# A Perceptual Quality Metric for Video Frame Interpolation

Qiqi Hou, Abhijay Ghildyal, and Feng Liu

Portland State University

{qiqi2,abhijay,fliu}@pdx.edu

Fig. 1: Unique distortions in video frame interpolation results.

**Abstract.** Research on video frame interpolation has made significant progress in recent years. However, existing methods mostly use off-the-shelf metrics to measure the quality of interpolation results with the exception of a few methods that employ user studies, which is time-consuming. As video frame interpolation results often exhibit unique artifacts, existing quality metrics sometimes are not consistent with human perception when measuring the interpolation results. Some recent deep learning-based perceptual quality metrics are shown more consistent with human judgments, but their performance on videos is compromised since they do not consider temporal information. In this paper, we present a dedicated perceptual quality metric for measuring video frame interpolation results. Our method learns perceptual features directly from videos instead of individual frames. It compares pyramid features extracted from video frames and employs Swin Transformer blocks-based spatio-temporal modules to extract spatio-temporal information. To train our metric, we collected a new video frame interpolation quality assessment dataset. Our experiments show that our dedicated quality metric outperforms state-of-the-art methods when measuring video frame interpolation results.

## 1 Introduction

Video frame interpolation aims to generate frames between consecutive frames. It has attracted a lot of attention because of its wide applications in video editing [53], video generation [38, 39], optical flow estimation [49, 89], and video compression [87]. Consequentially, great progress has been made and many video frame interpolation methods are available [4, 15, 30, 31, 40, 42, 57, 58, 64, 72, 73].

To evaluate the quality of video frame interpolation results, most interpolation methods rely on traditional metrics, such as Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index Measure (SSIM) [79]. These metrics estimate perceptual similarity between images by comparing the difference between pixels or carefully designed low-level visual patterns. While such methods achieve promising results, adapting them to measure video frame interpolation results is challenging. First, these metrics have been designed for general tasks such as video compression and other common low-level computer vision tasks, like super resolution, deblurring, and denoising, not specifically for video frame interpolation. Hence, these metrics are not optimal for measuring the quality of video frame interpolation results. As shown in Figure 1, video frame interpolation methods exhibit unique distortions. For example, the telegraph pole is twisted out of shape and the wings of the duck have ghosting artifacts. Second, traditional metrics rely on low-level features, such as per-pixel distance, and structure similarity [79]. However, human perception is highly complex, and hence, low-level features are insufficient for assessing the perceptual quality [97]. These shortcomings motivate us to develop a dedicated quality metric for video frame interpolation.

Our research is inspired by the recent deep learning approaches to image quality assessment [6, 17, 20, 21, 32, 33, 35, 36, 41, 76, 91, 97]. Deep learning approaches, such as LPIPS [97], take an image and its reference image as input and output the perceptual score. Instead of comparing low-level features, they learn perceptual similarity from the internal features of deep convolutional networks. Such methods have shown great success for image perceptual quality assessment. However, applying these metrics to video frame interpolation results frame by frame ignores the temporal information in the videos, which often make these metrics inconsistent with human perception.

This paper presents a learned perceptual video similarity metric dedicated to video frame interpolation results. It takes an interpolated video and its reference ground-truth video as input and outputs the perceptual similarity between them. It builds upon state-of-the-art image perceptual quality metrics but extends to videos, leading to a naturally spatio-temporal architecture. Specifically, our network extracts the features from each frame. Then, we design a dedicated spatio-temporal module to capture the spatio-temporal information from videos. This spatio-temporal module consists of the Swin Transformer blocks [47], which can explore the interaction among different regions, learn to attend the important regions, and extract spatio-temporal video information.

To train our network, we collected a large Video Frame Interpolation Perceptual Similarity (VFIPS) dataset that contains 25,887 video triplets. Each video triplet contains two interpolated videos, their reference video, and the corresponding perceptual preference. Our dataset consists of various artifacts generated by eleven state-of-the-art video frame interpolation methods [4, 5, 15, 30, 40, 42, 46, 57–59, 64]. With this dataset, we train our video perceptual metric using Binary Cross Entropy (BCE) as the loss function in an end-to-end manner.

Our experiments show that our metric outperforms existing metrics significantly when measuring video frame interpolation results.

This paper contributes to the research on video frame interpolation and perceptual quality assessment by 1) providing the first video perceptual similarity metric dedicated to video frame interpolation, 2) designing a novel neural network architecture for video perceptual quality assessment based on the Swin Transformers, and 3) building a large video frame interpolation perceptual similarity dataset.

## 2    Related Work

This paper investigates quality assessment of video frame interpolation results. Below, we first briefly survey video frame interpolation methods. Then we discuss traditional video quality metrics and recent deep-learning-based approaches. We also discuss video quality assessment datasets and vision transformers.

**Video frame interpolation.** Video frame interpolation aims to estimate intermediate frames between two consecutive frames. They can be classified into three categories: kernel-based methods [59–62, 92], phase-based methods [54, 55], and flow-based methods [4, 5, 13, 16, 18, 28–30, 40, 42, 46, 48, 57, 58, 64, 65, 68, 70, 73, 94, 96]. Kernel-based methods [59–61, 92] estimate a kernel for each pixel in the frame indicating the weights for its neighboring pixels to synthesize the new pixel. They generate the synthesized frame in a local convolutional manner. Phase-based methods [54, 55] generate intermediate frames by per-pixel phase modification. Currently, most of methods are flow based [4, 5, 30, 40, 42, 46, 48, 57, 58, 64, 68, 70, 73, 94]. Typically, their networks contain two modules: optical flow module and frame synthesized module. The optical flow module is used to estimate the optical flow between input frames. By warping the input frame based on the estimated optical flow, the frame synthesized module synthesizes the intermediate frames. Most of the video frame interpolation methods use PSNR and SSIM as their evaluation metrics. Some methods [40, 58, 93] also explore Interpolation Error (IE) [2], Normalized Interpolation Error (NIE) [2], and Learned Perceptual Image Patch Similarity (LPIPS) [97] to measure their results. However, these metrics are insufficient to evaluate the perceptual similarity. Traditional quality metrics, such as PSNR, only use low-level features and LPIPS lacks temporal information. In this paper, we propose a learned video quality metric dedicated to video frame interpolation.

**Video quality metrics.** Based on the availability of the reference video, traditional video quality metrics can be classified into No-Reference (NR), Reduced-Reference (RR), and Full Reference (FR). Due to the limitation of space, we refer readers to a comprehensive survey for traditional methods [85]. Our method is most related to the FR methods, which have access to the reference video. Traditional FR methods infer the perceptual similarity from the differences from pixels or low-level features [8, 23, 27, 50, 69, 81–84, 86, 88, 90]. For instance, Wange *et al.* observed that the human visual system (HVS) is adaptive to structural similarity [79]. They evaluated the perceptual similarity from the structural defor-

mity between a reference and distortion image. In comparison to SSIM, Pinson *et al.* had a better correlation with HVS on video quality assessment [66]. They also considered multiple perceptual-based features, including blurring, unnatural motion, noise, color distortion, and block distortion. Li *et al.* proposed a multi-method metric to fuse multiple video perceptual quality metrics using a support vector machine [44]. It can preserve the strength of the individual metrics and achieve promising results. However, these metrics only use low-level features, which is insufficient for many nuances of human perception. Recently, with the success of deep learning, many research works introduce the deep learning method to video quality assessment [12, 78]. However, these metrics are designed for the traditional artifacts like video compression, while our method focuses on artifacts that arise from video frame interpolation.

**Deep learning based quality metrics.** Our method is most related to the recent deep learning based metrics for image quality assessment [6, 17, 20, 21, 32, 33, 35, 36, 41, 76, 91, 97]. Instead of using low-level features, these methods learn features from deep convolutional networks and train a network in an end-to-end manner. For full-reference image quality assessment (FR-IQA), Kim *et al.* estimated the perceptual similarity using a deep network [32]. Their method takes a distorted images and an error map as input and estimates the sensitivity map. They get the subjective score by multiplying the sensitivity map with the error map. Zhang *et al.* explored the convolutional features from a classification network [97]. By comparing the features extracted using the convolutional network, their method is able to predict the perceptual similarity robustly. They also built a large-scale dataset for image perceptual quality assessment, which contains traditional distortions and CNN-based distortions. Their dataset also contains distortions from frame interpolation. However, it is designed for image quality assessment and only contains single images. In contrast, our dataset contains videos for frame interpolation, which enable us to learn the temporal information. Ding *et al.* proposed an image quality metric with tolerance to texture resampling [21]. Bhardwaj *et al.* trained their metric using the information theory-guided loss function [6]. Czolbe *et al.* trained an image quality metric for GAN using Watson's perceptual model [17]. While these methods have achieved great success on image quality assessment, directly adapting these methods to videos will be disadvantageous to these methods because they ignore the spatio-temporal consistency. In contrast, our work builds upon these deep learning methods and considers the spatio-temporal information in our spatio-temporal network module, thus providing a dedicated solution to video frame interpolation quality assessment.

**Video quality datasets.** Traditional video quality datasets focus on video compression and transmission [1, 3, 24, 25, 56, 69, 71]. Video Quality Experts Group (VQEG) dataset was proposed for the secondary distribution of television [1]. Most videos in the VQEG dataset are interlaced. To address this problem, Kalpana *et al.* built the Laboratory for Image and Video Engineering (LIVE) Video Quality Database [3, 71]. It is designed for the H.264 compression, MPEG-2 compression, and video transmission. In their recent works, they extended the

dataset for the distortion in the streaming [24, 56]. Li *et al.* proposed a video quality dataset for video compressions, including H.264/AVC, HEVC, and VP9 [69]. Different from these datasets, our work focuses on distortions from video frame interpolation. Recently, Danier *et al.* proposed the BVI-VFI dataset for video frame interpolation [19]. It is a relatively small dataset and hence not suitable for training deep learning models. Therefore, we make use of it as a test dataset to compare our method against various traditional and recently proposed metrics.

**Vision Transformer.** There has been considerable interest in vision transformers due to their impressive performance in image classification [22, 43, 47], object detection [9], and image restoration [11]. Recently, the vision transformers have also been introduced for the image quality assessment [14, 95]. Their networks combined vision transformer [22] and CNN, and achieved promising results for image quality assessment. Our work adapts the recent proposed Swin Transformer [47] for video quality assessment.

## 3    Video Frame Interpolation Quality Dataset

We collected a Video Frame Interpolation Perceptual Similarity (VFIPS) dataset. It consists of a wide variety of distortions from video frame interpolation. Each sample consists of two videos synthesized from different interpolation methods, its reference video, and its perceptual judgments $h \in \{0, 0.33, 0.66, 1\}$. We can denote it as $\{\mathbf{V}_A, \mathbf{V}_B, \mathbf{V}_R, h\}$. Below, we briefly describe how we prepare and annotate this dataset.

### 3.1    Data Preparation

**Source videos**. From YouTube, we collected 96 videos under the Creative Commons Attribution license (reuse allowed), which enables us to share this dataset with the community. Among them, 45 videos are 120 fps, and 51 videos are 60 fps. Most videos last over 10 minutes. The resolutions of the source videos are either 1080p or 4K. We downsampled these videos to 540p with the bicubic downsampling function from OpenCV. From these videos, we extracted video clips for quality assessment. After eliminating the video clips containing cut shots or interlaced frames, we curate, in total, 23,856 12-frame video clips. These 12-frame sequences serve as a reference in our dataset.

**Videos synthesized by frame interpolation methods**. We generated the synthesized videos for each reference video with 11 state-of-the-art video frame interpolation methods, including SepConv [59], Super-SloMo [30], CtxSyn [57], CyclicGen [46], MEMC-Net [5], DAIN [4], CAIN [15], RRIN [42], AdaCoF [40], BMBC [64], and SoftSplat [58]. We set ×4 scale for 120-fps videos and ×2 scale for 60-fps videos.

For each reference sequence, we need to compare two video frame interpolation results. To make it easy to view and find differences between videos, we keep the size of the cropped video patches as $256 \times 256$. Specifically, we calculated the mean $\ell_1$ error map between two synthesized videos and selected the patch location with the highest error by sliding patch windows.

### 3.2   Annotation

Compared to annotating images, annotating videos often requires much greater effort from users. For instance, a video contains many frames, but the distortions might only show in one or just a few frames. In such cases, a user needs to play the video multiple times to analyze differences and make a final judgment. In our case, a user often takes more than 1 minute to annotate 1 sample. Hence, can we make use of some existing perceptual quality metrics to help us annotate more samples?

Recent developments in perceptual quality metrics show that considerable progress has been made with regard to correlating these metrics with human judgment [6, 17, 21, 32, 76, 97]. The result of a single image quality assessment can provide helpful prior knowledge for the quality assessment of videos. For instance, by visual inspection, we found that if all frames of one video have significantly better quality than the frames in another video, this video often has a better quality. Based on this observation, we make use of the widely adopted state-of-the-art image quality metric LPIPS [97] to help annotate samples where two videos have significantly different LPIPS scores. These LPIPS annotated examples are only used for training and not used for testing. While they are not perfect, they can be used to train a good VFI quality metric. A similar strategy was also adopted in the PIEAPP metric [67]. For samples which is hard to make a judgment via LPIPS, we collect human annotations.

**Automatic annotation.** We select 500 video triplets for our study to test the hypothesis that *when the difference in LPIPS scores of two videos is significant, the judgment via LPIPS is consistent with the human judgment*. We collect human annotations for these 500 triplets. Each video triplet contains one reference video and two synthesized videos. For each synthesized video, we calculate its LPIPS score as the mean of the LPIPS scores of the frames in that video. We tested whether the judgments via LPIPS are consistent with the human judgments. Evidence from this study suggests that the judgment via LPIPS can be reliably used for annotating triplets with large LPIPS difference.

Figure 2 shows the LPIPS difference versus the human judgment, where the LPIPS difference is the absolute difference between the LPIPS scores of a pair of videos. The judgment via LPIPS is consistent with the human judgment with the increase in the LPIPS difference. We can find that 97.6% judgments via LPIPS are consistent with the human judgments when the LPIPS difference is greater than 0.15. Based on this observation, we select 0.15 as the LPIPS-difference threshold. If a new video triplet's LPIPS difference is greater than 0.15, we take the LPIPS judgment as its ground truth label. For the rest of the video triplets where LPIPS difference is less than 0.15, we collect human annotations. It helps save time and effort involved in the recruitment and collection of high-quality human annotations. In the end, for our dataset, we have 18,939 video triplets with ground truth labels annotated via LPIPS.

**Manual annotation.** For the samples where LPIPS-difference is less than 0.15, we collect judgments on perceptual quality by humans.
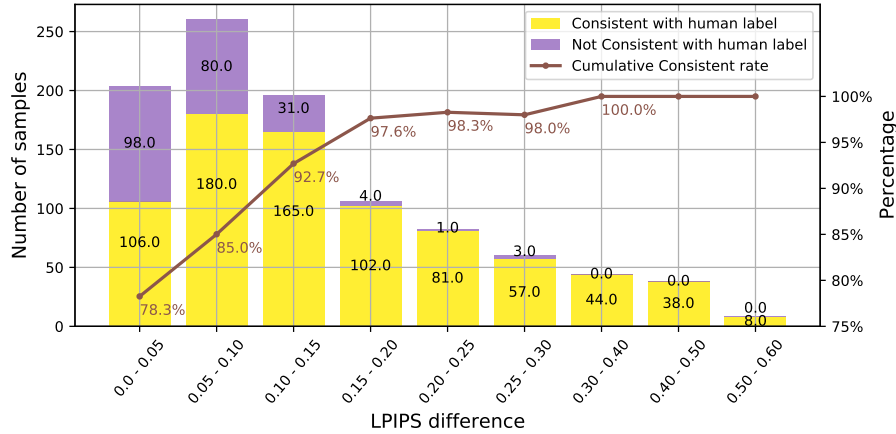
Fig. 2: LPIPS difference vs. human labeling result.

**User interface.** For the annotation, we presented participants with a reference video and two corresponding video frame interpolation results. The videos were played simultaneously to help participants compare the two synthesized videos. For judgment, participants were asked which of the two synthesized videos has a higher perceptual quality. We played the videos in a loop without interruption or human intervention. Figure 3 shows the user interface. It has four options {"A(Sure)", "A(Maybe)", "B(Maybe)", "B(Sure)"}. Please note that the middle point in Figure 3 is not an option.

**Playback setting.** To find the best playback setting, we tested video clips of length from 4 to 16 frames. If a video is too short, human annotators would not get enough temporal information to make a decision. But a too-long video often leads to ambiguity. For example, the quality of a video can be high in the first half but be poor in the last half. In such cases, it is difficult for users to make a decision. We empirically found that a video clip with 12 frames works well for users to annotate the quality. We also explored the playback speed in our study. At a high fps, such as 8 fps, users will find it extremely hard to compare the distortions, as each frame only shows for a very short time. However, a low fps will make it harder to judge the temporal consistency. We empirically found that 2 fps works well.

We collected annotations on 5,948 video triplet samples from 41 participants. Following LPIPS [97], each sample was labeled by 3 participants, and the mean of the judgments was taken to decide which video had a higher quality. All participants that we recruited were volunteers from diverse backgrounds. For our training dataset, we randomly select 5,353 samples out of the 5,948 human-annotated video triplets and all 18,939 sample triplets with judgment via LPIPS. The rest 595 human-annotated samples were used for validation.
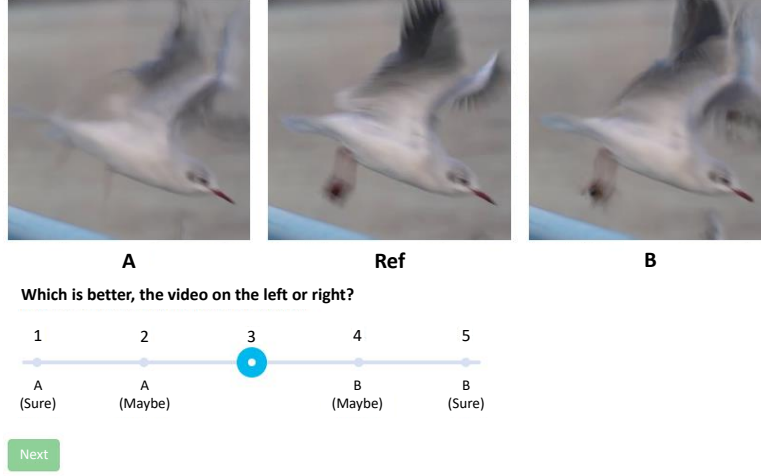
Fig. 3: User interface. Two frame interpolation results and one reference are played simultaneously in a loop.

## 4    Video Frame Interpolation Quality Metric

Our method takes a video $\mathbf{V} = \{\mathbf{I}_0, \mathbf{I}_1, \cdots, \mathbf{I}_N\}$ and its reference video $\mathbf{V}_R = \{\mathbf{I}_{R,0}, \mathbf{I}_{R,1}, \cdots, \mathbf{I}_{R,N}\}$ as inputs and estimates the perceptual similarity $d$ between these two videos, where $N$ is the number of frames.

Figure 4 shows the architecture of our network. Our network first extracts the feature maps for each image in the videos. For each feature map, we design a spatio-temporal (ST) module to capture the spatio-temporal information and estimate the perceptual similarity of features. Finally, we predict the perceptual similarity between the input video $\mathbf{V}$ and its reference $\mathbf{V}_R$ by averaging the similarity across all features. Our network is trained in a Siamese manner. Below we describe our network in more detail.

**Feature extraction.** As shown in Figure 4, we extract features from the video $\mathbf{V}$ and its reference video $\mathbf{V}_R$. Specifically, we build a pyramid network with five levels. Each level contains two $3 \times 3$ `Conv2D` layers. The stride of the second `Conv2D` layer is set to 2 to have a large valid receptive field. From the first to the fifth level, the numbers of channels are 16, 32, 64, 96, and 128. It can be represented as follows.

$$\{\mathbf{F}_{level,i}\}_{level=1}^{L} = f_{ext}(\mathbf{I}_i), \tag{1}$$

where $f_{ext}(\cdot)$ indicates the operation of the extraction network. $\mathbf{F}$ indicates the extracted features. $L$ indicates the number of feature levels. The feature extraction network shares weights for all frames and videos. For each level, we concatenate the features from all frames as the final features,

$$\mathbf{F}_{level} = f_{cat}(\mathbf{F}_{level,0}, \cdots, \mathbf{F}_{level,N}), \tag{2}$$
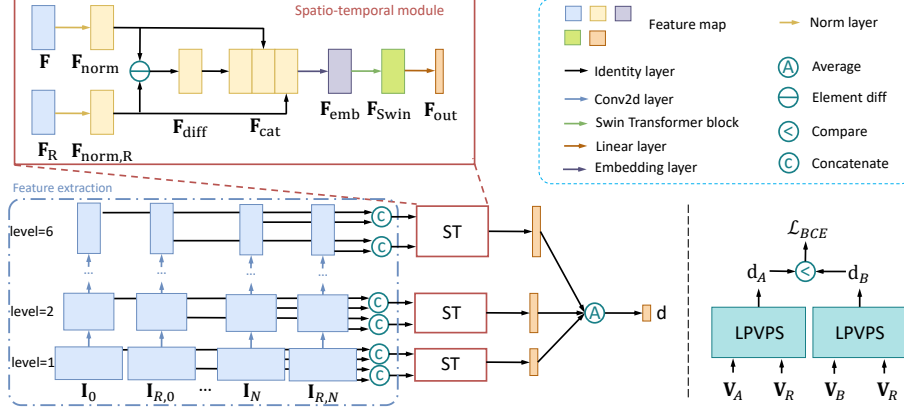
Fig. 4: Our network architecture. (Left) Learned Perceptual Video Patch Similarity (LPVPS) takes a video $\mathbf{V}$ and its reference $\mathbf{V}_R$ as input and predicts their perceptual similarity $d$. (Right) Our network is trained in a Siamese manner.

where $f_{cat}(\cdot)$ is the concatenation operation. $N$ is the number of frames.

**Spatio-temporal (ST) module** is designed to measure the perceptual distance between the feature maps $\mathbf{F}$ and its reference $\mathbf{F}_R$. It first estimates the difference between two normalized features as follows.

$$\mathbf{F}_{diff} = \|f_{norm}(\mathbf{F}) - f_{norm}(\mathbf{F}_R)\|_1, \tag{3}$$

where $f_{norm}(\cdot)$ indicates the operation of unit normalization in the channel dimension [97]. As reported in previous work on image perceptual metrics [26, 51, 52], the perceptual similarity is not only related to the difference between two images, but also the content of images. Therefore, our network leverages both the feature difference $\mathbf{F}_{diff}$ and the source features.

$$\mathbf{F}_{cat} = f_{cat}(\mathbf{F}_{diff}, f_{norm}(\mathbf{F}), f_{norm}(\mathbf{F}_R)). \tag{4}$$

As shown in Figure 4, we first adopt a linear embedding layer to project the features $\mathbf{F}_{cat}$ to a fixed dimension 32.

$$\mathbf{F}_{emb} = f_{emb}(\mathbf{F}_{cat}), \tag{5}$$

where $f_{emb}(\cdot)$ indicates the linear embedding layer that consists of a `Conv2D` layer with a $1 \times 1$ kernel.

We adopt a Swin Transformer block [47] to learn the temporal information across frames. Compared to the original design of Swin Transformer block [47], we do not use `Layer Norm (LN)` layers, which is critical to measure the difference between two videos. `LN` layers normalize the differences between two features and compromise the accuracy of the network. We empirically find that the dimension of the embedded features as 32, the head number as 2, and the window sizes of 4 can get good results and require relatively less computation and memory.

Table 1: Comparison with state-of-the-art methods.

| | Method | VFIPS (val.) 2AFC | BVI-VFI [19] (test) | | |
|---|---|---|---|---|---|
| | | | SROCC | PLCC | KROCC |
| Image | PSNR | 0.763 | 0.742 | 0.722 | 0.656 |
| | SSIM [79] | 0.784 | 0.739 | 0.746 | 0.639 |
| | MS-SSIM [80] | 0.794 | 0.772 | 0.789 | 0.667 |
| | LPIPS (VGG) [97] | 0.808 | 0.628 | 0.796 | 0.517 |
| | DISTS [21] | 0.801 | 0.597 | 0.763 | 0.517 |
| | PIM-1 [6] | 0.787 | 0.492 | 0.668 | 0.428 |
| | Watson-DFT [17] | 0.800 | 0.628 | 0.706 | 0.538 |
| Video | STRRED [7] | 0.777 | 0.614 | 0.682 | 0.539 |
| | VMAF [69] | 0.805 | 0.583 | 0.614 | 0.483 |
| | DeepVQA [77] | 0.588 | 0.369 | 0.271 | 0.300 |
| | VSFA [41] | 0.660 | 0.108 | 0.486 | 0.050 |
| | Ours | **0.830** | **0.794** | **0.870** | **0.700** |

ST modules are applied to the features at different levels, which can capture the spatio-temporal information at multiple scales. It is similar to the pyramid architectures of optical flow networks [45, 75]. The final distance is calculated by averaging the outputs at all the $L$ levels from ST modules.

$$d = \frac{1}{LN} \sum_{l=0}^{L} \sum_{n=0}^{N} \mathbf{F}_{Swin,l}^n \qquad (6)$$

where $N$ indicates the number of elements in $\mathbf{F}_{Swin,l}$.

**Loss function.** As shown in Figure 4, our network is trained in a Siamese manner. Given two videos with different distortions $\mathbf{V}_A$, $\mathbf{V}_B$, its reference video $\mathbf{V}_R$, and its judgement $\hat{h}$, we first predict the perceptual similarity $d_A$ and $d_B$ from $(\mathbf{V}_A, \mathbf{V}_R)$ and $(\mathbf{V}_B, \mathbf{V}_R)$, respectively. We calculate the probability $p$ with a sigmoid layer,

$$p = sigmoid(d_A - d_B). \qquad (7)$$

Then, we calculate the binary cross entropy loss $\mathcal{L}_{BCE}$ as:

$$\mathcal{L}_{BCE} = -(\hat{h} \log(p) + (1 - \hat{h}) \log(1 - p)). \qquad (8)$$

**Training.** We use PyTorch to train our neural network. Following [97], our learning rate is set to 0.0001. We use a mini-batch size of 8 and train the network for 20 epochs. We randomly resize the videos in the scale of [0.5, 1]. Our network is randomly initialized and uses AdamW [34] as the optimizer.

## 5   Experiments

We use the BVI-VFI [19] dataset as our test set. It contains 36 reference videos at 3 different frame rates: 30fps, 60fps, and 120fps. All the videos last 5 seconds.

Table 2: Comparison on the X-TEST(4K) dataset [73].

| Method | PSNR | SSIM [79] | MSSSIM [80] | LPIPS [97] | STRRED [7] | VMAF [69] | Ours |
|--------|------|-----------|-------------|------------|------------|-----------|------|
| 2AFC | 0.752 | 0.637 | 0.737 | 0.748 | 0.722 | 0.735 | **0.789** |

Each reference video has 5 distorted videos generated from different video frame interpolation algorithms, including frame repeating, frame averaging, DVF [48], QVI [93] and ST-MFNet [18]. Please note that video frame interpolation methods used in the BVI-VFI dataset are different from the ones used in our VFIPS dataset. To test our metric on the BVI-VFI dataset, we evaluate our metric in a sliding-window manner and take the average score. We selected Spearman Rank Order Correlation Coefficient (SROCC), Pearson Linear Correlation Coefficient (PLCC), and Kendall Rank Order Correlation Coefficient (KROCC) scores as our evaluation metrics. We calculated the scores for each reference video and reported the mean score as the final result. We also report our results on our VFIPS validation set. Following LPIPS [97], we report the Two Alternative Forced Choice (2AFC) scores [97]. We compare our method to the representative state-of-the-art metrics and conduct ablation studies to evaluate our metric.

### 5.1   Comparisons to Existing Metrics

We compare our method to both the state-of-the-art image perceptual metrics, including PSNR, SSIM [79], MS-SSIM [80], LPIPS(VGG) [97], DISTS [21], PIM-1 [6], and Watson-DFT [17], and the state-of-the-art video perceptual metrics, including VMAF [69], DeepVQA [32], and STRRED [74]. We also compare with the Non-Reference metrics, VSFA [41]. For each method, we obtained the results by using the official codes/models provided by their authors, except for DeepVQA which we adopt its re-implementation from Tencent [77]. For the image perceptual similarity metrics, we evaluate the score per frame and take the average as the final score for that video sequence.

   As shown in Table 1, our method outperforms the state-of-the-art methods by a considerable margin on both the VFIPS dataset and the BVI-VFI dataset. Especially when compared to the image-based similarity methods such as LPIPS, our method provides significant gains in terms of 2AFC on the VFIPS dataset and SROCC, PLCC, and KROCC on the BVI-VFI dataset. We attribute this improvement to the spatial-temporal module that captures spatio-temporal information. Compared to the video-based methods, our method again achieves significant performance gains on all metrics on both datasets, specifically, 0.025 in terms of 2AFC on the VFIPS validation set and 0.180 in SROCC, 0.188 in PLCC, 0.161 in KROCC on the BVI-VFI dataset. Figure 5 shows the visual examples on the VFIPS validation set. The predictions of our method are more consistent with humans. In the first example, Video 2 suffers less distortions to the person's head. In the second example, Video 2 is temporally consistent. In the third example, the hand in Video 2 is distorted more. As indicated, our predictions are consistent with humans in these examples. Figure 6 shows the visual
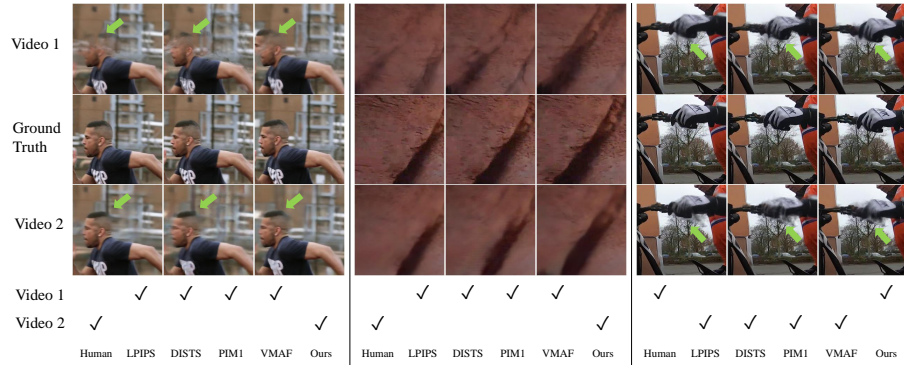
Fig. 5: Visual examples on the VFIPS dataset. Green arrows are used to label the area with noticeable difference. We mark the preference of each method using "✓". Compared to other methods, our method is consistent with humans.
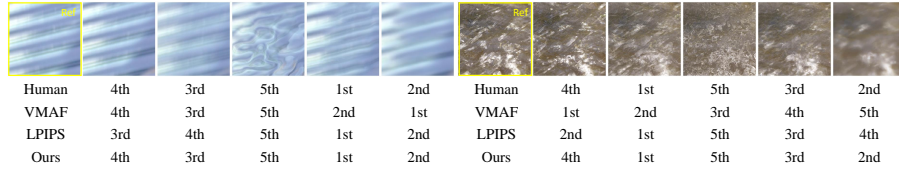


Fig. 6: Visual examples on the BVI-VFI dataset [19]. Yellow rectangles are used to show the reference video. We report the rank for the distorted videos. Compared to other metrics, our metric is more consistent with humans.

examples from the BVI-VFI dataset [19]. For each reference video, it shows the rank for the distorted videos. Our predictions are more consistent with humans.

We also conduct our experiments on the X-TEST dataset [73], which has 15 testing videos with 4K resolution. We use SepConv, Super-Slomo, AdaCof, and XVFI to generate VFI results. Following Sim *et al.* [73], we set the temporal distance of 32 frames for the large motion. For each video, we interpolate 32 frames and take the middle 12 frames for testing. There are 90 pairs in total. Each pair is annotated by 3 users. As reported in Table 2, our method outperforms other methods on such a large-motion dataset.

## 5.2   Ablation Studies

**Feature extraction network.** We investigate the impact of architectures of the feature extraction network in Table 3. Following LPIPS [97], we replace our extraction network with several classic architectures, including AlexNet [37], I3D [10], and ours-3D network. We use the official implementation of AlexNet and I3D [10]. For ours-3D, we replace the `Conv2D` layers in our feature extraction module with `Conv3D` layers with $3 \times 3 \times 3$ kernels. We measure the runtime for a 12-frame $256 \times 256$ video on a single Nvidia RTX A5000 GPU.

Table 3: Effectiveness of different feature extraction network on the BVI-VFI datasets [19].

| Extractor | SROCC | PLCC | KROCC | Param(M) | Runtime(ms) |
|-----------|-------|------|-------|----------|-------------|
| AlexNet [37] | 0.761 | 0.832 | 0.650 | 14.5 | 12.8 |
| I3D [10] | 0.659 | 0.758 | 0.550 | 20.3 | 33.2 |
| Ours-3D | 0.728 | 0.738 | 0.639 | 8.6 | 13.6 |
| Ours-2D | **0.794** | **0.870** | **0.700** | **4.6** | **10.4** |

As shown in Table 3, our method not only achieves the best performance but also needs the least parameters. Compared to AlexNet, our feature extraction network achieves better performance while our method only needs 34.1% parameters and is 1.23× faster. Compared to I3D and ours-3D, our network achieves much better results and has less parameters. We attribute the improvement to that our feature extraction network keeps more temporal information. Specifically, I3D and ours-3D downsample the features temporally, while our network keeps features of all the frames. The temporal compression in the feature extraction network might hurt the performance of the whole metric.

**Spatio-temporal module.** We study the spatio-temporal module in Table 4. We use different blocks for the spatio-temporal module. "Conv3D" indicates the Conv3D layers with $12 \times 3 \times 3$ kernels. "Original Swin" indicates the official Swin Transformer blocks. "Ours-Swin w. LN" indicates our Swin Transformer block with LN layers. Compared to the original Swin, it contains fewer parameters, including 32 vs. 96 (channels), 4 vs. 7 (window size). It has slightly worse performance. As shown in Table 4, original Swin and ours-Swin w. LN do not perform as well as Conv3D. We attribute this performance loss to the LN normalizing the differences between features. Removing the LN layers improves the results by 0.070 on SROCC, 0.124 on PLCC, and 0.089 on KROCC on the BVI-VFI dataset.

Table 4: Effectiveness of the spatio-temporal module on the BVI-VFI dataset [19].

| ST Module | SROCC | PLCC | KROCC |
|-----------|-------|------|-------|
| None | 0.617 | 0.663 | 0.539 |
| Conv3D | 0.761 | 0.819 | 0.661 |
| Original Swin | 0.728 | 0.766 | 0.639 |
| Ours-Swin w. LN | 0.724 | 0.746 | 0.611 |
| Ours-Swin | **0.794** | **0.870** | **0.700** |

**Automatic annotations by LPIPS.** We study the impact of annotations in Table 5. As discussed in Section 3.2, our training set contains two parts: samples obtained through automatic annotation or human label-

Table 5: Effectiveness of annotations on the BVI-VFI dataset [19]

| Annations | SROCC | PLCC | KROCC |
|-----------|-------|------|-------|
| Human | 0.719 | 0.753 | 0.611 |
| Automatic | 0.653 | 0.687 | 0.567 |
| All | **0.794** | **0.870** | **0.700** |

ing. We train our network with different training subsets: human annotations, automatic annotations, and a combination of both. As shown in Table 5, the combined set achieves the best performance. The automatic training data can increase the diversity of the training set and improve the generalization capability of the metric.

## 6   Limitations and Future Work

Many image-based quality assessment metrics can generate an error map between a distorted image and its reference to analyze altered pixels. However, our metric cannot produce such an error map for individual frames in the video since our spatio-temporal module contains convolutional layers that fuse features from these frames at an early stage in the inference. In the future, it will be interesting to design a network that can generate such error maps for each frame to provide more information on the distortion within and across the frames.

Recently, image perceptual quality metrics, such as LPIPS [97], are used as loss to create visually pleasant images or videos [63]. In theory, our network can also be used as a loss function to train a video frame interpolation network to produce visually pleasant videos. However, our metric has a high requirement for GPU memory. It takes about 2500 Mb memory to process one $256 \times 256$ 12-frame image sequence, which might become a bottleneck in training a video frame interpolation network. In the future, we aim to reduce the network size and study how our video frame interpolation quality metric will help optimize video frame interpolation networks as a learned perceptual loss.

## 7   Conclusion

This paper presented a video perceptual quality metric for video frame interpolation. Our metric first extracts pyramid features for individual frames in the videos. Then it compares features at each level using a spatio-temporal module to capture the spatio-temporal information. The spatio-temporal module is composed of the Swin Transformer blocks to capture the spatio-temporal information. We also collected a dataset for video frame interpolation. Our annotations were from humans and a widely adopted image perceptual quality metric. Our experiments showed that our dedicated video quality metric outperforms existing metrics for assessing video frame interpolation results.

# References

1. Antkowiak, J., Jamal Baina, T., Baroncini, F.V., Chateau, N., FranceTelecom, F., Pessoa, A.C.F., Stephanie Colonnese, F., Contin, I.L., Caviedes, J., Philips, F.: Final report from the video quality experts group on the validation of objective models of video quality assessment (2000)
2. Baker, S., Scharstein, D., Lewis, J., Roth, S., Black, M.J., Szeliski, R.: A database and evaluation methodology for optical flow. International journal of computer vision **92**(1), 1–31 (2011)
3. Bampis, C.G., Li, Z., Katsavounidis, I., Huang, T.Y., Ekanadham, C., Bovik, A.C.: Towards perceptually optimized end-to-end adaptive video streaming. arXiv preprint arXiv:1808.03898 (2018)
4. Bao, W., Lai, W.S., Ma, C., Zhang, X., Gao, Z., Yang, M.H.: Depth-aware video frame interpolation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3703–3712 (2019)
5. Bao, W., Lai, W.S., Zhang, X., Gao, Z., Yang, M.H.: Memc-net: Motion estimation and motion compensation driven neural network for video interpolation and enhancement. IEEE Transactions on Pattern Analysis and machine Intelligence (2019)
6. Bhardwaj, S., Fischer, I., Ballé, J., Chinen, T.: An unsupervised information-theoretic perceptual quality metric. In: Advances in Neural Information Processing Systems 33 (2020)
7. Bovik, A., Soundararajan, R., Bampis, C.: On the robust performance of the st-rred video quality predictor (2017)
8. van den Branden Lambrecht, C.J.: Color moving pictures quality metric. In: Proceedings of 3rd IEEE International Conference on Image Processing. vol. 1, pp. 885–888. IEEE (1996)
9. Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: End-to-end object detection with transformers. In: European Conference on Computer Vision. pp. 213–229. Springer (2020)
10. Carreira, J., Zisserman, A.: Quo vadis, action recognition? a new model and the kinetics dataset. In: proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 6299–6308 (2017)
11. Chen, H., Wang, Y., Guo, T., Xu, C., Deng, Y., Liu, Z., Ma, S., Xu, C., Xu, C., Gao, W.: Pre-trained image processing transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 12299–12310 (2021)
12. Chen, L.H., Bampis, C.G., Li, Z., Bovik, A.C.: Learning to distort images using generative adversarial networks. IEEE Signal Processing Letters **27**, 2144–2148 (2020)
13. Cheng, X., Chen, Z.: Multiple video frame interpolation via enhanced deformable separable convolution. IEEE Transactions on Pattern Analysis and Machine Intelligence (2021)
14. Cheon, M., Yoon, S.J., Kang, B., Lee, J.: Perceptual image quality assessment with transformers. 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW) pp. 433–442 (2021)
15. Choi, M., Kim, H., Han, B., Xu, N., Lee, K.M.: Channel attention is all you need for video frame interpolation. In: Proceedings of the AAAI Conference on Artificial Intelligence. pp. 10663–10671 (2020)
16. Choi, M., Lee, S., Kim, H., Lee, K.M.: Motion-aware dynamic architecture for efficient frame interpolation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 13839–13848 (2021)

17. Czolbe, S., Krause, O., Cox, I., Igel, C.: A loss function for generative neural networks based on watson's perceptual model. In: Advances in Neural Information Processing Systems. pp. 2051–2061 (2020)
18. Danier, D., Zhang, F., Bull, D.: St-mfnet: A spatio-temporal multi-flow network for frame interpolation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3521–3531 (2022)
19. Danier, D., Zhang, F., Bull, D.: A subjective quality study for video frame interpolation. arXiv preprint arXiv:2202.07727 (2022)
20. Dendi, S.V.R., Krishnappa, G., Channappayya, S.S.: Full-reference video quality assessment using deep 3d convolutional neural networks. In: 2019 National Conference on Communications (NCC). pp. 1–5. IEEE (2019)
21. Ding, K., Ma, K., Wang, S., Simoncelli, E.P.: Image quality assessment: Unifying structure and texture similarity. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–1 (2020)
22. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale. In: 9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021. OpenReview.net (2021), `https://openreview.net/forum?id=YicbFdNTTy`
23. Gardiner, P., Ghanbari, M., Pearson, D., Tan, K.: Development of a perceptual distortion meter for digital video. In: 1997 International Broadcasting Convention IBS 97. pp. 493–497. IET (1997)
24. Ghadiyaram, D., Pan, J., Bovik, A., Moorthy, A., Panda, P., Yang, K.: Live-qualcomm mobile in-capture video quality database (2017)
25. Ghadiyaram, D., Pan, J., Bovik, A.C.: A subjective and objective study of stalling events in mobile streaming videos. IEEE Transactions on Circuits and Systems for Video Technology **29**(1), 183–197 (2017)
26. Goodman, N.: Seven strictures on similarity (1972)
27. Hamada, T., Miyaji, S., Matsumoto, S.: Picture quality assessment system by three-layered bottom-up noise weighting considering human visual perception. SMPTE journal **108**(1), 20–26 (1999)
28. Hu, P., Niklaus, S., Sclaroff, S., Saenko, K.: Many-to-many splatting for efficient video frame interpolation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3553–3562 (2022)
29. Huang, Z., Zhang, T., Heng, W., Shi, B., Zhou, S.: Rife: Real-time intermediate flow estimation for video frame interpolation. arXiv preprint arXiv:2011.06294 (2020)
30. Jiang, H., Sun, D., Jampani, V., Yang, M.H., Learned-Miller, E., Kautz, J.: Super slomo: High quality estimation of multiple intermediate frames for video interpolation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 9000–9008 (2018)
31. Kalluri, T., Pathak, D., Chandraker, M., Tran, D.: Flavr: Flow-agnostic video representations for fast frame interpolation. arXiv preprint arXiv:2012.08512 (2020)
32. Kim, J., Lee, S.: Deep learning of human visual sensitivity in image quality assessment framework. In: Proceedings of the IEEE conference on Computer Vision and Pattern Recognition. pp. 1676–1684 (2017)
33. Kim, W., Kim, J., Ahn, S., Kim, J., Lee, S.: Deep video quality assessor: From spatio-temporal visual sensitivity to a convolutional neural aggregation network. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 219–234 (2018)

34. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
35. Korhonen, J.: Two-level approach for no-reference consumer video quality assessment. IEEE Transactions on Image Processing **28**(12), 5923–5938 (2019)
36. Korhonen, J., Su, Y., You, J.: Blind natural video quality prediction via statistical temporal features and deep spatial features. In: Proceedings of the 28th ACM International Conference on Multimedia. pp. 3311–3319 (2020)
37. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. Advances in neural information processing systems **25**, 1097–1105 (2012)
38. Kuroki, Y., Nishi, T., Kobayashi, S., Oyaizu, H., Yoshimura, S.: A psychophysical study of improvements in motion-image quality by using high frame rates. Journal of the Society for Information Display **15**(1), 61–68 (2007)
39. Kuroki, Y., Takahashi, H., Kusakabe, M., Yamakoshi, K.i.: Effects of motion image stimuli with normal and high frame rates on eeg power spectra: comparison with continuous motion image stimuli. Journal of the Society for Information Display **22**(4), 191–198 (2014)
40. Lee, H., Kim, T., Chung, T.y., Pak, D., Ban, Y., Lee, S.: Adacof: Adaptive collaboration of flows for video frame interpolation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5316–5325 (2020)
41. Li, D., Jiang, T., Jiang, M.: Quality assessment of in-the-wild videos. In: Proceedings of the 27th ACM International Conference on Multimedia. pp. 2351–2359 (2019)
42. Li, H., Yuan, Y., Wang, Q.: Video frame interpolation via residue refinement. In: IEEE International Conference on Acoustics, Speech and Signal Processing. pp. 2613–2617. IEEE (2020)
43. Li, Y., Zhang, K., Cao, J., Timofte, R., Van Gool, L.: Localvit: Bringing locality to vision transformers. arXiv preprint arXiv:2104.05707 (2021)
44. Li, Z., Aaron, A., Katsavounidis, I., Moorthy, A., Manohara, M.: Toward a practical perceptual video quality metric. The Netflix Tech Blog **6**(2) (2016)
45. Liu, P., Lyu, M., King, I., Xu, J.: Selflow: Self-supervised learning of optical flow. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4571–4580 (2019)
46. Liu, Y.L., Liao, Y.T., Lin, Y.Y., Chuang, Y.Y.: Deep video frame interpolation using cyclic frame generation. In: Proceedings of the AAAI Conference on Artificial Intelligence. vol. 33, pp. 8794–8802 (2019)
47. Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., Guo, B.: Swin transformer: Hierarchical vision transformer using shifted windows. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 10012–10022 (2021)
48. Liu, Z., Yeh, R.A., Tang, X., Liu, Y., Agarwala, A.: Video frame synthesis using deep voxel flow. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 4463–4471 (2017)
49. Long, G., Kneip, L., Alvarez, J.M., Li, H., Zhang, X., Yu, Q.: Learning image matching by simply watching video. In: European Conference on Computer Vision. pp. 434–450. Springer (2016)
50. Lubin, J.: A human vision system model for objective picture quality measurements. In: IEE conference publication. vol. 1, pp. 498–503. IET (1997)
51. Markman, A.B., Gentner, D.: Nonintentional similarity processing. The new unconscious pp. 107–137 (2005)

52. Medin, D.L., Goldstone, R.L., Gentner, D.: Respects for similarity. Psychological review **100**(2), 254 (1993)
53. Meyer, S., Cornillère, V., Djelouah, A., Schroers, C., Gross, M.: Deep video color propagation. In: Proceedings of the British Machine Vision Conference (BMVC) (2018)
54. Meyer, S., Djelouah, A., McWilliams, B., Sorkine-Hornung, A., Gross, M., Schroers, C.: Phasenet for video frame interpolation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 498–507 (2018)
55. Meyer, S., Wang, O., Zimmer, H., Grosse, M., Sorkine-Hornung, A.: Phase-based frame interpolation for video. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1410–1418 (2015)
56. Min, X., Zhai, G., Zhou, J., Farias, M.C., Bovik, A.C.: Study of subjective and objective quality assessment of audio-visual signals. IEEE Transactions on Image Processing **29**, 6054–6068 (2020)
57. Niklaus, S., Liu, F.: Context-aware synthesis for video frame interpolation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 1701–1710 (2018)
58. Niklaus, S., Liu, F.: Softmax splatting for video frame interpolation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 5437–5446 (2020)
59. Niklaus, S., Mai, L., Liu, F.: Video frame interpolation via adaptive convolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 670–679 (2017)
60. Niklaus, S., Mai, L., Liu, F.: Video frame interpolation via adaptive separable convolution. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 261–270 (2017)
61. Niklaus, S., Mai, L., Wang, O.: Revisiting adaptive convolutions for video frame interpolation. arXiv preprint arXiv:2011.01280 (2020)
62. Niklaus, S., Mai, L., Wang, O.: Revisiting adaptive convolutions for video frame interpolation. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 1099–1109 (2021)
63. Niklaus, S., Zhang, X.C., Barron, J.T., Wadhwa, N., Garg, R., Liu, F., Xue, T.: Learned dual-view reflection removal. In: Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision. pp. 3713–3722 (2021)
64. Park, J., Ko, K., Lee, C., Kim, C.S.: Bmbc: Bilateral motion estimation with bilateral cost volume for video interpolation. arXiv preprint arXiv:2007.12622 (2020)
65. Park, J., Lee, C., Kim, C.S.: Asymmetric bilateral motion estimation for video frame interpolation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14539–14548 (2021)
66. Pinson, M.H., Wolf, S.: A new standardized method for objectively measuring video quality. IEEE Transactions on broadcasting **50**(3), 312–322 (2004)
67. Prashnani, E., Cai, H., Mostofi, Y., Sen, P.: Pieapp: Perceptual image-error assessment through pairwise preference. In: IEEE Conference on Computer Vision and Pattern Recognition. pp. 1808–1817 (2018)
68. Rakêt, L.L., Roholm, L., Bruhn, A., Weickert, J.: Motion compensated frame interpolation with a symmetric optical flow constraint. In: International Symposium on Visual Computing. pp. 447–457. Springer (2012)
69. Rassool, R.: Vmaf reproducibility: Validating a perceptual practical video quality metric. In: 2017 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB). pp. 1–2 (2017). https://doi.org/10.1109/BMSB.2017.7986143

70. Reda, F.A., Sun, D., Dundar, A., Shoeybi, M., Liu, G., Shih, K.J., Tao, A., Kautz, J., Catanzaro, B.: Unsupervised video interpolation using cycle consistency. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 892–900 (2019)
71. Seshadrinathan, K., Soundararajan, R., Bovik, A.C., Cormack, L.K.: Study of subjective and objective quality assessment of video. IEEE Transactions on Image Processing **19**(6), 1427–1441 (2010)
72. Shi, Z., Xu, X., Liu, X., Chen, J., Yang, M.H.: Video frame interpolation transformer. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 17482–17491 (2022)
73. Sim, H., Oh, J., Kim, M.: Xvfi: Extreme video frame interpolation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14489–14498 (2021)
74. Soundararajan, R., Bovik, A.C.: Video quality assessment by reduced reference spatio-temporal entropic differencing. IEEE Transactions on Circuits and Systems for Video Technology **23**(4), 684–694 (2013)
75. Sun, D., Yang, X., Liu, M.Y., Kautz, J.: Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 8934–8943 (2018)
76. Talebi, H., Milanfar, P.: Nima: Neural image assessment. IEEE Transactions on Image Processing **27**(8), 3998–4011 (2018)
77. Tencent: Deep learning-based Video Quality Assessment. https://github.com/Tencent/DVQA (2022), [Online; accessed 19-July-2022]
78. Tu, Z., Wang, Y., Birkbeck, N., Adsumilli, B., Bovik, A.C.: Ugc-vqa: Benchmarking blind video quality assessment for user generated content. IEEE Transactions on Image Processing **30**, 4449–4464 (2021)
79. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004)
80. Wang, Z., Simoncelli, E.P., Bovik, A.C.: Multiscale structural similarity for image quality assessment. In: The Thrity-Seventh Asilomar Conference on Signals, Systems & Computers, 2003. vol. 2, pp. 1398–1402. Ieee (2003)
81. Watson, A.B.: Toward a perceptual video-quality metric. In: Human Vision and Electronic Imaging III. vol. 3299, pp. 139–147. International Society for Optics and Photonics (1998)
82. Watson, A.B., Hu, Q.J., McGowan III, J.F.: Digital video quality metric based on human vision. Journal of Electronic imaging **10**(1), 20–29 (2001)
83. Webster, A.A., Jones, C.T., Pinson, M.H., Voran, S.D., Wolf, S.: Objective video quality assessment system based on human perception. In: Human vision, visual processing, and digital display IV. vol. 1913, pp. 15–26. International Society for Optics and Photonics (1993)
84. Winkler, S.: Digital video quality: vision models and metrics. John Wiley & Sons (2005)
85. Winkler, S., Mohandas, P.: The evolution of video quality measurement: From psnr to hybrid metrics. IEEE transactions on Broadcasting **54**(3), 660–668 (2008)
86. Wolf, S., Pinson, M.H., Webster, A.A., Cermak, G.W., Tweedy, E.P.: Objective and subjective measures of mpeg video quality. Society of Motion Picture and Television Engineers **160**,  178 (1997)
87. Wu, C.Y., Singhal, N., Krahenbuhl, P.: Video compression through image interpolation. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 416–431 (2018)

88. Wu, H.R., Rao, K.R.: Digital video image quality and perceptual coding. CRC press (2017)
89. Wulff, J., Black, M.J.: Temporal interpolation as an unsupervised pretraining task for optical flow estimation. In: German Conference on Pattern Recognition. pp. 567–582. Springer (2018)
90. Xiao, F., et al.: Dct-based video quality evaluation. Final Project for EE392J **769** (2000)
91. Xu, M., Chen, J., Wang, H., Liu, S., Li, G., Bai, Z.: C3dvqa: Full-reference video quality assessment with 3d convolutional neural network. In: ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). pp. 4447–4451. IEEE (2020)
92. Xu, X., Li, M., Sun, W.: Learning deformable kernels for image and video denoising. arXiv preprint arXiv:1904.06903 (2019)
93. Xu, X., Siyao, L., Sun, W., Yin, Q., Yang, M.H.: Quadratic video interpolation. Advances in Neural Information Processing Systems **32** (2019)
94. Xue, T., Chen, B., Wu, J., Wei, D., Freeman, W.T.: Video enhancement with task-oriented flow. International Journal of Computer Vision **127**(8), 1106–1125 (2019)
95. You, J., Korhonen, J.: Transformer for image quality assessment. In: 2021 IEEE International Conference on Image Processing (ICIP). pp. 1389–1393. IEEE (2021)
96. Yu, Z., Zhang, Y., Liu, D., Zou, D., Chen, X., Liu, Y., Ren, J.S.: Training weakly supervised video frame interpolation with events. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 14589–14598 (2021)
97. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 586–595 (2018)