 2020.

Xianhang Cheng and Zhenzhong Chen. Multiple video frame interpolation via enhanced deformable separable convolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10): 7029–7045, 2022.

François Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1800–1807, 2017.

Benjamin Coors, Alexandru Paul Condurache, and Andreas Geiger. SphereNet: Learning spherical representations for detection and classification in omnidirectional images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 525–541, 2018.

Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 764–773, 2017.

Duolikun Danier, Fan Zhang, and David Bull. FloLPIPS: A bespoke video quality metric for frame interpolation. In *Picture Coding Symposium (PCS)*, pp. 283–287, 2022.

Duolikun Danier, Fan Zhang, and David Bull. LDMVFI: Video frame interpolation with latent diffusion models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pp. 1472–1480, 2024.

Tianyu Ding, Luming Liang, Zhihui Zhu, et al. CDFI: Compression-driven network design for frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8001–8011, 2021.

Xiangling Ding, Pu Huang, Dengyong Zhang, et al. MSEConv: A unified warping framework for video frame interpolation. *ACM Transactions on Asian and Low-Resource Language Information Processing*, 2024.

Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. FlowNet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 2758–2766, 2015.

Marc Eder, Mykhailo Shvets, John Lim, and Jan-Michael Frahm. Tangent images for mitigating spherical distortion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12426–12434, 2020.

Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12873–12883, 2021.

Rita Fermanian, Thomas Maugey, and Christine Guillemot. SphereDRUNet: A spherical denoiser for omnidirectional images. In *Proceedings of the IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct)*, pp. 417–422, 2023.

Jonathan Ho, Ajay N. Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 33, pp. 6840–6851, 2020.

Zhewei Huang, Tianyuan Zhang, Wen Heng, et al. Real-time intermediate flow estimation for video frame interpolation. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 624–642, 2022.

Zhilin Huang, Yijie Yu, Ling Yang, Chujun Qin, Bing Zheng, Xiawu Zheng, Zikun Zhou, Yaowei Wang, and Wenming Yang. Motion-aware latent diffusion models for video frame interpolation. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 1043–1052, 2024.

Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1125–1134, 2017.

Siddhant Jain, Daniel Watson, Eric Tabellion, et al. Video interpolation with diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7341–7351, 2024.

Wooseok Jeon, Seunghyun Shin, Dongmin Shin, and Hae-Gon Jeon. Motion prior distillation in time reversal sampling for generative inbetweening. In *International Conference on Learning Representations (ICLR)*, 2026.

Huaizu Jiang, Deqing Sun, Varun Jampani, et al. Super SloMo: High quality estimation of multiple intermediate frames for video interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9000–9008, 2018.

Xin Jin, Longhai Wu, Jie Chen, Youxin Chen, Jayoon Koo, and Cheul-hee Hahm. A unified pyramid recurrent network for video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1578–1587, 2023.

Tarun Kalluri, Deepak Pathak, Manmohan Chandraker, et al. FLAVR: Flow-agnostic video representations for fast frame interpolation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pp. 2071–2082, 2023.

Hyeongmin Lee, Taeoh Kim, Tae-young Chung, et al. AdaCoF: Adaptive collaboration of flows for video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5315–5324, 2020.

Pengcheng Lei, Faming Fang, Tieyong Zeng, and Guixu Zhang. Flow guidance deformable compensation network for video frame interpolation. *IEEE Transactions on Multimedia*, 26:1801–1812, 2024.

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

Xing Li, Yang Zhou, and Haibing Yin. DM-360VFI: 360° spherical video interpolation using diffusion model. In *2025 International Symposium on Machine Learning and Media Computing (MLMC)*. IEEE, 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9801–9810, 2023.

Chengxu Liu, Huan Yang, Jianlong Fu, and Xueming Qian. Learning trajectory-aware transformer for video super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5687–5696, 2022.

Chunxu Liu, Guozhen Zhang, Rui Zhao, and Limin Wang. Sparse global matching for video frame interpolation with large motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19125–19134, 2024.

Haoyu Lu, Guoxing Yang, Nanyi Fei, Yuqi Huo, Zhiwu Lu, Ping Luo, and Mingyu Ding. VDT: General-purpose video diffusion transformers via mask modeling. In *International Conference on Learning Representations (ICLR)*, 2024.

Liying Lu, Ruizheng Wu, Huaijia Lin, et al. Video frame interpolation with transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 3532–3542, 2022.

Zonglin Lyu, Ming Li, Jianbo Jiao, and Chen Chen. Frame interpolation with consecutive brownian bridge diffusion. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pp. 3449–3458, 2024.

Yongrui Ma, Shijie Zhao, Mingde Yao, Junlin Li, Li Zhang, Xiaohong Liu, Qi Dou, Jinwei Gu, and Tianfan Xue. Realtime video frame interpolation using one-step diffusion sampling. In *International Conference on Learning Representations*, 2026.

Simon Niklaus, Long Mai, and Feng Liu. Video frame interpolation via adaptive convolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2270–2279, 2017a.

Simon Niklaus, Long Mai, and Feng Liu. Video frame interpolation via adaptive separable convolution. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 261–270, 2017b.

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 4195–4205, 2023.

Anurag Ranjan and Michael J. Black. Optical flow estimation using a spatial pyramid network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4161–4170, 2017.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10684–10695, 2022.

Wonyong Seo, Jihyong Oh, and Munchurl Kim. BiM-VFI: Bidirectional motion field-guided frame interpolation for video with non-uniform motions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 7244–7253, 2025.

Hao Shi, Yifan Zhou, Kailun Yang, Xiaoting Yin, Ze Wang, Yaozu Ye, Zhe Yin, Shi Meng, Peng Li, and Kaiwei Wang. PanoFlow: Learning 360° optical flow for surrounding temporal understanding. *IEEE Transactions on Intelligent Transportation Systems*, 24(5):5570–5585, 2023.

Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations (ICLR)*, 2021.

Yu-Chuan Su and Kristen Grauman. Learning spherical convolution for fast features from 360° imagery. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pp. 529–537, 2017.

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–9, 2015.

Keisuke Tateno, Nassir Navab, and Federico Tombari. Distortion-aware convolutional filters for dense prediction in panoramic images. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 732–750, 2018.

Zachary Teed and Jia Deng. RAFT: Recurrent all-pairs field transforms for optical flow. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 402–419, 2020.

Quang Nhat Tran and Shih-Hsuan Yang. Video frame interpolation via down-up scale generative adversarial networks. *Computer Vision and Image Understanding*, 220:103434, 2022.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 30, pp. 5998–6008, 2017.

Qian Wang, Weiqi Li, Chong Mou, Xinhua Cheng, and Jian Zhang. 360DVD: Controllable panorama video generation with 360-degree video diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6913–6923, 2024.

Guangyang Wu, Xin Tao, Changlin Li, Wenyi Wang, Xiaohong Liu, and Qingqing Zheng. Perception-oriented video frame interpolation via asymmetric blending. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2753–2762, 2024.

Yanyu Xu, Ziheng Zhang, and Shenghua Gao. Spherical DNNs and their applications in 360° images and videos. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(10): 7235–7252, 2022.

Serin Yang, Taesung Kwon, and Jong Chul Ye. ViBiDSampler: Enhancing video interpolation using bidirectional diffusion sampler. In *International Conference on Learning Representations (ICLR)*, 2025.

Youngho Yoon, Inchul Chung, Lin Wang, and Kuk-Jin Yoon. SphereSR: 360° image super-resolution with arbitrary projection via continuous spherical image representation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5667–5676, 2022.

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. In *International Conference on Learning Representations (ICLR)*, 2025.

Guozhen Zhang, Yijing Zhu, Haonan Wang, et al. Extracting motion and appearance via inter-frame attention for efficient video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 5682–5692, 2023.

Guozhen Zhang, Chunxu Liu, Yutao Cui, Xiaotong Zhao, Kai Ma, and Limin Wang. VFIMamba: Video frame interpolation with state space models. In *Advances in Neural Information Processing Systems (NeurIPS)*, volume 37, pp. 107225–107248, 2024.

Guozhen Zhang, Haiguang Wang, Chunyu Wang, Yuan Zhou, Qinglin Lu, and Limin Wang. Arbitrary generative video interpolation. In *International Conference on Learning Representations (ICLR)*, 2026.

Jiaming Zhang, Kailun Yang, Chaoxiang Ma, Simon Reiß, Kunyu Peng, and Rainer Stiefelhagen. Bending reality: Distortion-aware transformers for adapting to panoramic semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 16917–16927, 2022.

Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 586–595, 2018.

Zihao Zhang, Haoran Chen, Haoyu Zhao, Guansong Lu, Yanwei Fu, Hang Xu, and Zuxuan Wu. EDEN: Enhanced diffusion for high-quality large-motion video frame interpolation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 2105–2115, 2025.

Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable ConvNets v2: More deformable, better results. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9308–9316, 2019.

A Additional Ablation Results

Table 6 reports cumulative component ablations from the baseline to the full model. Panorama-aware pyramid fusion provides the largest single-step gain on WEB360 and FlowScape, while spherical-coordinate position encoding and flow-aware attention together account for most of the improvement on 360VDS.

Table 6: Cumulative component ablations from the baseline to the full model.

SC-PE	Flow Attention	Panorama Pyramid	Motion Embedding	WEB360		360VDS		FlowScape	
				LPIPS↓	FloLPIPS↓	LPIPS↓	FloLPIPS↓	LPIPS↓	FloLPIPS↓
×	×	×	×	0.0381	0.0629	0.0347	0.0620	0.0448	0.0689
✓	×	×	×	0.0304	0.0521	0.0279	0.0437	0.0335	0.0581
✓	✓	×	×	0.0253	0.0431	0.0182	0.0366	0.0261	0.0459
✓	✓	✓	×	0.0167	0.0348	0.0152	0.0295	0.0135	0.0259
✓	✓	✓	✓	0.0141	0.0280	0.0125	0.0236	0.0129	0.0243