# LDMVFI: Video Frame Interpolation with Latent Diffusion Models

**Duolikun Danier, Fan Zhang, David Bull**

University of Bristol
{duolikun.danier, fan.zhang, dave.bull}@bristol.ac.uk

**Abstract**

Existing works on video frame interpolation (VFI) mostly employ deep neural networks that are trained by minimizing the L1, L2, or deep feature space distance (e.g. VGG loss) between their outputs and ground-truth frames. However, recent works have shown that these metrics are poor indicators of perceptual VFI quality. Towards developing perceptually-oriented VFI methods, in this work we propose latent diffusion model-based VFI, LDMVFI. This approaches the VFI problem from a generative perspective by formulating it as a conditional generation problem. As the first effort to address VFI using latent diffusion models, we rigorously benchmark our method on common test sets used in the existing VFI literature. Our quantitative experiments and user study indicate that LDMVFI is able to interpolate video content with favorable perceptual quality compared to the state of the art, even in the high-resolution regime. Our code is available at https://github.com/danier97/LDMVFI.

## 1 Introduction

Video frame interpolation (VFI) aims to generate intermediate frames between two existing consecutive video frames. It is commonly used to synthetically increase frame rate, e.g., to produce jitter-free slow-motion content (Jiang et al. 2018). VFI has also been used in video compression (Wu, Singhal, and Krahenbuhl 2018), view synthesis (Flynn et al. 2016), medical imaging (Karargyris and Bourbakis 2010) and animation production (Siyao et al. 2021).

Existing VFI methods (Jiang et al. 2018; Xue et al. 2019; Lee et al. 2020; Niklaus and Liu 2020; Kalluri et al. 2023) are mostly based on deep neural networks. These deep models differ in architectural designs and motion modeling approaches, but are mostly trained to minimize the L1, L2, or VGG (Simonyan and Zisserman 2015) feature distance between their outputs and the ground-truth intermediate frames. However, recent works (Men et al. 2020; Danier, Zhang, and Bull 2022d) have shown that these optimization objectives are not indicative of the perceptual quality of interpolated videos, as they correlate poorly with human judgments. As a result, it has been reported (Danier, Zhang, and Bull 2022d) that existing methods, while achieving high PSNR values, tend to under-perform perceptually, especially under challenging scenarios involving dynamic textures with complex motion.

Copyright © 2024, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

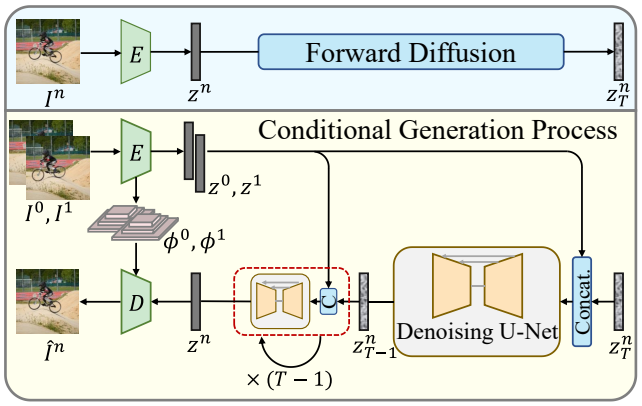


Figure 1: Overview of the diffusion processes in LDMVFI. The encoder and decoder enable projection between image and latent spaces, and the diffusion processes take place in the latent space.

A potential approach to improve the perceptual performance of VFI methods is to develop more accurate perceptual metrics for training VFI models, and this is the focus of another research area, namely Video Quality Assessment (Saha et al. 2023). In this work, instead of relying on particular metrics, we explore a new direction for perception-oriented VFI based on diffusion models (Ho, Jain, and Abbeel 2020; Rombach et al. 2022). Diffusion models have recently exhibited remarkable performance in generating realistic, perceptually-optimized images and videos, reportedly outperforming other generative models including GANs and VAEs (Dhariwal and Nichol 2021; Ho et al. 2022). However, despite their ability to synthesize high-fidelity visual content, the application of diffusion models for VFI has not been fully investigated.

In the above context, we propose a latent diffusion model for video frame interpolation (LDMVFI), where VFI is formulated as a conditional image generation problem. Specifically, we adopt the recently proposed *latent diffusion models* (Rombach et al. 2022) (LDMs) within a framework comprising an autoencoding model that projects images into

a latent space, and a denoising U-Net which performs reverse diffusion process in that latent space. To better adapt LDMs to VFI, we devise VFI-specific components, notably a novel vector quantization-based VFI-autoencoding model, VQ-FIGAN, with which our method shows superior performance over vanilla LDMs.

Despite the paradigmatic shift from the mainstream VFI methods, we adhere to the commonly adopted VFI benchmarking protocol and evaluate the proposed method on various VFI test sets, covering both low and high resolution content (up to 4K). Our results demonstrate that LDMVFI performs favorably against the state of the art in terms of three perceptual metrics (Zhang et al. 2018; Danier, Zhang, and Bull 2022b; Heusel et al. 2017). We also conducted a user study to collect subjective judgments on the quality of full HD videos interpolated by our method benchmarked against several competitive counterparts; this further confirms the favorable perceptual performance of LDMVFI.

To the best of our knowledge, this work is the first to address VFI as a conditional generation problem using latent diffusion models, and the first to demonstrate the potential of the new paradigm for perception-oriented VFI. Our contributions are summarized below.

- We present LDMVFI, a latent diffusion-based method that leverages the high-fidelity image synthesis ability of diffusion models to perform VFI.
- We introduce novel VFI-specific components into LDMs, including a vector-quantized autoencoding model, VQ-FIGAN, which further enhances VFI performance.
- We demonstrate, through quantitative and qualitative experiments, that the proposed method performs favorably against the state of the art.

## 2 Related Work

**Video Frame Interpolation.** Existing VFI approaches are mostly based on deep learning, and can be generally categorized as flow-based or kernel-based. Flow-based methods rely on optical flow estimation to generate interpolated frames. To obtain the optical flow from input frames to the non-existent middle frame (or the other way around), some methods (Jiang et al. 2018; Niklaus and Liu 2018, 2020; Sim, Oh, and Kim 2021) assume certain motion types to infer the intermediate optical flows using the flows between two input frames, while others (Liu et al. 2017; Xue et al. 2019; Park et al. 2020; Park, Lee, and Kim 2021; Lu et al. 2022; Kong et al. 2022) directly estimate the intermediate flows. On the other hand, kernel-based methods argue that optical flows can be unreliable in dynamic texture scenes, so they predict locally adaptive convolution kernels to synthesize output pixels, allowing more flexible many-to-one mapping. While earlier methods (Niklaus, Mai, and Liu 2017) in this class predict fixed-size kernels, more recent ones (Lee et al. 2020; Ding et al. 2021; Cheng and Chen 2020, 2022; Danier, Zhang, and Bull 2022a) tend to adopt deformable convolution (Dai et al. 2017) kernels. Other than these two classes, there are also attempts to combine flows and kernels (Bao et al. 2021; Danier, Zhang, and Bull 2022c), and to perform end-to-end frame synthesis (Choi et al. 2020; Kalluri et al. 2023).

It is noted that the above methods are trained by optimizing PSNR-oriented loss functions, i.e., the L1/L2 distance between the model outputs and the ground-truth frames. To improve perceptual performance, some existing methods (Niklaus and Liu 2018, 2020) use the VGG (Simonyan and Zisserman 2015) feature-based loss in combination with the L1 loss. However, it has been previously reported (Danier, Zhang, and Bull 2022d) that these distances do not fully reflect the perceptual quality of interpolated videos, exhibiting poor correlation performance with subjective ground truth. As a result, it has been observed (Danier, Zhang, and Bull 2022d) that some state-of-the-art methods show unsatisfactory perceptual performance, especially on dynamic textures with complex motions.

**Diffusion Models.** Recently, diffusion models (DMs) (Ho, Jain, and Abbeel 2020; Rombach et al. 2022) have demonstrated remarkable performance in synthesizing high-fidelity images and videos. In their original form (Ho, Jain, and Abbeel 2020), DMs generate new images by progressively denoising a Gaussian noise image; the process corresponds to the reverse of a Markov chain that gradually adds noise to a clean image. DMs have been reported to offer superior performance (Dhariwal and Nichol 2021; Ho et al. 2022) compared to GANs (Goodfellow et al. 2020) and VAEs (Kingma and Welling 2014) in image generation tasks. The only previous application of DMs on VFI is (Voleti, Jolicoeur-Martineau, and Pal 2022), but this work focused on low-resolution images and the model lacked any VFI-specific innovations, showing limited interpolation performance. The recently proposed latent diffusion models (LDMs) have demonstrated strong ability to synthesize high-resolution images by performing diffusion processes in latent space. However, we observe that LDMs have not previously been exploited for VFI.

## 3 Proposed Method: LDMVFI

Given two consecutive frames $I^0, I^1$ from a video, VFI aims to generate the non-existent intermediate frame $I^n$ where $n = 0.5$ for $\times 2$ upsampling. Approaching VFI from a generative perspective, our goal is to learn a parametric approximation of the conditional distribution $p(I^n|I^0, I^1)$ using a dataset $\mathcal{D} = \{I_s^0, I_s^n, I_s^1\}_{s=1}^S$. To achieve this, we adopt the latent diffusion models (Rombach et al. 2022) to perform conditional generation for VFI. The proposed LDMVFI contains two main components: (i) a VFI-specific **autoencoding model**, VQ-FIGAN, that projects frames into a latent space, and reconstructs the target frame from the latent encoding; (ii) a **denoising U-Net** that performs reverse diffusion process in the latent space for conditional image generation. Figure 1 shows the overview of the LDMVFI.

### 3.1 Latent Diffusion Models

Latent diffusion models (LDMs) (Rombach et al. 2022) are built upon the denoising diffusion probabilistic models (Ho, Jain, and Abbeel 2020) (referred to as diffusion models hereafter), which are a class of generative models that learns a

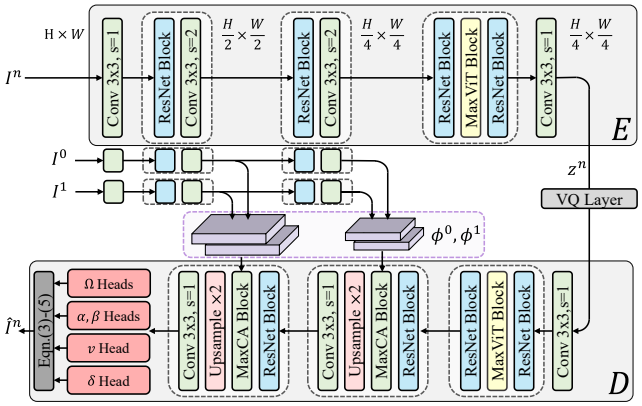


Figure 2: The architecture of the VFI autoencoding model, VQ-FIGAN. It differs from the original VQGAN (Esser, Rombach, and Ommer 2021) in three aspects: (i) the use of features extracted by the encoder from neighboring frames during the decoding via MaxViT-based cross attention; (ii) use of more efficient MaxViT block instead of the vanilla self-attention; (iii) frame synthesis via adaptive deformable convolution. The kernel ($\Omega$), offset ($\alpha, \beta$), visibility ($v$) and residual ($\delta$) heads contain $3\times\{\texttt{conv3x3, ReLU}\}$.

data distribution $p(x)$ by learning the reverse process of a pre-defined Markov Chain of length $T$ (i.e. the forward process) that gradually adds Gaussian noise to the data. Specifically, a time-conditioned neural network $\epsilon_\theta(x_t, t)$ is trained to denoise the data at step $t = 1, \ldots, T$ with the objective

$$\mathcal{L}_{\text{DM}} = \mathbb{E}_{x_0, \epsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I}), t\sim\mathcal{U}(1,T)}\left[\, \|\epsilon - \epsilon_\theta(x_t, t)\|^2 \,\right], \quad (1)$$

where $x_t$ is sampled from the forward diffusion process and $\mathcal{U}(1, T)$ denotes uniform distribution over $\{1, \ldots, T\}$. This corresponds to a reweighted version of the variational lower bound on $\log p(x)$. More details of diffusion processes and the loss are provided in the *Supplementary*.

To sample new images, one can start from a Gaussian noise and gradually denoise it using the denoising network $\epsilon_\theta$. Under the conditional generation setting, we can condition $\epsilon_\theta$ on an additional input $y$, which can denote the two input frames in the context of VFI.

Latent diffusion models contain an image encoder $E : x \mapsto z$ that encodes an image $x$ into a lower-dimensional latent representation $z$, and a decoder $D : z \mapsto x$ that reconstructs the image $x$. The forward and reverse diffusion processes then happen in the latent space, and the training objective for learning the reverse diffusion process becomes

$$\mathcal{L}_{\text{LDM}} = \mathbb{E}_{E(x_0), \epsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I}), t\sim\mathcal{U}(1,T)}\left[\, \|\epsilon - \epsilon_\theta(z_t, t)\|^2 \,\right]. \quad (2)$$

By projecting images into a compact latent space, LDMs allow the diffusion process to concentrate on the semantically important portions of the data and enable more computationally efficient sampling process.

### 3.2 Autoencoding with VQ-FIGAN

In the original form of LDMs, the autoencoding model $\{E, D\}$ is considered as a perceptual image codec. Its design purpose is to project images into efficient latent representations where high-frequency details are removed during encoding and recovered in the decoding. However, in the context of VFI, such information is likely to affect the perceived quality of the interpolated videos, so the limited reconstruction ability of the decoder can negatively impact the VFI performance. To enhance high-frequency detail recovering, we propose a VFI-specific autoencoding model: VQ-FIGAN, which is illustrated in Figure 2. While the backbone model is similar to the original VQGAN (Esser, Rombach, and Ommer 2021) used in (Rombach et al. 2022), there are three major differences.

Firstly, we take advantage of a property of the VFI task - that the neighboring frames are available during inference, in order to design a frame-aided decoder. Specifically, given the ground-truth target frame $I^n \in \mathbb{R}^{H\times W\times 3}$, the encoder $E$ produces the latent encoding $z^n = E(I^n)$, where $z^n \in \mathbb{R}^{\frac{H}{f}\times\frac{W}{f}\times 3}$, and $f$ is a hyper-parameter (Figure 2 shows the case for $f = 4$). Then, the decoder $D$ outputs a reconstructed frame, $\hat{I}^n$, taking input $z^n$, as well as the feature pyramids, $\phi^0, \phi^1$, extracted by $E$ from two neighboring frames, $I^0, I^1$. During decoding, these feature pyramids are fused with the decoded features from $z^n$ at multiple layers using the MaxCA blocks, which are newly designed MaxViT (Tu et al. 2022)-based Cross Attention blocks, where the query embeddings for the attention mechanism are generated using decoded features from $z^n$, and the key and value embeddings are obtained from $\phi^0, \phi^1$.

Secondly, we notice the original VQGAN uses the full self-attention (Vaswani et al. 2017) as in the vision transformer (Dosovitskiy et al. 2021), which can be computationally heavy (quadratic complexity) and memory-intensive, especially when the image resolution is high. For more efficient inference on high-resolution (e.g. full HD) videos, we employ the recently proposed MaxViT block (Tu et al. 2022) to perform self-attention. The multi-axis self-attention layer in the MaxViT block combines windowed attention and dilated grid attention to perform both local and global operations, while achieving linear complexity with respect to the input size.

Thirdly, instead of having the decoder directly output the reconstructed image $\hat{I}^n$, the proposed VQ-FIGAN outputs the deformable convolution-based interpolation kernels (Dai et al. 2017; Lee et al. 2020; Cheng and Chen 2020) to enhance VFI performance. Specifically, given that $H, W$ are the frame height and width, the output of the decoder network contains parameters of the locally adaptive deformable convolution kernels of size $K \times K$: $\{\Omega^\tau, \alpha^\tau, \beta^\tau\}$ where $\tau = 0, 1$ indexes the input frames. Here $\Omega \in [0, 1]^{H\times W\times K\times K}$ contains the weights of the kernels, and $\alpha, \beta \in \mathbb{R}^{H\times W\times K\times K}$ are their spatial offsets (horizontal and vertical respectively). The decoder also outputs a visibility map $v \in [0, 1]^{H\times W}$ to account for occlusion (Jiang et al. 2018), and a residual map $\delta \in \mathbb{R}^{H\times W}$ to further enhance VFI performance (Cheng and Chen 2022). To generate the interpolated frame, firstly locally adaptive deformable con-

volutions are performed for each input frame $\{I^\tau\}_{\tau=0,1}$:

$$I^{n\tau}(h,w) = \sum_{i=1}^{K}\sum_{j=1}^{K} \Omega^\tau_{h,w}(i,j) \cdot P^\tau_{h,w}(i,j), \quad (3)$$

$$P^\tau_{h,w}(i,j) = I^\tau(h + \alpha^\tau_{h,w}(i,j), w + \beta^\tau_{h,w}(i,j)), \quad (4)$$

in which $I^{n\tau}$ denotes the result obtained from $I^\tau$, and $P^\tau_{h,w}$ is the patch sampled from $I^\tau$ for output location $(h,w)$. These intermediate results are then combined using the visibility and residual maps:

$$\hat{I}^n = v \cdot I^{n0} + (1-v) \cdot I^{n1} + \delta. \quad (5)$$

We adopt the separable deformable convolution implementation in (Cheng and Chen 2020) which exploits separability properties of kernels (Rigamonti et al. 2013) to reduce memory requirements while maintaining VFI performance.

**Training VQ-FIGAN.** We follow the original training settings of VQGAN in (Esser, Rombach, and Ommer 2021; Rombach et al. 2022), where the loss function consists of an LPIPS-based (Zhang et al. 2018) perceptual loss, a patch-based adversarial loss (Isola et al. 2017) and a latent regularization term based on a vector quantization (VQ) layer (Van Den Oord, Vinyals et al. 2017). We refer the readers to (Esser, Rombach, and Ommer 2021) for the details.

### 3.3 Conditional Generation with LDM

The trained VQ-FIGAN allows us to access a compact latent space in which we perform forward diffusion by gradually adding Gaussian noise to the latent $z^n$ of the target frame $I^n$ according to a pre-defined noise schedule, and learn the reverse (denoising) process to perform conditional generation. To this end, we adopt the noise-prediction parameterization (Ho, Jain, and Abbeel 2020) of DMs and train a denoising U-Net by minimizing the re-weighted variational lower bound on the conditional log-likelihood $\log p_\theta(z^n|z^0,z^1)$ where $z^0, z^1$ are the latent encodings of the two input frames.

**Training.** Specifically, the denoising U-Net $\epsilon_\theta$ takes as input the "noisy" latent encoding $z^n_t$ for the target frame $I^n$ (sampled from the $t$-th step in the forward diffusion process of length $T$), the diffusion step $t$, as well as the conditioning latents $z^0, z^1$ for the input frames $I^0, I^1$. It is trained to predict the noise added to $z^n$ at each time step $t$ by minimizing

$$\mathcal{L} = \mathbb{E}_{z^n,z^0,z^1,\epsilon\sim\mathcal{N}(\mathbf{0},\mathbf{I}),t}\left[\, \left\|\epsilon - \epsilon_\theta(z^n_t, t, z^0, z^1)\right\|^2 \right], \quad (6)$$

where $t \sim \mathcal{U}(1,T)$. The derivation and full details of the training procedure of $\epsilon_\theta$ are provide in the *Supplementary*. Intuitively, the training is performed by alternately adding a random Gaussian noise to $z^n$ according to a pre-defined noising schedule, and having the network $\epsilon_\theta$ predict the noise added given the step $t$, conditioning on $z^0, z^1$.

**Inference.** To interpolate $\hat{I}^n$ from $I^0, I^1$, we start by sampling a Gaussian noise $z^n_T$ in the latent space, and perform $T$ steps of denoising until we obtain $z^n_0$. Within each step, firstly the network $\epsilon_\theta$ predicts the noise $\hat{\epsilon}$. Then $z^n_{t-1}$ is calculated using $\hat{\epsilon}$ and relevant parameters of the pre-defined forward process. Finally, the decoder $D$ produces the interpolated frame from the denoised latent $z^n_0$, with the help of feature pyramids $\phi^0, \phi^1$ extracted by the encoder $E$ from $I^0, I^1$. Full details are provided in the *Supplementary*.

**Network Architecture.** We employ the time-conditioned U-Net as in (Rombach et al. 2022) for $\epsilon_\theta$, but with one modification: all the vanilla self-attention blocks (Vaswani et al. 2017) are replaced with the aforementioned MaxViT blocks for computational efficiency. The conditioning mechanism for the U-Net is concatenation of $z^n_t$ and $z^0, z^1$ at the input. The architecture is detailed in the *Supplementary*.

## 4 Experimental Setup

**Implementation Details.** We set the downsampling factor of VQ-FIGAN to $f = 32$, by repeating the `ResNetBlock+Conv3x3` layer in the encoder and the corresponding layer in the decoder three times (see Figure 2). The size of the kernels output by the decoder is $K = 5$. Regarding the diffusion processes, following (Rombach et al. 2022), we adopt a linear noise schedule and a codebook size of 8192 for vector quantization in VQ-FIGAN. We sample from all diffusion models with the DDIM (Song, Meng, and Ermon 2021) sampler for 200 steps (details provided in the *Supplementary*). We also follow (Rombach et al. 2022) to train the VQ-FIGAN using the ADAM (Kingma and Ba 2015) optimizer and the denoising U-Net using the Adam-W optimizer (Loshchilov and Hutter 2019), with the initial learning rates set to $10^{-5}$ and $10^{-6}$ respectively. All models were trained until convergence, which corresponds to around 70 epochs for VQ-FIGAN, and around 60 epochs for the U-Net. NVIDIA RTX 3090 GPUs were used for all training and evaluation.

**Training Dataset.** We utilize the most commonly used training set in VFI, Vimeo90k (Xue et al. 2019). However, previous works (Sim, Oh, and Kim 2021) have discussed the limited range of motion magnitudes and the diversity of Vimeo90k. To better test the learning capability and performance of VFI methods on a wider range of scenarios, we follow (Danier, Zhang, and Bull 2022c) to additionally incorporate samples from the BVI-DVC dataset (Ma, Zhang, and Bull 2021). The final training set thus comprises 64612 and 17600 frame triplets (using only the three frames in the center) from Vimeo90k-septuplets and BVI-DVC respectively. For data augmentation, we randomly crop $256 \times 256$ patches and perform random flipping and temporal order reversing. It is noted that most existing works use only Vimeo90k-triplet for training, so for reference, we also provide evaluation results for LDMVFI trained on this dataset alone (see *Supplementary*).

**Test Datasets.** We evaluate models on the most commonly used VFI benchmarks, including Middlebury (Baker et al. 2011), UCF-101 (Soomro, Zamir, and Shah 2012), DAVIS (Perazzi et al. 2016), SNU-FILM (Choi et al. 2020), and VFITex (Danier, Zhang, and Bull 2022c). These test sets cover resolutions from $225 \times 225$ up to 4K, and various levels of VFI difficulties. To further assess the perceptual performance, we perform user study using the BVI-HFR (Mackin,

| | Middlebury | UCF-101 | DAVIS | VFITex | RT (sec) | #P (M) |
|---|---|---|---|---|---|---|
| BMBC | 0.023 / **0.037** / 12.974 | 0.034 / 0.045 / 33.171 | 0.125 / 0.185 / 15.354 | 0.220 / 0.282 / 50.393 | 0.51 | 11.0 |
| AdaCoF | 0.031 / 0.052 / 15.633 | 0.034 / 0.046 / 32.783 | 0.148 / 0.198 / 17.194 | 0.204 / 0.273 / 42.255 | 0.01 | 21.8 |
| CDFI | 0.022 / 0.043 / 12.224 | 0.036 / 0.049 / 33.742 | 0.157 / 0.211 / 18.098 | 0.218 / 0.286 / 43.498 | 0.02 | 5.0 |
| XVFI | 0.036 / 0.070 / 16.959 | 0.038 / 0.050 / 33.868 | 0.129 / 0.185 / 16.163 | 0.188 / 0.255 / 42.055 | 0.08 | 5.6 |
| ABME | 0.027 / 0.040 / **11.393** | 0.058 / 0.069 / 37.066 | 0.151 / 0.209 / 16.931 | 0.254 / 0.341 / 53.317 | 0.27 | 18.1 |
| IFRNet | 0.020 / 0.039 / 12.256 | 0.032 / 0.044 / 28.803 | 0.114 / 0.170 / 14.227 | 0.200 / 0.273 / 42.266 | 0.02 | 5.0 |
| VFIformer | 0.031 / 0.065 / 15.634 | 0.039 / 0.051 / 34.112 | 0.191 / 0.242 / 21.702 | OOM | 1.74 | 5.0 |
| ST-MFNet | N/A / N/A / N/A | 0.036 / 0.049 / 34.475 | 0.125 / 0.181 / 15.626 | 0.216 / 0.276 / 41.971 | 0.14 | 21.0 |
| FLAVR | N/A / N/A / N/A | 0.035 / 0.046 / 31.449 | 0.209 / 0.248 / 22.663 | 0.234 / 0.295 / 56.690 | 0.02 | 42.1 |
| MCVD | 0.123 / 0.138 / 41.053 | 0.155 / 0.169 / 102.054 | 0.247 / 0.293 / 28.002 | OOM | 52.55 | 27.3 |
| LDMVFI | **0.019** / 0.044 / 16.167 | **0.026 / 0.035 / 26.301** | **0.107 / 0.153 / 12.554** | **0.150 / 0.207 / 32.316** | 8.48 | 439.0 |

Table 1: Quantitative comparison (LPIPS↓ / FloLPIPS↓ / FID↓) of LDMVFI ($f = 32$) and 10 tested methods on Middlebury, UCF-101, DAVIS and VFITex. Note ST-MFNet and FLAVR require four input frames so cannot be evaluated on Middlebury dataset which contains frame triplets. For each column, the best result is **boldfaced**. The last two columns show the average run time (RT) needed to to interpolate one 480p frame, and the number of parameters (#P) in each model.

| | SNU-FILM-Easy | SNU-FILM-Medium | SNU-FILM-Hard | SNU-FILM-Extreme |
|---|---|---|---|---|
| BMBC | 0.020 / 0.031 / 6.162 | 0.034 / 0.059 / 12.272 | 0.068 / 0.118 / 25.773 | 0.145 / 0.237 / 49.519 |
| AdaCoF | 0.021 / 0.033 / 6.587 | 0.039 / 0.066 / 14.173 | 0.080 / 0.131 / 27.982 | 0.152 / 0.234 / 52.848 |
| CDFI | 0.019 / 0.031 / 6.133 | 0.036 / 0.066 / 12.906 | 0.081 / 0.141 / 29.087 | 0.163 / 0.255 / 53.916 |
| XVFI | 0.022 / 0.037 / 7.401 | 0.039 / 0.072 / 16.000 | 0.075 / 0.138 / 29.483 | 0.142 / 0.233 / 54.449 |
| ABME | 0.022 / 0.034 / 6.363 | 0.042 / 0.076 / 15.159 | 0.092 / 0.168 / 34.236 | 0.182 / 0.300 / 63.561 |
| IFRNet | 0.019 / 0.030 / 5.939 | 0.033 / 0.058 / 12.084 | 0.065 / 0.122 / **25.436** | 0.136 / 0.229 / 50.047 |
| ST-MFNet | 0.019 / 0.031 / 5.973 | 0.036 / 0.061 / **11.716** | 0.073 / 0.123 / 25.512 | 0.148 / 0.238 / 53.563 |
| FLAVR | 0.022 / 0.034 / 6.320 | 0.049 / 0.077 / 15.006 | 0.112 / 0.169 / 34.746 | 0.217 / 0.303 / 72.673 |
| MCVD | 0.199 / 0.230 / 32.246 | 0.213 / 0.243 / 37.474 | 0.250 / 0.292 / 51.529 | 0.320 / 0.385 / 83.156 |
| LDMVFI | **0.014 / 0.024 / 5.752** | **0.028 / 0.053** / 12.485 | **0.060 / 0.114** / 26.520 | **0.123 / 0.204 / 47.042** |

Table 2: Quantitative comparison results on SNU-FILM (note VFIformer is not included because the GPU goes out of memory).

Zhang, and Bull 2018) dataset which covers a wide range of texture and motion types.

**Evaluation Methods.** As the main focus of this work is on improving the perceptual quality of interpolated content, we adopt a perceptual image quality metric LPIPS (Zhang et al. 2018), and a bespoke VFI metric, FloLPIPS (Danier, Zhang, and Bull 2022b) for performance evaluation. These metrics have shown superior correlation with human judgments of VFI quality compared to commonly used quality measurements, PSNR and SSIM (Wang et al. 2004). We also evaluate FID (Heusel et al. 2017) which measures the similarity between the distributions of interpolated and ground-truth frames; this was previously used as a perceptual metric for video compression (Yang, Timofte, and Van Gool 2022), enhancement (Yang 2021) and colorization (Kang et al. 2023).

To measure the true perceptual performance of VFI methods, we also conducted a psychophysical experiment, in which the proposed method was compared against the state of the art (see Sec. 5.2). For completeness, we also provide benchmark results based on PSNR and SSIM in the *Supplementary*, noting that these are limited in reflecting the perceptual quality of interpolated content (Danier, Zhang, and Bull 2022d) and are therefore not the focus of this paper.

# 5 Experiments

## 5.1 Quantitative Evaluation

The proposed LDMVFI was compared against 10 recent state-of-the-art VFI methods, including BMBC (Park et al. 2020), AdaCoF (Lee et al. 2020), CDFI (Ding et al. 2021), XVFI (Sim, Oh, and Kim 2021), ABME (Park, Lee, and Kim 2021), IFRNet (Kong et al. 2022), VFIformer (Lu et al. 2022), ST-MFNet (Danier, Zhang, and Bull 2022c), FLAVR (Kalluri et al. 2023), and MCVD (Voleti, Jolicoeur-Martineau, and Pal 2022). It it noted that MCVD is the only existing diffusion-based VFI method. All these models were re-trained on our training dataset for fair comparison.

**Performance.** Table 1 shows the performance of the evaluated methods on the Middlebury, UCF-101, DAVIS, and VFITex test sets. It should be noted that the FID scores on Middlebury and UCF-101 might be unreliable because they contain very few test frames (12 and 100 respectively). It can be observed from the table that LDMVFI outperforms all the other VFI methods in most cases, and the performance gain against the second best method is most significant (approx. 20%) on VFITex which contains mainly dynamic textures (e.g. fire, water, foliage) and exhibits complex motions (see further discussion on this in the *Supplementary*). The model

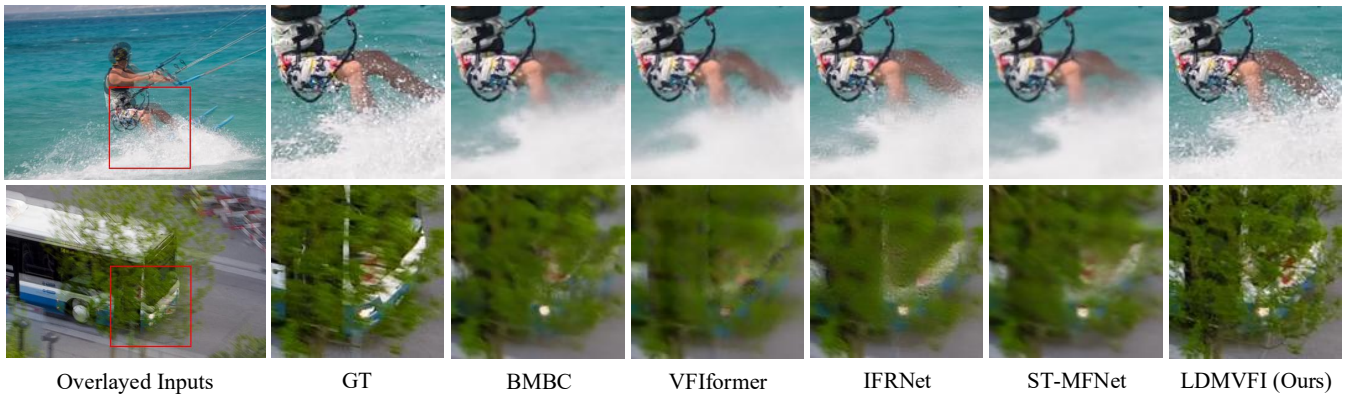


Figure 3: Visual examples of frames interpolated by the state-of-the-art methods and the proposed LDMVFI. Under large and complex motions, our method preserves the most high-frequency details, delivering superior perceptual quality.

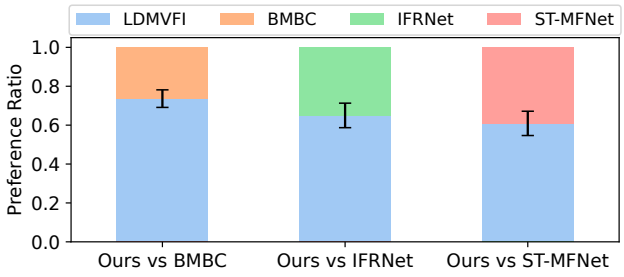


Figure 4: Results of the user study in terms of preference ratio. Error bar reflects the standard error over test sequences.

performance on the four splits of the SNU-FILM dataset are summarized in Table 2, which again demonstrates the favorable perceptual quality of videos interpolated by LDMVFI. It is also significant that the other diffusion-based VFI method, MCVD, does not perform satisfactorily overall, which implies that directly applying the original diffusion model formulation to VFI is not sufficient to enhance performance. This further shows the effectiveness of LDMVFI. Multi-frame (i.e. $\times 4$) interpolation results are provided in the *Supplementary*.

**Complexity.** The average time taken to interpolate a 480p frame on an RTX 3090 GPU and the number of parameters of each model are shown in Table 1 (the last two columns). It is observed that the inference speed of LDMVFI is much lower compared to other methods, and this is mainly due to the iterative denoising operation performed during sampling. This is a common drawback of existing diffusion models. Various methods have been proposed to speed up sampling process of diffusion models (Karras et al. 2022), which can also be applied to LDMVFI. The number of parameters in LDMVFI is also large, and this is because we adopted (with some modifications, see Sec. 3.2) the existing denoising U-Net (Rombach et al. 2022) designed for generic image generation. We leave the design of a more efficient denoising U-Net and the improvement of LDMVFI sampling speed as future work. See more discussion on the limitations in the *Supplementary*.

## 5.2 Subjective Experiment

To further confirm the superior perceptual quality of the videos interpolated by LDMVFI compared to the state of the art, and also to measure its temporal consistency, we conducted a subjective experiment where human participants were hired to rate the quality of videos interpolated by ours and competing methods.

**Test Videos.** We use the 22 high-quality full HD 30fps videos from the BVI-HFR (Mackin, Zhang, and Bull 2018) dataset as source content. These videos cover a wide range of video features related to motion and texture as shown by the analysis in the original paper (Mackin, Zhang, and Bull 2018), allowing for a more thorough benchmarking of VFI methods. To generate the test content, the 22 videos were first truncated to 5 seconds (150 frames) following (Moss et al. 2015). Then we used four different VFI methods to interpolate all videos to 60fps. Other than LDMVFI, the tested methods include ST-MFNet, IFRNet and BMBC, which showed the most competitive quantitative performance on the more challenging test sets (e.g. DAVIS, SNU-FILM-extreme). As a result, we obtain 88 test videos generated by the four VFI methods.

**Test Methodology.** Following previous works (Liu et al. 2017; Niklaus and Liu 2018; Kalluri et al. 2023), the 2AFC approach is adopted for the subjective experiment, where the participant is asked to choose the one with better perceived quality from a pair of videos. Specifically, in each test session, the participant was displayed 66 pairs of videos where one video in each pair is interpolated by LDMVFI and the other is interpolated by ST-MFNet, IFRNet or BMBC. The display order of the 66 pairs, and the order of test videos within each pair are both randomized. The user is unaware of the methods used for generating the videos. Each pair is presented twice to the participant, who is then asked to provide an answer to the question "which of the two videos is of higher quality?". Twenty participants were hired in total. See the *Supplementary* for more details.

| | $f$ | AE Model | Cond. Mode | Middlebury | UCF-101 | DAVIS |
|---|---|---|---|---|---|---|
| V1 | 32 | frame | concat | 0.077 / 0.085 / 40.399 | 0.063 / 0.067 / 60.742 | 0.168 / 0.200 / 21.812 |
| V2 | 32 | MaxCA+frame | concat | 0.028 / 0.045 / 19.485 | 0.032 / 0.041 / 29.578 | 0.135 / 0.176 / 19.980 |
| V3 | 8 | MaxCA+kernel | concat | 0.018 / 0.046 / 19.552 | 0.028 / 0.036 / 27.278 | 0.125 / 0.168 / 15.535 |
| V4 | 16 | MaxCA+kernel | concat | 0.017 / 0.037 / 12.539 | 0.026 / 0.035 / 26.805 | 0.107 / 0.154 / 12.720 |
| V5 | 64 | MaxCA+kernel | concat | 0.022 / 0.044 / 15.041 | 0.026 / 0.036 / 26.055 | 0.114 / 0.157 / 12.330 |
| V6 | 32 | MaxCA+kernel | N/A | 0.100 / 0.108 / 38.538 | 0.042 / 0.053 / 32.189 | 0.193 / 0.214 / 8.639 |
| Ours | 32 | MaxCA+kernel | concat | 0.019 / 0.044 / 16.167 | 0.026 / 0.035 / 26.301 | 0.107 / 0.153 / 12.554 |

Table 3: Ablation experiment results (LPIPS↓ / FloLPIPS↓ / FID↓), showing performance of variants of the proposed LDMVFI.

**Results.** After collecting all the user data, for each of the 22 source sequences, the ratio of users that preferred LDMVFI is calculated. Figure 4 reports the average preference ratios for LDMVFI and the standard error over the sequences. It can be seen that in all comparisons, LDMVFI achieved higher preference ratios. T-test analysis (see the *Supplementary*) on the sequence-wise preference ratios shows that the advantage of LDMVFI over the other three tested methods is statistically significant at 95% confidence level. These results further confirm the superior perceptual performance of LDMVFI.

**Visual Examples.** Figure 3 shows example frames interpolated by LDMVFI and the competing methods, clearing demonstrating that the LDMVFI results have the best visual quality. See more visual examples in the *Supplementary*.

### 5.3 Ablation Study

In this section we experimentally validate and study different components and hyper-parameters in LDMVFI. The ablation study results are summarized in Table 3, which shows evaluation results of 6 variants of the proposed model on three test sets. See the *Supplementary* for full ablation study results, where more aspects of the model are analyzed, including discussions on the variability of LDMVFI results.

**Effectiveness of VQ-FIGAN.** To validate the effectiveness of the proposed VQ-FIGAN design, we tested two variants of the model: V1 and V2. V2 outputs the frame $\hat{I}^n$ directly instead of predicting the deformable kernels, i.e. the convolutional heads after the last `Conv3x3` layer are removed (see Figure 2). For V1, a further change is made by removing the feature-aided reconstruction process that involves $\phi^0, \phi^1$ and replacing `MaxCABlocks` in the decoder with `ResNetBlocks`. As such, there is no information from the neighbor frames during decoding. Note that V1 is similar to the original VQGAN (Esser, Rombach, and Ommer 2021). Table 3 shows that without the deformable convolution-based synthesis, the performance of V2 sees an evident decrease. Furthermore, V1 shows a more severe drop in performance indicating the effectiveness of using features of neighbor frames during reconstruction.

**Downsampling Factor $f$.** Here we study how the dimension of the latent space affects the VFI performance. Specifically, to obtain V4 ($f = 16$), we remove one `ResNetBlock+Conv3x3` layer and one `ResNetBlock+MaxCABlock+Upsample+Conv3x3` layer from the encoder and decoder of VQ-FIGAN respectively (see Figure 2). We repeat this process once to obtain V3 ($f = 8$). To create V5 ($f = 64$) we add these layers instead of removing them. Table 3 shows that as $f$ increases from 8 to 32, there is generally an increasing trend in model performance (except that V4 outperformed LDMVFI on Middlebury). Such improvement is more obvious on DAVIS which mainly contains challenging large motion content. However, looking at $f = 64$, the general performance deteriorates (in terms of LPIPS and FloLPIPS). The reason for this can be that to a reasonable extent, increasing $f$ allows the VQ-FIGAN decoder to make use of more information from the neighboring frames which can benefit frame interpolation, while preserving sufficient information for conditional generation in the latent space. However, if the downsampling is too aggressive, the information needed to perform reverse latent diffusion can be insufficient, resulting in degraded quality of latent generation. A similar trade-off between downsampling factor and model performance was also observed in (Rombach et al. 2022).

**Effectiveness of Diffusion Process.** Here (V6) the denoising U-Net, hence the reverse diffusion process, is removed, and the decoder directly takes as input the concatenation of $z^0$ and $z^1$. Table 3 shows that without the diffusion process, the performance drops significantly, implying the effectiveness of the latent diffusion.

## 6 Conclusion

In this work we propose LDMVFI, the first approach that addresses video frame interpolation as a conditional generation problem using latent diffusion models. It contains two major components: an autoencoding model that provides access to a compact latent space, and a denoising U-Net that performs reverse diffusion on latent representations. To leverage latent diffusion models for VFI, we present several innovative designs including a VFI-specific autoencoding network, VQ-FIGAN, which employs efficient self-attention modules and deformable kernel-based frame synthesis techniques. LDMVFI was comprehensively evaluated on a wide range of test sets (including 4K content) using both quantitative metrics and subjective experiments. The results demonstrate its favorable perceptual performance over the state of the art.

## Acknowledgements

This work was supported in part by the China Scholarship Council, in part by the University of Bristol, and in part by the UK Research and Innovation (UKRI) MyWorld Strength in Places Program.

## References

Baker, S.; Scharstein, D.; Lewis, J.; Roth, S.; Black, M. J.; and Szeliski, R. 2011. A database and evaluation methodology for optical flow. *Int. J. of Comput. Vis.*, 92(1): 1–31.

Bao, W.; Lai, W.-S.; Zhang, X.; Gao, Z.; and Yang, M.-H. 2021. MEMC-Net: Motion estimation and motion compensation driven neural network for video interpolation and enhancement. *IEEE Trans. on Pattern Anal. Mach. Intell.*, 43(3): 933–948.

Cheng, X.; and Chen, Z. 2020. Video frame interpolation via deformable separable convolution. In *Proc. of the AAAI Conf. on Artificial Intell.*, 10607–10614.

Cheng, X.; and Chen, Z. 2022. Multiple video frame interpolation via enhanced deformable separable convolution. *IEEE Trans. on Pattern Anal. Mach. Intell.*, 44(10): 7029–7045.

Choi, M.; Kim, H.; Han, B.; Xu, N.; and Lee, K. M. 2020. Channel attention is all you need for video frame interpolation. In *Proc. of the AAAI Conf. on Artificial Intell.*, volume 34, 10663–10671.

Dai, J.; Qi, H.; Xiong, Y.; Li, Y.; Zhang, G.; Hu, H.; and Wei, Y. 2017. Deformable convolutional networks. In *Proc. of the IEEE Int. Conf. on Comput. Vis.*, 764–773.

Danier, D.; Zhang, F.; and Bull, D. 2022a. Enhancing deformable convolution based video frame interpolation with coarse-to-fine 3D CNN. In *IEEE Int. Conf. on Image Process.*, 1396–1400.

Danier, D.; Zhang, F.; and Bull, D. 2022b. FloLPIPS: A Bespoke Video Quality Metric for Frame Interpolation. In *IEEE Picture Coding Symposium*, 283–287.

Danier, D.; Zhang, F.; and Bull, D. 2022c. ST-MFNet: A spatio-temporal multi-flow network for frame interpolation. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 3521–3531.

Danier, D.; Zhang, F.; and Bull, D. 2022d. A subjective quality study for video frame interpolation. In *IEEE Int. Conf. on Image Process.*, 1361–1365.

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

Ding, T.; Liang, L.; Zhu, Z.; and Zharkov, I. 2021. CDFI: Compression-Driven Network Design for Frame Interpolation. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 8001–8011.

Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; Uszkoreit, J.; and Houlsby, N. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *Int. Conf. on Learn. Represent.*

Esser, P.; Rombach, R.; and Ommer, B. 2021. Taming transformers for high-resolution image synthesis. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 12873–12883.

Flynn, J.; Neulander, I.; Philbin, J.; and Snavely, N. 2016. Deepstereo: Learning to predict new views from the world's imagery. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 5515–5524.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2020. Generative adversarial networks. *Communications of the ACM*, 63(11): 139–144.

Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; and Hochreiter, S. 2017. GANs trained by a two time-scale update rule converge to a local nash equilibrium. *Adv. in Neural Inform. Process. Syst.*, 30.

Ho, J.; Jain, A.; and Abbeel, P. 2020. Denoising diffusion probabilistic models. *Adv. in Neural Inform. Process. Syst.*, 33: 6840–6851.

Ho, J.; Saharia, C.; Chan, W.; Fleet, D. J.; Norouzi, M.; and Salimans, T. 2022. Cascaded Diffusion Models for High Fidelity Image Generation. *J. of Machine Learning Research*, 23(47): 1–33.

Isola, P.; Zhu, J.-Y.; Zhou, T.; and Efros, A. A. 2017. Image-to-image translation with conditional adversarial networks. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 1125–1134.

Jiang, H.; Sun, D.; Jampani, V.; Yang, M.-H.; Learned-Miller, E.; and Kautz, J. 2018. Super slomo: High quality estimation of multiple intermediate frames for video interpolation. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 9000–9008.

Kalluri, T.; Pathak, D.; Chandraker, M.; and Tran, D. 2023. FLAVR: Flow-agnostic video representations for fast frame interpolation. In *Proc. of the IEEE Winter Conf. on Applications of Comput. Vis.*, 2071–2082.

Kang, X.; Lin, X.; Zhang, K.; Hui, Z.; Xiang, W.; He, J.-Y.; Li, X.; Ren, P.; Xie, X.; Timofte, R.; et al. 2023. NTIRE 2023 video colorization challenge. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 1570–1581.

Karargyris, A.; and Bourbakis, N. 2010. Three-dimensional reconstruction of the digestive wall in capsule endoscopy videos using elastic video interpolation. *IEEE Trans. on Medical Imaging*, 30(4): 957–971.

Karras, T.; Aittala, M.; Aila, T.; and Laine, S. 2022. Elucidating the Design Space of Diffusion-Based Generative Models. In Oh, A. H.; Agarwal, A.; Belgrave, D.; and Cho, K., eds., *Adv. in Neural Inform. Process. Syst.*, volume 35, 26565–26577.

Kingma, D. P.; and Ba, J. 2015. Adam: A Method for Stochastic Optimization. In *Int. Conf. on Learn. Represent.*

Kingma, D. P.; and Welling, M. 2014. Auto-Encoding Variational Bayes. In *Int. Conf. on Learn. Represent.*

Kong, L.; Jiang, B.; Luo, D.; Chu, W.; Huang, X.; Tai, Y.; Wang, C.; and Yang, J. 2022. IFRNet: Intermediate feature refine network for efficient frame interpolation. In *Proc. of*

*the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 1969–1978.

Lee, H.; Kim, T.; Chung, T.-y.; Pak, D.; Ban, Y.; and Lee, S. 2020. AdaCoF: Adaptive collaboration of flows for video frame interpolation. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 5316–5325.

Liu, Z.; Yeh, R. A.; Tang, X.; Liu, Y.; and Agarwala, A. 2017. Video frame synthesis using deep voxel flow. In *Proc. of the IEEE Int. Conf. on Comput. Vis.*, 4463–4471.

Loshchilov, I.; and Hutter, F. 2019. Decoupled Weight Decay Regularization. In *Int. Conf. on Learn. Represent.*

Lu, L.; Wu, R.; Lin, H.; Lu, J.; and Jia, J. 2022. Video frame interpolation with transformer. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 3532–3542.

Ma, D.; Zhang, F.; and Bull, D. 2021. BVI-DVC: A Training Database for Deep Video Compression. *IEEE Trans. on Multimedia*, 1–1.

Mackin, A.; Zhang, F.; and Bull, D. R. 2018. A study of high frame rate video formats. *IEEE Trans. on Multimedia*, 21(6): 1499–1512.

Men, H.; Hosu, V.; Lin, H.; Bruhn, A.; and Saupe, D. 2020. Visual quality assessment for interpolated slow-motion videos based on a novel database. In *IEEE Int. Conf. on Quality of Multimedia Experience*, 1–6.

Moss, F. M.; Wang, K.; Zhang, F.; Baddeley, R.; and Bull, D. R. 2015. On the optimal presentation duration for subjective video quality assessment. *IEEE Trans. on Circuit Syst. Video Technol.*, 26(11): 1977–1987.

Niklaus, S.; and Liu, F. 2018. Context-aware synthesis for video frame interpolation. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 1701–1710.

Niklaus, S.; and Liu, F. 2020. Softmax splatting for video frame interpolation. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 5437–5446.

Niklaus, S.; Mai, L.; and Liu, F. 2017. Video frame interpolation via adaptive convolution. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 670–679.

Park, J.; Ko, K.; Lee, C.; and Kim, C.-S. 2020. BMBC: Bilateral motion estimation with bilateral cost volume for video interpolation. In *Proc. of the Eur. Conf. on Comput. Vis.*, 109–125. Springer.

Park, J.; Lee, C.; and Kim, C.-S. 2021. Asymmetric Bilateral Motion Estimation for Video Frame Interpolation. In *Proc. of IEEE Int. Conf. on Comput. Vis.*, 14539–14548.

Perazzi, F.; Pont-Tuset, J.; McWilliams, B.; Van Gool, L.; Gross, M.; and Sorkine-Hornung, A. 2016. A benchmark dataset and evaluation methodology for video object segmentation. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 724–732.

Rigamonti, R.; Sironi, A.; Lepetit, V.; and Fua, P. 2013. Learning separable filters. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 2754–2761.

Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; and Ommer, B. 2022. High-resolution image synthesis with latent diffusion models. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 10684–10695.

Saha, A.; Pentapati, S. K.; Shang, Z.; Pahwa, R.; Chen, B.; Gedik, H. E.; Mishra, S.; and Bovik, A. C. 2023. Perceptual Video Quality Assessment: The Journey Continues! *Frontiers in Signal Processing*, 3.

Sim, H.; Oh, J.; and Kim, M. 2021. XVFI: eXtreme Video Frame Interpolation. In *Proc. of the IEEE Int. Conf. on Comput. Vis.*, 14489–14498.

Simonyan, K.; and Zisserman, A. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *Int. Conf. on Learn. Represent.*

Siyao, L.; Zhao, S.; Yu, W.; Sun, W.; Metaxas, D.; Loy, C. C.; and Liu, Z. 2021. Deep animation video interpolation in the wild. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 6587–6595.

Song, J.; Meng, C.; and Ermon, S. 2021. Denoising Diffusion Implicit Models. In *Int. Conf. on Learn. Represent.*

Soomro, K.; Zamir, A. R.; and Shah, M. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*.

Tu, Z.; Talebi, H.; Zhang, H.; Yang, F.; Milanfar, P.; Bovik, A.; and Li, Y. 2022. Maxvit: Multi-axis vision transformer. In *Proc. of the Eur. Conf. on Comput. Vis.*, 459–479. Springer.

Van Den Oord, A.; Vinyals, O.; et al. 2017. Neural discrete representation learning. *Adv. in Neural Inform. Process. Syst.*, 30.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Adv. in Neural Inform. Process. Syst.*, 30.

Voleti, V.; Jolicoeur-Martineau, A.; and Pal, C. 2022. MCVD - Masked Conditional Video Diffusion for Prediction, Generation, and Interpolation. In *Adv. in Neural Inform. Process. Syst.*, volume 35, 23371–23385.

Wang, Z.; Bovik, A. C.; Sheikh, H. R.; and Simoncelli, E. P. 2004. Image quality assessment: from error visibility to structural similarity. *IEEE Trans. on Image Process.*, 13(4): 600–612.

Wu, C.-Y.; Singhal, N.; and Krahenbuhl, P. 2018. Video compression through image interpolation. In *Proc. of the Eur. Conf. on Comput. Vis.*, 416–431.

Xue, T.; Chen, B.; Wu, J.; Wei, D.; and Freeman, W. T. 2019. Video enhancement with task-oriented flow. *Int. J. of Comput. Vis.*, 127(8): 1106–1125.

Yang, R. 2021. NTIRE 2021 challenge on quality enhancement of compressed video: Dataset and study. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 667–676.

Yang, R.; Timofte, R.; and Van Gool, L. 2022. Perceptual Learned Video Compression with Recurrent Conditional GAN. In *Processings of the Int. Joint Conf. on Artificial Intell.*, 1537–1544.

Zhang, R.; Isola, P.; Efros, A. A.; Shechtman, E.; and Wang, O. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *Proc. of the IEEE Conf. on Comput. Vis. and Pattern Recog.*, 586–595.