STDAN: Deformable Attention Network for Space-Time Video Super-Resolution

Hai Wang, *Member, IEEE*, Xiaoyu Xiang, Yapeng Tian, *Member, IEEE*, Wenming Yang, *Senior Member, IEEE*, and Qingmin Liao, *Senior Member, IEEE*

***Abstract*— The target of space–time video super-resolution (STVSR) is to increase the spatial–temporal resolution of low-resolution (LR) and low-frame-rate (LFR) videos. Recent approaches based on deep learning have made significant improvements, but most of them only use two adjacent frames, that is, short-term features, to synthesize the missing frame embedding, which cannot fully explore the information flow of consecutive input LR frames. In addition, existing STVSR models hardly exploit the temporal contexts explicitly to assist high-resolution (HR) frame reconstruction. To address these issues, in this article, we propose a deformable attention network called STDAN for STVSR. First, we devise a long short-term feature interpolation (LSTFI) module that is capable of excavating abundant content from more neighboring input frames for the interpolation process through a bidirectional recurrent neural network (RNN) structure. Second, we put forward a spatial–temporal deformable feature aggregation (STDFA) module, in which spatial and temporal contexts in dynamic video frames are adaptively captured and aggregated to enhance SR reconstruction. Experimental results on several datasets demonstrate that our approach outperforms state-of-the-art STVSR methods. The code is available at https://github.com/littlewhitesea/STDAN.**

***Index Terms*— Deformable attention, feature aggregation, feature interpolation, space–time video super-resolution (STVSR).**

I. Introduction

THE goal of space–time video super-resolution (STVSR) is to reconstruct photorealistic high-resolution (HR) and high-frame-rate (HFR) videos from corresponding low-resolution (LR) and low-frame-rate (LFR) ones. STVSR methods have attracted much attention in the computer vision community since HR slow-motion videos provide more visually appealing content for viewers. Many traditional algorithms [2], [3], [4], [8], [19] are proposed to solve the STVSR task. However, due to their strict assumptions in their manually designed regularization, these methods mostly suffer from the ubiquitous object and camera motions in videos.

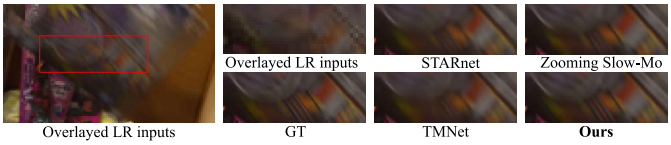


Fig. 1. Example of STVSR. Compared with three recent SOTA STVSR methods, our network can reconstruct more accurate structures.

In recent years, deep learning approaches have made great progress in diverse visual tasks [9], [20], [26], [33], [34], [35], [50], [57], [66]. Particularly, video super-resolution (VSR) [42], [57] and video frame interpolation (VFI) [25], [35] networks among these approaches can be combined together to tackle STVSR. Specifically, the VFI model interpolates the missing LR video frames. Then, the VSR model can be adapted to reconstruct HR frames. Nevertheless, the two-stage STVSR approaches usually have large model sizes, and the essential association between temporal interpolation and spatial super-resolution is not explored.

To build an efficient model and explore mutual information between temporal interpolation and spatial super-resolution, several one-stage STVSR networks [13], [40], [67], [68] are proposed. These approaches can simultaneously handle the space and time super-resolution of videos in diverse scenes. Most of them only leverage corresponding two adjacent frames for interpolating the missing frame feature. However, other neighboring input LR frames can also contribute to the interpolation process. In addition, existing one-stage STVSR networks are limited in fully exploiting spatial and temporal contexts among various frames for SR reconstruction. To alleviate these problems, in this article, we propose a one-stage framework named STDAN for STVSR, which is superior to recent methods, as illustrated in Fig. 1. The cores of STDAN are: 1) a feature interpolation module known as long-short term feature interpolation (LSTFI) and 2) a feature aggregation module known as spatial–temporal deformable feature aggregation (STDFA).

The LSTFI module, composed of long-short term cells (LSTCs), utilizes a bidirectional recurrent neural network

Manuscript received 14 July 2022; revised 7 December 2022; accepted 31 January 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62171251, in part by the Natural Science Foundation of Guangdong Province under Grant 2020A1515010711, in part by the Special Foundations for the Development of Strategic Emerging Industries of Shenzhen under Grant JCYJ20200109143010272 and Grant CJGJZD20210408092804011, and in part by the Oversea Cooperation Foundation of Tsinghua. *(Corresponding author: Wenming Yang.)*

Hai Wang, Wenming Yang, and Qingmin Liao are with the Tsinghua Shenzhen International Graduate School and the Department of Electronic Engineering, Tsinghua University, Shenzhen 518055, China (e-mail: hwangshenzhen@163.com; yangelwm@163.com; liaoqm@tsinghua.edu.cn).

Xiaoyu Xiang is with the Core AI Team, Meta Reality Labs, Menlo Park, CA 94025 USA (e-mail: xiaoyu.xiang.ai@gmail.com).

Yapeng Tian is with the Department of Computer Science, The University of Texas at Dallas, Richardson, TX 75080 USA (e-mail: yapeng.tian@utdallas.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2023.3243029.

Digital Object Identifier 10.1109/TNNLS.2023.3243029

(RNN) [41] structure to synthesize features for missing intermediate frames. Specifically, to interpolate the intermediate feature, we adopt the forward and backward deformable alignment [13] for dynamically sampling two neighboring frame features. Then, the preliminary intermediate feature in the current LSTC is mingled with the hidden state that contains long-term temporal context from previous LSTCs to obtain the final interpolated features.

The STDFA module aims to capture spatial–temporal contexts among different frames to enhance SR reconstruction. To dynamically aggregate the spatial–temporal information, we propose to use deformable attention to adaptively discover and leverage relevant spatial and temporal information. The process of STDFA can be divided into two phases: cross-frame spatial aggregation and adaptive temporal aggregation. Through deformable attention, the cross-frame spatial aggregation phase dynamically fuses useful content from different frames. The adaptive temporal aggregation phase mixes the temporal contexts among these fused frame features further to acquire enhanced features.

The contributions of this work are threefold.

1) We design a deformable attention network (STDAN) to deal with STVSR. Our STDAN with fewer parameters achieves state-of-the-art (SOTA) performance on multiple datasets.
2) We propose an LSTFI module, where abundant information from more neighboring frames is explored for the interpolation process of missing frame features.
3) We put forward an STDFA module, which can dynamically capture spatial and temporal contexts among video frames for enhancing features to reconstruct HR frames.

II. Related Work

In this section, we discuss some relevant works on VSR, VFI, and STVSR.

A. Video Super-Resolution

The goal of VSR [42], [48], [52], [59] is to generate temporally coherent HR videos from corresponding LR ones. Since input LR video frames are consecutive, many researchers focus on how to aggregate the temporal contexts from the neighboring frames for super-resolving the reference frame. Several VSR approaches [23], [24], [36], [55], [56] adopt optical flow to align the reference frame with neighboring video frames. Nevertheless, the estimated optical flow may be inaccurate due to the occlusion and fast motions, leading to poor reconstruction results. To avoid using optical flow, deformable convolution [43], [44] is applied in [42], [57], and [58] to perform the temporal alignment in a feature space. Combining the advantages of optical flow and deformable convolution, Lin et al. [28] further put forward a flow-guided deformable alignment mechanism to capture temporal contexts between video frames. In addition, Li et al. [47] established a multicorrespondence aggregation network to exploit similar patches between and within frames. Dynamic filters [53] and nonlocal [46], [51] modules are also exploited to aggregate the temporal information. Very recently, Fuoli et al. [29] and Liang et al. [30] explored deformable attention to aggregate temporal information for VSR, which achieved good performance.

B. Video Frame Interpolation

VFI [27], [31], [32], [35], [66] aims to synthesize the missing intermediate frame with two adjacent video frames, which is extensively used in slow-motion video generation. Specifically, for generating the intermediate frame, U-Net structure modules [25] are employed to compute optical flows and visibility maps between two input frames. To cope with occlusion in VFI, contextual features [21] are further introduced into the interpolation process. Furthermore, Bao et al. [35] proposed a depth-aware module to detect occlusions explicitly for VFI. On the other hand, unlike most VFI methods using optical flow, Niklaus et al. [22], [33] adopted the adaptive convolution to predict kernels directly and then leveraged these kernels to estimate pixels of the intermediate video frame. Recently, attention mechanism [35] and deformable convolution [31], [61] are explored.

C. Space–Time Video Super-Resolution

Compared to VSR, STVSR needs to implement super-resolution in time and space dimensions, which is a more ill-posed problem. Thanks to the powerful nonlinear modeling capability, deep neural network models [13], [40], [67], [68], [69] have made significant advances on STVSR tasks. Specifically, through merging VSR and VFI into a joint framework, Kang et al. [67] put forward a DNN model for STVSR. To exploit mutually informative relationships between time and space dimensions, STARnet [69] with an extra optical flow branch is proposed to generate HR slow-motion videos. In addition, Xiang et al. [13] developed a deformable ConvLSTM [62] module, which can achieve sequence-to-sequence (S2S) learning in STVSR. Based on [13], Xu et al. [40] proposed a temporal modulation block to perform controllable STVSR. Recently, Geng et al. [68] proposed an STVSR network based on the Swin Transformer. However, most of them only leverage two adjacent frame features to interpolate the intermediate frame feature, and they hardly explore spatial and temporal contexts explicitly among video frames. To address these problems, we propose a spatial–temporal deformable network to: 1) use more content from input LR frames for the interpolation process and 2) employ deformable attention to dynamically capture spatial–temporal contexts for HR frame reconstruction.

III. Our Method

The architecture of our proposed network is illustrated in Fig. 2, which consists of four parts: the feature extraction module, the LSTFI module, the STDFA module, and the frame feature reconstruction module. Given an LR and LFR video with N frames, $\{I^{\mathrm{lr}}_{2t-1}\}^{N}_{t=1}$, our STDAN can generate $2N-1$ consecutive HR and high frame rate (HFR) frames: $\{I^{\mathrm{hr}}_{t}\}^{2N-1}_{t=1}$. The structure of each module is described in the following.

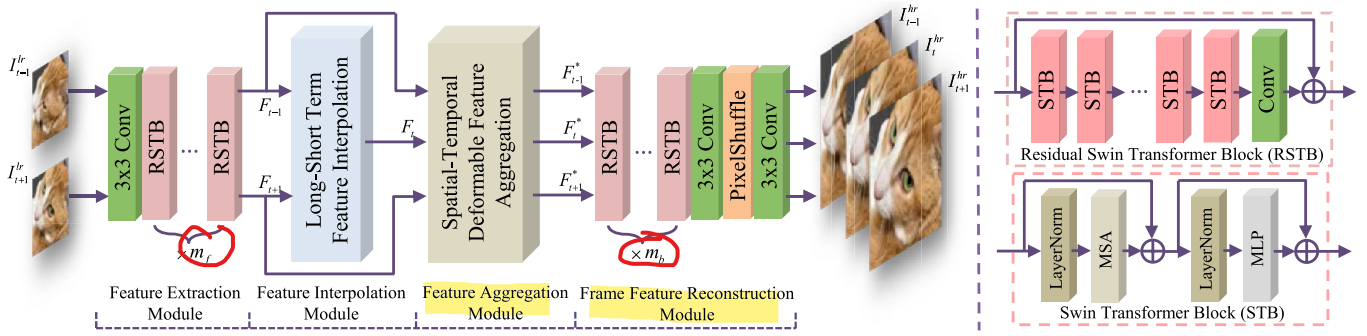


Fig. 2. Architecture of our proposed STDAN. LSTFI is capable of exploring more neighboring LR frames to synthesize the intermediate frame in the feature space. STDFA is utilized to capture spatial–temporal contexts by deformable attention. This figure only shows two input LR video frames from a long video sequence for a presentation.

A. Frame Feature Extraction

We first use a 3×3 convolutional layer in the feature extraction module to get shallow features $\{F^s_{2t-1}\}^N_{t=1}$ from the N input LR video frames. Considering that these shallow features lack long-range spatial information due to the locality of the naive convolutional layer, which may cause poor quality in the next feature interpolation module, we hope to extract these shallow features further to establish the correlation between two distant locations.

Recently, Transformer-based models have realized good performance in computer vision [5], [6], [7], [10], owing to the strong capacity of Transformer to model long-range dependency. However, the computation cost of self-attention in the Transformer is high, which limits its extensive application in video-related tasks. To overcome the drawback, Liu et al. [11] put forward Swin Transformer block (STB) to achieve linear computational complexity with respect to image size. Based on efficient and effective STB [11], Liang et al. [12] proposed residual STB (RSTB) to construct SwinIR for image restoration. Thanks to the powerful ability to model long-range dependency of RSTB, SwinIR [12] obtains SOTA performance compared with CNN-based methods. In this article, to acquire features $\{F_{2t-1}\}^N_{t=1}$ that capture long-range spatial information, we also use m_f RSTBs [12] to extract shallow features $\{F^s_{2t-1}\}^N_{t=1}$ further, as illustrated in Fig. 2. We can see that the RSTB is a residual block with several STBs and one convolutional layer. In addition, given a tensor X_{in} as input, the detailed process of STB to output X_{out} is formulated as

$$\begin{aligned} X &= \text{MSA}(\text{LayerNorm}(X_{\text{in}})) + X_{\text{in}} \\ X_{\text{out}} &= \text{MLP}(\text{LayerNorm}(X)) + X \end{aligned} \tag{1}$$

where MSA and X denote the multihead self-attention module and intermediate results, respectively.

B. Long-Short Term Feature Interpolation

To implement the super-resolution in the time dimension, we also utilize a feature interpolation module to synthesize the intermediate frames in the LR feature space, such as [13] and [40]. Specifically, given the two extracted features: F_1 and F_3, the feature interpolation module can synthesize the feature F_2 corresponding to the missing frame I^{lr}_2. Generally, to obtain the intermediate feature, we should capture the pixelwise motion information first. Optical flow is usually adopted to estimate the motion between video frames. However, there are several shortcomings in using optical flow for interpolation. The computational cost is high to calculate optical flow precisely, and estimated optical flow may be inaccurate due to the occlusion or motion blur, which causes poor interpolation results.

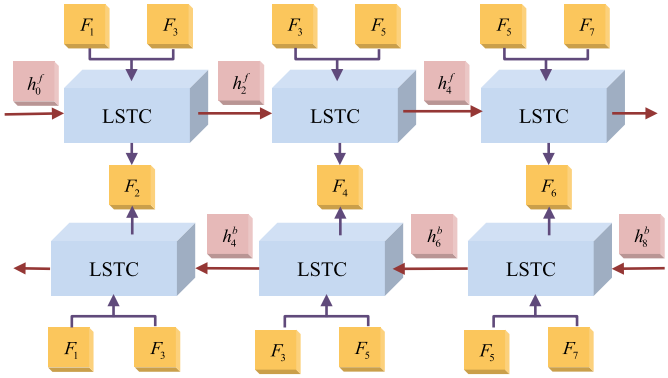


Fig. 3. Framework of our LSTFI module. It consists of LSTCs with bidirectional RNN, which can fully exploit the whole input video frame features during the interpolation process. Note that the two neighboring frame features and the hidden state from the previous LSTC provide short- and long-term contents for interpolation results, respectively. Here, h^f_0 and h^b_8 denote initialized hidden states for the forward and backward recurrent propagations, respectively. More specifically, h^f_0 serves as the forward hidden state for predicting the first missing frame feature, F_2, while h^b_8 is regarded as the backward hidden state for predicting the last missing frame feature, F_6.

Considering the drawback of optical flow, Xiang et al. [13] employed multilevel deformable convolution [42] to perform frame feature interpolation. The learned offset used in deformable convolution can implicitly capture forward and backward motion information, and achieve good performance. However, the synthesis of intermediate frame feature [13], [40] only utilizes the two neighboring frame features, which cannot fully explore the information from the other input frames to assist in the process. Unlike feature interpolation in previous STVSR algorithms [13], [40], we propose an LSTFI module to realize the intermediate frame in our STDAN, which is capable of exploiting helpful information from more input frames.

As illustrated in Fig. 3, we adopt a bidirectional RNN [41] to construct the LSTFI module, which consists of two branches

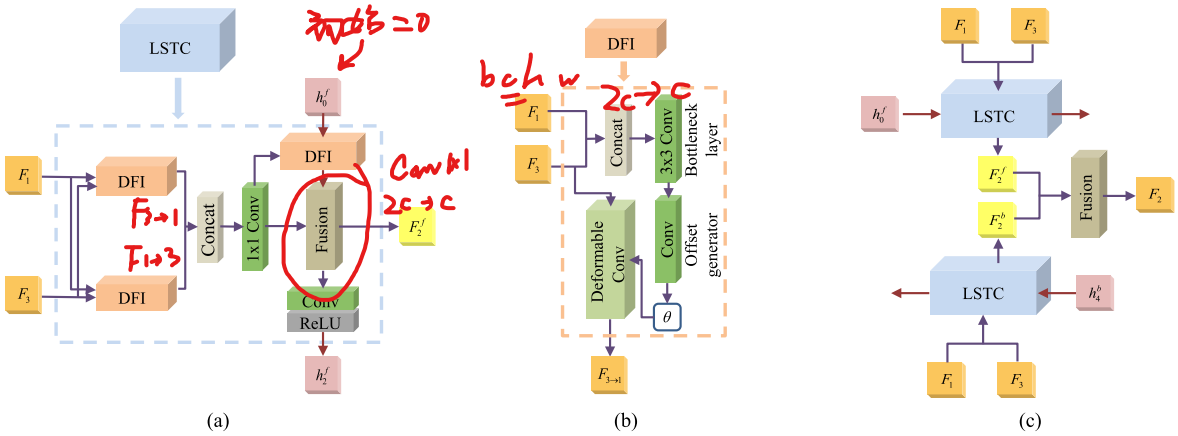


Fig. 4. Overview of the proposed LSTC and fusion process of interpolation results from forward and backward branches. We adopt DFI block [13] to adaptively align the hidden state from the previous LSTC with the current preliminary interpolation result. Note that the final intermediate frame feature is achieved by fusing the interpolation results from forward and backward branches. (a) Architecture of LSTC. (b) Architecture of DFI. (c) Fusion process of forward and backward features.

in forward and backward directions. Take the forward branch as an example. Two neighboring frame features and the hidden state from the previous LSTC are fed into each LSTC, and then, the LSTC generates the corresponding intermediate frame feature and current hidden state used for subsequent LSTCs. Here, the two neighboring frame features and hidden state serve as short- and long-term information for the intermediate feature, respectively. However, each branch's hidden state only considers the unidirectional information flow. To fully mine the information flow of these frame features for the interpolation procedure, we fuse interpolation results from LSTCs in the forward and backward branches to acquire the final intermediate frame feature.

The architecture of LSTC and the fusion process are shown in Fig. 4. Given two neighboring frame features F_1 and F_3, we employ deformable feature interpolation (DFI) block [13] to capture the forward and backward motion between the two features implicitly. For simplification, we take the feature $F_{3\to1}$ that has experienced backward motion compensation as an example. As illustrated in Fig. 4(b), the two frame features are concatenated along channel dimension and then pass through offset generation function H_{og}^b to predict an offset with backward motion information

$$\theta_{3\to1} = H_{\text{og}}^b([F_3, F_1]) \tag{2}$$

where H_{og}^b consists of convolutional layers and $[.,.]$ denotes the concatenation along channel dimension. With the learned offset, we adopt deformable convolution [44] as a motion compensation function to obtain compensated feature

$$F_{3\to1} = \text{DConv}([F_3, \theta_{3\to1}]) \tag{3}$$

where DConv denotes the operation of deformable convolution.

To blend the features $F_{1\to3}$ and $F_{3\to1}$ that have experienced forward and backward motion compensation, respectively, a 1×1 convolutional layer is applied, which can perform pixel-level linear weighting to achieve preliminary interpolation feature F_2^p. Note that the acquisition of feature F_2^p only utilizes the short-term information. In order to combine long-term information h_0^f, the hidden state from the previous LSTC, we first use the other DFI block to align h_0^f with the current feature F_2^p since there may be some misalignment. The process is expressed as

$$h_{0\to2}^f = \text{DAlign}\left(h_0^f, F_2^p\right) \tag{4}$$

where $\text{DAlign}(\cdot)$ indicates the operation of DFI block. At the end of LSTC, we apply a fusion function into aligned hidden state $h_{0\to2}^f$ and preliminary interpolation result F_2^p to obtain forward intermediate feature

$$F_2^f = H_{\text{fs}}\left(F_2^p, h_{0\to2}^f\right) \tag{5}$$

where H_{fs} refers to the fusion function. Then, the intermediate feature F_2^f passes through a convolutional layer and an activation layer in sequence to produce a hidden state h_2^f for the subsequent LSTC.

For fully exploring the whole input frame features for interpolation, the bidirectional RNN structure is utilized in our LSTFI module, so we fuse the forward intermediate feature F_2^f and the backward intermediate feature F_2^b to get the final intermediate frame feature F_2, as shown in Fig. 4(c).

C. Spatial–Temporal Deformable Feature Aggregation

With the assistance of the LSTFI module, we now have $2N-1$ frame features, where the generation of $N-1$ intermediate frame features combines their adjacent frame features with hidden states. Although the hidden states can introduce certain temporal information, the whole interpolation procedure hardly explicitly explores the temporal information between various frames. In addition, the N input frame features are merely processed independently in the feature extraction module. However, these frame features $\{F_t\}_{t=1}^{2N-1}$ are consecutive, which means that there are abundant temporal content without being exploited among these features.

For a feature vector $\mathbf{f}_t$ whose location is $\mathbf{p}_o$ on feature F_t, the simplest method to aggregate temporal information is adaptive fusion with the feature vector on the same location

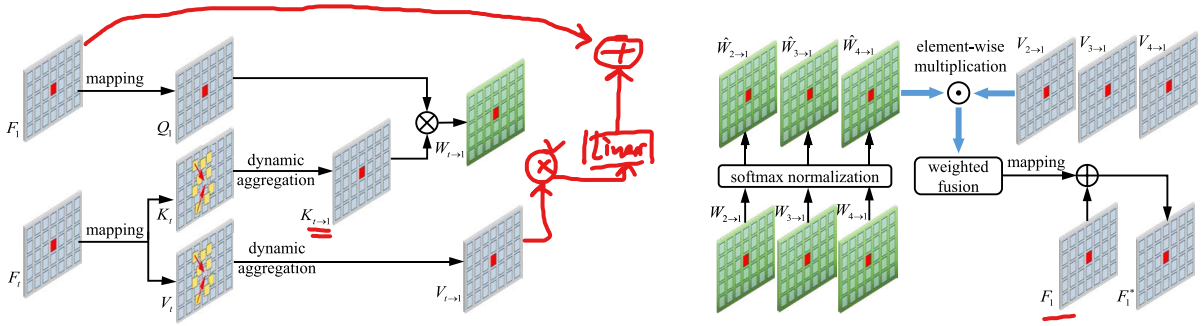


Fig. 5. Detailed process of the STDFA module. Note that we only show the case when the number of frame features is 4. Under the case, the value of t can be 2, 3, or 4 for frame feature F_1. (a) Cross-frame spatial aggregation. (b) Adaptive temporal aggregation.

from the other $2N-2$ frame features. However, the aggregation approach has several drawbacks. Generally, the corresponding point on other frame features may not be in the same location due to inter-frame motion. Furthermore, there are multiple helpful feature vectors for $\mathbf{f}_t$ from each of the $2N-2$ frame features. Based on the above analysis, we propose an STDFA module to mix cross-frame spatial information adaptively and capture long-range temporal information.

Specifically, we utilize the STDFA module to learn the residual auxiliary information from the remaining $2N-2$ frame features for each frame feature F_t. As presented in Fig. 5, the processing of the STDFA module can be divided into two parts: spatial aggregation and temporal aggregation. To adaptively fuse cross-frame spatial content of frame feature F_i from the other frame features, we perform deformable attention to each pair: F_i and F_j ($j \in [1, 2N-1]$, $j \neq i$). In detail, frame feature F_i passes through a linear layer to get embedded feature Q_i. Similarly, frame feature F_j is fed into two linear layers to obtain embedded features K_j and V_j, respectively.

To implement deformable attention between F_i and F_j, we first predict the offset map

$$\Delta M_{j\to i} = H_{\text{og}}([Q_i, K_j]) \tag{6}$$

where H_{og} indicates offset generation function consisting of several convolutional layers with $k \times k$ kernel. The offset map $\Delta M_{j\to i}$ at position $\mathbf{p}_o$ is expressed as

$$\Delta M_{j\to i}(\mathbf{p}_o) = \left[\Delta\mathbf{p}_1, \Delta\mathbf{p}_2, \ldots, \Delta\mathbf{p}_\xi, \ldots, \Delta\mathbf{p}_{k^2}\right]. \tag{7}$$

Then, the offsets $\Delta M_{j\to i}(\mathbf{p}_o)$ are combined with k^2 prespecified sampling locations to perform deformable sampling. Here, we denote the prespecified sampling location as $\mathbf{p}_\xi$, and the value set of $\mathbf{p}_\xi$ of $k \times k$ kernel is defined as

$$\mathbf{p}_\xi \in \left\{\left(-\left\lfloor\frac{k}{2}\right\rfloor, -\left\lfloor\frac{k}{2}\right\rfloor\right), \ldots, \left(\left\lfloor\frac{k}{2}\right\rfloor, \left\lfloor\frac{k}{2}\right\rfloor\right)\right\} \tag{8}$$

where $\lfloor\cdot\rfloor$ denotes the rounding down function.

With the offsets $\Delta M_{j\to i}(\mathbf{p}_o)$, the embedded feature vector $Q_i(\mathbf{p}_o)$ can attend k^2 related points in K_j. Nevertheless, not all the information of these k^2 points is helpful for $Q_i(\mathbf{p}_o)$. In addition, each point on embedded feature Q_i needs to search k^2 points, which inevitably causes a large storage occupation. To avoid irrelevant points and reduce storage occupation, we only choose the first T points that are most relevant.

To select the T points, we calculate the inner product between two embedded feature vectors as the relevance score

$$\text{RS}_{j\to i}(\mathbf{p}_o, \xi) = Q_i(\mathbf{p}_o) \cdot K_j(\mathbf{p}_o + \mathbf{p}_\xi + \Delta\mathbf{p}_\xi). \tag{9}$$

The larger the score, the more relevant the two points are. According to this criterion, we can determine the T points. In the following, to distinguish the selected T points from original k^2 points, we denote the prespecified sampling location and learned offset of the T points as $\overline{\mathbf{p}}_\xi$ and $\Delta\overline{\mathbf{p}}_\xi$, respectively.

To adaptively mingle the spatial information from the T locations for each embedded feature vector $Q_i(\mathbf{p}_o)$, we first adopt the softmax function to calculate the weight of these points

$$w_\xi = \frac{e^{Q_i(\mathbf{p}_o)\cdot K_j(\mathbf{p}_o+\overline{\mathbf{p}}_\xi+\Delta\overline{\mathbf{p}}_\xi)}}{\sum_{\xi=1}^{T} e^{Q_i(\mathbf{p}_o)\cdot K_j(\mathbf{p}_o+\overline{\mathbf{p}}_\xi+\Delta\overline{\mathbf{p}}_\xi)}}. \tag{10}$$

Then, with the weights and the embedded feature vector $K_j(\mathbf{p}_o + \overline{\mathbf{p}}_\xi + \Delta\overline{\mathbf{p}}_\xi)$, we can obtain corresponding updated embedded feature vector

$$K_{j\to i}(\mathbf{p}_o) = \sum_{\xi=1}^{T} w_\xi \cdot K_j\left(\mathbf{p}_o + \overline{\mathbf{p}}_\xi + \Delta\overline{\mathbf{p}}_\xi\right). \tag{11}$$

Same as $K_{j\to i}(\mathbf{p}_o)$, the updated vector $V_{j\to i}(\mathbf{p}_o)$ can be also achieved with the weight w_ξ. Finally, we calculate the updated relevant weight map $W_{j\to i}$ at each position $\mathbf{p}_o$ between Q_i and $K_{j\to i}$ for the following temporal aggregation:

$$W_{j\to i}(\mathbf{p}_o) = Q_i(\mathbf{p}_o) \cdot K_{j\to i}(\mathbf{p}_o). \tag{12}$$

To capture the temporal contexts of frame feature vector $F_i(\mathbf{p}_o)$ from the remaining $2N-2$ features, we also utilize the softmax function to adaptively aggregate feature vectors $V_{j\to i}(\mathbf{p}_o)$. Specifically, the normalized temporal weight of each vector $V_{j\to i}(\mathbf{p}_o)$ ($j \in [1, 2N-1]$, $j \neq i$) is expressed as

$$\hat{W}_{j\to i}(\mathbf{p}_o) = \frac{e^{W_{j\to i}(\mathbf{p}_o)}}{\sum_{j=1}^{2N-1, j\neq i} e^{W_{j\to i}(\mathbf{p}_o)}}. \tag{13}$$

Then, through fusing embedded feature vector $V_{j\to i}(\mathbf{p}_o)$ ($j \in [1, 2N-1]$, $j \neq i$) with the corresponding normalized weight, we can attain the embedded feature V_i^* that aggregates the spatial and temporal contexts from other $2N-2$ embedded features. The weighted fusion process is defined as

$$V_i^*(\mathbf{p}_o) = \sum_{j=1}^{2N-1, j\neq i} \hat{W}_{j\to i}(\mathbf{p}_o) \cdot V_{j\to i}(\mathbf{p}_o). \tag{14}$$

In the tail of the STDFA module, the embedded feature V_i^* is sent into a linear layer to acquire the residual auxiliary feature F_i^{res}. Finally, we add the frame feature F_i and the residual auxiliary feature F_i^{res} to get the enhanced feature F_i^* that aggregates spatial and temporal contexts from the other $2N-2$ frame features.

D. High-Resolution Frame Reconstruction

To reconstruct HR frames from the enhanced features $\{F_t^*\}_{t=1}^{2N-1}$, we first employ m_b RSTBs [12] to map feature F_t^* to deep feature F_t^d. Then, these deep features further pass through an upsampling module to realize the HR video frames $\{I_t^{\text{hr}}\}_{t=1}^{2N-1}$. Specifically, the upsampling module consists of the PixelShuffle layer [16] and several convolutional layers. For optimizing our proposed network, we adopt the Charbonnier function [17] as the reconstruction loss

$$L_{\text{rec}} = \sqrt{\left\| I_t^{\text{hr}} - I_t^{\text{GT}} \right\|^2 + \epsilon^2} \tag{15}$$

where I_t^{GT} indicates the ground truth of the tth reconstructed video frame I_t^{hr} and the value of ϵ is empirically set to 1×10^{-3}. With the loss function, our STDAN can be end-to-end trained to generate HR slow-motion videos from corresponding LR and LFR counterparts.

IV. Experiments

In this section, we first introduce the datasets and evaluation metrics used in our experiments. Then, the implementation details of our STDAN are elaborated. Next, we compare our proposed network with SOTA methods on public datasets. Finally, we carry out ablation studies to investigate the effect of the modules adopted in our STDAN.

A. Datasets and Evaluation Metrics

1) Datasets: We use the Vimeo-90K dataset [36] to train our network. Specifically, the Vimeo-90K dataset consists of more than 60 000 training video sequences, and each video sequence has seven frames. We adopt the raw seven frames as our HR and HFR supervisions. The corresponding four LR and LFR frames are downscaled by a factor of 4 with bicubic sampling from these odd-numbers ones. The Vimeo-90K also provides corresponding testsets that can be divided into Vimeo-Slow, Vimeo-Medium, and Vimeo-Fast according to the degree of motion. The three testsets serve as the evaluation datasets in our experiments. Same as STVSR methods [13], [40], six video sequences in Vimeo-Medium testset and three sequences in Vimeo-Slow testset are removed to avoid infinite values on PSNR. In addition, we report the results on Vid4 [37] and SPMC-11 [23] of different approaches.

2) Evaluation Metrics: To compare diverse STVSR networks quantitatively, peak signal-to-noise ratio (PSNR) and Structural SIMilarity (SSIM) [38] are adopted in our experiments as evaluation metrics. In this article, we calculate the PSNR and SSIM metrics on the Y channel of the YCbCr color space. In addition, we also compare the parameters and inference speed of various models.

B. Implementation Details

In our STDAN, the number of RSTBs in the feature extraction module and frame feature reconstruction module is 2 and 6, respectively, where each RSTB contains six STBs. In addition, the numbers of feature and embedded feature channels are set to be 64 and 72 separately. In the LSTFI module, we utilize a pyramid, cascading, and deformable (PCD) structure in [42] to achieve DFI. The hidden states in the forward and backward branches are initialized to zeros. In the STDFA module, the values of k and T are set to 3 and 2, respectively. We augment the training frames by randomly flipping horizontally and 90° rotations during the training process. Then, we crop the input LR frames with a size of 32×32 at random to the network, and the batch size is set to 18. Our model is trained by Adam [39] optimizer by setting $\beta_1 = 0.9$ and $\beta_2 = 0.999$. We employ cosine annealing to decay the learning rate [45] from $2e-4$ to $1e-7$. We implement the STDAN with PyTorch and train our model on six NVIDIA GTX-1080Ti GPUs.

C. Comparison With State-of-the-Art Methods

We compare our STDAN with existing SOTA one-stage STVSR approaches: STARnet [69], Zooming Slow-Mo [13], RSTT [68], and TMNet [40]. In addition, we also compare the performance of our network with SOTA two-stage STVSR algorithms, such as Zooming Slow-Mo [13] and TMNet [40]. Specifically, two-stage STVSR methods are composed of SOTA VFI and SR algorithms. These VFI networks are SuperSloMo [25], SepConv [33], and DAIN [35], respectively, while SOTA SR approaches are RCAN [60], RBPN [56], and EDVR [42].

Quantitative results of various STVSR methods are shown in Table I. From the table, we can see the following.

1) Our STDAN with fewer parameters obtains SOTA performance on both Vid4 [37] and Vimeo [36]. Here, we must indicate the reason leading to the lower results on Vid4 of the compared methods. Since the used transformer block requires that the length and width of the input video frame must be a multiple of 8, however, the size of LR frames on Vid4 is not always multiples of 8. Thus, we conducted a padding operation to the input LR frames before feeding them into the network. For fair comparisons, we adopted the same operation for all the compared methods. It causes that the Vid4 results of compared methods in Table I are lower than their reported results in the original papers.
2) For the SPMC-11 [23] dataset, our model is only 0.1 dB lower than STARnet [69] in terms of PSNR, but our STDAN acquires better results than it on SSIM [38] index, which demonstrates that our network can recover more correct structures. In addition, our model only needs one 13th parameter of STARnet.

Visual comparison of different models is displayed in Fig. 6. We observe that our STDAN, with the proposed LSTFI and STDFA modules, restores more accurate structures and fewer motion blurs compared with other STVSR approaches, which

TABLE I

QUANTITATIVE COMPARISONS OF OUR STDAN AND OTHER SOTA METHODS FOR STVSR. THE BEST TWO RESULTS ARE HIGHLIGHTED IN RED AND BLUE. NOTE THAT WE CONDUCT A PADDING OPERATION TO THE INPUT LR FRAMES BEFORE FEEDING THEM INTO THE NETWORKS, SO THE RESULTS OF COMPARATIVE METHODS ON VID4 ARE DIFFERENT FROM THE REPORTED RESULTS IN THE ORIGINAL PAPERS

VFI Method	(V)SR Method	Vid4		SPMC-11		Vimeo-Slow		Vimeo-Medium		Vimeo-Fast		Speed	Parameters
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	FPS	(Million)
SuperSloMo	Bicubic	22.84	0.5772	24.91	0.6874	28.37	0.8102	29.94	0.8477	31.88	0.8793	-	19.8
SuperSloMo	RCAN	23.78	0.6385	26.50	0.7527	30.69	0.8624	32.50	0.8884	34.52	0.9076	2.49	19.8+16.0
SuperSloMo	RBPN	24.00	0.6587	26.14	0.7582	30.48	0.8584	32.79	0.8930	34.73	0.9108	2.06	19.8+12.7
SuperSloMo	EDVR	24.22	0.6700	26.46	0.7689	30.99	0.8673	33.85	0.8967	35.05	0.9136	6.85	19.8+20.7
SepConv	Bicubic	23.51	0.6273	25.67	0.7261	29.04	0.8290	30.61	0.8633	32.27	0.8890	-	21.7
SepConv	RCAN	24.99	0.7259	28.16	0.8226	32.13	0.8967	33.59	0.9125	34.97	0.9195	2.42	21.7+16.0
SepConv	RBPN	25.75	0.7829	28.65	0.8614	32.77	0.9090	34.09	0.9229	35.07	0.9238	2.01	21.7+12.7
SepConv	EDVR	25.89	0.7876	28.86	0.8665	32.96	0.9112	34.22	0.9240	35.23	0.9252	6.36	21.7+20.7
DAIN	Bicubic	23.55	0.6268	25.68	0.7263	29.06	0.8289	30.67	0.8636	32.41	0.8910	-	24.0
DAIN	RCAN	25.03	0.7261	28.15	0.8224	32.26	0.8974	33.82	0.9146	35.27	0.9242	2.23	24.0+16.0
DAIN	RBPN	25.76	0.7783	28.57	0.8598	32.92	0.9097	34.45	0.9262	35.55	0.9300	1.88	24.0+12.7
DAIN	EDVR	25.90	0.7830	28.77	0.8649	33.11	0.9119	34.66	0.9281	35.81	0.9323	5.20	24.0+20.7
STARnet		25.99	0.7819	29.04	0.8509	33.10	0.9164	34.86	0.9356	36.19	0.9368	14.08	111.61
Zooming Slow-Mo		26.14	0.7974	28.80	0.8635	33.36	0.9138	35.41	0.9361	36.81	0.9415	16.50	11.10
RSTT		26.20	0.7991	28.86	0.8634	33.50	0.9147	35.66	0.9381	36.80	0.9403	15.36	7.67
TMNet		26.23	0.8011	28.78	0.8640	33.51	0.9159	35.60	0.9380	37.04	0.9435	14.69	12.26
STDAN (Ours)		26.28	0.8041	28.94	0.8687	33.66	0.9176	35.70	0.9387	37.10	0.9437	13.80	8.29

Fig. 6. Visual comparisons of different STVSR approaches on Vid4 and Vimeo datasets. We can see that our model can recover more accurate structures.

confirms the higher value on PSNR and SSIM achieved by our model.

D. Ablation Study

To investigate the effect of the proposed modules in our STDAN, we conduct comprehensive ablation studies in this section.

1) Feature Aggregation: To valid the effect of the proposed STDFA module, we establish a baseline: model Ω_1. It only adopts short-term information to perform interpolation and then directly reconstructs HR video frames through the frame feature reconstruction module without the feature aggregation process. In contrast, we compare three different models: Ω_2, Ω_3, and Ω_4 with feature aggregation. For the spatial–temporal

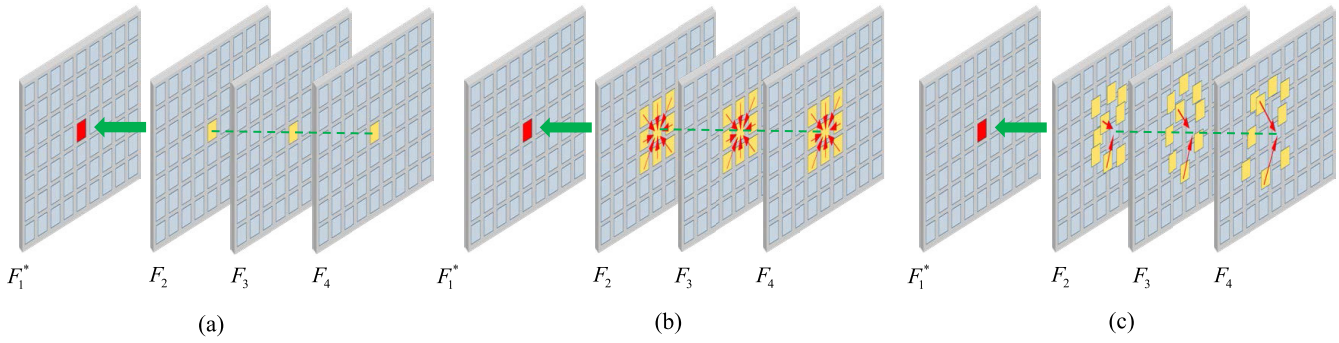


Fig. 7. Three different aggregation methods in the feature aggregation module. "STFA" refers to spatial–temporal feature aggregation. Note that we only show four frames for an illustration, and STFA in a deformable window denotes our STDFA module. (a) STFA in a 1×1 fixed window. (b) STFA in a 3×3 fixed window. (c) STFA in a deformable window.

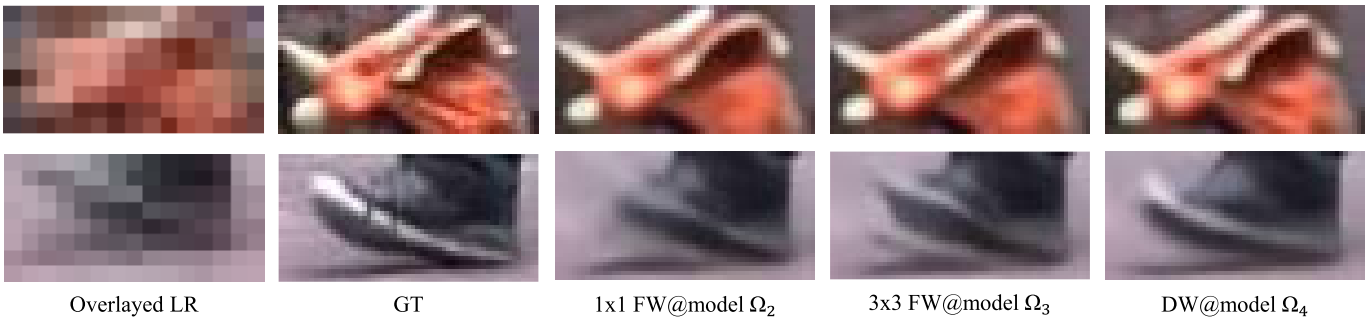


Fig. 8. Ablation study on the feature aggregation module. "FW" indicates the fixed window, while "DW" refers to the deformable window.

TABLE II

ABLATION STUDY ON THE PROPOSED MODULES. OUR LSTFI LEVERAGES MORE INPUT LR FRAMES TO ASSIST IN THE INTERPOLATION PROCESS. THE PROPOSED SPATIAL–TEMPORAL FEATURE AGGREGATION IN THE DEFORMABLE WINDOW CAN ADAPTIVELY CAPTURE SPATIAL–TEMPORAL CONTEXTS AMONG DIFFERENT FRAMES FOR HR FRAME RECONSTRUCTION. "STFA" INDICATES SPATIAL–TEMPORAL FEATURE AGGREGATION

Method		Ω_1	Ω_2	Ω_3	Ω_4	Ω_5
Parameters (M)		5.44	5.54	5.54	5.82	8.29
Feature Interpolation	Short-term feature interpolation	✓	✓	✓	✓	
	Long-short term feature interpolation					✓
Feature Aggregation	STFA in a 1x1 fixed window		✓			
	STFA in a 3x3 fixed window			✓		
	STFA in a deformable window				✓	✓
Vid4 (slow motion)		25.27	25.69	25.85	25.97	26.28
Vimeo-Fast (fast motion)		35.88	36.22	36.41	36.63	37.10

feature aggregation process in the model Ω_2, as illustrated in Fig. 7(a), each feature vector aggregates the information at the same position of other frame features, that is, the feature vector attends the valuable spatial content in a 1×1 window. We enlarge the window size of the model Ω_3 to 3. Considering large motions between frames, a deformable window is applied in the model Ω_4. As shown in Fig. 7(c), model Ω_4 adopts the STDFA module to perform feature aggregation.

Quantitative results on Vid4 [37] and Vimeo-Fast [36] datasets are shown in Table II. From the table, we know that: 1) the feature aggregation module can improve the reconstruction results and 2) the larger the spatial range of feature aggregation, the more useful information can be captured to enhance the recovery quality of HR frames. Qualitative results of the three models are represented in Fig. 8, which confirms that the feature aggregation in the deformable window can acquire more helpful content.

2) Feature Interpolation: To investigate the effect of the proposed LSTFI module, we compare two models: Ω_4 and Ω_5. As shown in Fig. 3, the model Ω_5 with LSTFI can exploit short-term information of two neighboring frames and long-term information of hidden states from other LSTCs. In comparison, model Ω_4 only uses two adjacent frames to interpolate the feature of the intermediate frame. From Table II, combining long- and short-term information can achieve better feature interpolation results, which leads to high-quality HR frames with more details, as illustrated in Fig. 9.

3) Efficiency of Selecting the First T Points: We also investigate the efficiency of determining the first T points in our STDFA module. Specifically, the model's inference time of each Vimeo sequence without/with the keypoint selection is 0.542 s/0.543 s, which demonstrates that the utilization of the keypoint selection in our STDFA module cannot lead to a significant increase in the inference time of the model.

4) STDFA Module Versus 3-D Convolutional Mechanism: Compared with the 3-D convolutional mechanism, our STDFA module has three advantages. First, instead of sampling the frame feature at fixed locations in a 3-D convolutional mechanism, STDFA can sample the feature at deformable locations with the learned offsets, which is beneficial for handling different video motions. Second, adaptive temporal aggregation enables STDFA not only to capture temporal information of

Fig. 9. Ablation study on the proposed modules. We can see that STDFA can effectively suppress blurring artifacts and recover correct visual structures, and the LSTFI can further help to reconstruct fine details.

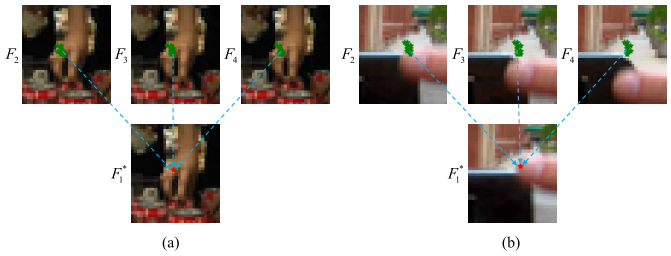


Fig. 10. Visualization of deformable sampling locations. The red star in the frame feature F_1^* denotes a feature vector, and the green stars in the other frame features indicate the corresponding sampling locations of the feature vector. Note that we directly show the sampling locations on the video frames rather than frame features, and we only show four frames for a better illustration. (a) Slow motion. (b) Fast motion.

adjacent video frames like what 3-D convolution usually does but also to aggregate long-range temporal contexts from more distinct video frames. Finally, rather than using globally shared and fixed convolution weights in a 3-D convolutional mechanism, our STDFA computes attention weights to dynamically leverage spatiotemporal contexts over video regions captured in terms of deformable offsets.

V. Discussion

A. Failure Analysis

Although our method can outperform existing SOTA methods, it is not perfect, especially when handling fast-motion videos. As shown in Fig. 10, we found that our deformable attention might sample wrong locations when video motions are fast. The key reason is that the predicted deformable offsets cannot accurately capture relevant visual contexts due to the large motions.

B. Conclusion

In this article, we propose a deformable attention network called STDAN for STVSR. Our STDAN can utilize more input video frames for the interpolation process. In addition, the network adopts deformable attention to dynamically capture spatial and temporal contexts among frames to enhance SR reconstruction. Thanks to the LSTFI and STDFA modules, our model demonstrates superior performance to recent SOTA STVSR approaches on public datasets.

References

[1] W. Yang, X. Zhang, Y. Tian, W. Wang, and J. Xue, "Deep learning for single image super-resolution: A brief review," *IEEE Trans. Multimedia*, vol. 21, no. 12, pp. 3106–3121, May 2019.

[2] U. Mudenagudi, S. Banerjee, and P. K. Kalra, "Space-time super-resolution using graph-cut optimization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 5, pp. 995–1008, May 2010.

[3] H. Takeda, P. V. Beek, and P. Milanfar, "Spatiotemporal video upscaling using motion-assisted steering kernel (mask) regression," in *High-Quality Visual Experience*. Berlin, Germany: Springer, 2010, pp. 245–274.

[4] O. Shahar, A. Faktor, and M. Irani, "Space-time super-resolution from a single video," in *Proc. CVPR*. Colorado Springs, CO, USA: IEEE, 2011, pp. 3353–3360, doi: 10.1109/CVPR.2011.5995360.

[5] N. Carion, F. Massa, G. Synnaeve, N. Usunier, A. Kirillov, and S. Zagoruyko, "End-to-end object detection with transformers," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 213–229.

[6] A. Dosovitskiy et al., "An image is worth 16×16 words: Transformers for image recognition at scale," 2020, *arXiv:2010.11929*.

[7] M. Chen, H. Peng, J. Fu, and H. Ling, "Autoformer: Searching transformers for visual recognition," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 12270–12280.

[8] T. Li, X. He, Q. Teng, Z. Wang, and C. Ren, "Space–time super-resolution with patch group cuts prior," *Signal Process., Image Commun.*, vol. 30, pp. 147–165, Jan. 2015.

[9] L. Zhang, J. Nie, W. Wei, Y. Li, and Y. Zhang, "Deep blind hyperspectral image super-resolution," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 6, pp. 2388–2400, Jun. 2020.

[10] B. Yan, H. Peng, J. Fu, D. Wang, and H. Lu, "Learning spatio-temporal transformer for visual tracking," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 10448–10457.

[11] Z. Liu et al., "Swin transformer: Hierarchical vision transformer using shifted windows," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2021, pp. 10012–10022.

[12] J. Liang, J. Cao, G. Sun, K. Zhang, L. Van Gool, and R. Timofte, "Swinir: Image restoration using swin transformer," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2021, pp. 1833–1844.

[13] X. Xiang, Y. Tian, Y. Zhang, Y. Fu, J. P. Allebach, and C. Xu, "Zooming slow-mo: Fast and accurate one-stage space-time video super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Oct. 2020, pp. 3370–3379.

[14] C. You, L. Han, A. Feng, R. Zhao, H. Tang, and W. Fan, "MEGAN: Memory enhanced graph attention network for space-time video super-resolution," in *Proc. IEEE/CVF Winter Conf. Appl. Comput. Vis. (WACV)*, Jan. 2022, pp. 1401–1411.

[15] K. C. Chan, X. Wang, K. Yu, C. Dong, and C. C. Loy, "Understanding deformable alignment in video super-resolution," in *Proc. AAAI Conf. Artif. Intell.*, vol. 35, no. 2, 2021, pp. 973–981.

[16] W. Shi et al., "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, Jun. 2016, pp. 1874–1883.

[17] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep Laplacian pyramid networks for fast and accurate super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2017, pp. 624–632.

[18] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 38, no. 2, pp. 295–307, Feb. 2015.

[19] E. Shechtman, Y. Caspi, and M. Irani, "Space-time super-resolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 4, pp. 531–545, Apr. 2005.

[20] M. Tassano, J. Delon, and T. Veit, "Fastdvdnet: Towards real-time deep video denoising without flow estimation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Oct. 2020, pp. 1354–1363.

[21] S. Niklaus and F. Liu, "Context-aware synthesis for video frame interpolation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 1701–1710.

[22] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive convolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 670–679.

[23] X. Tao, H. Gao, R. Liao, J. Wang, and J. Jia, "Detail-revealing deep video super-resolution," in *Proc. IEEE Int. Conf. Comput. Vis.*, Jun. 2017, pp. 4472–4480.

[24] J. Caballero et al., "Real-time video super-resolution with spatio–temporal networks and motion compensation," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2017, pp. 4778–4787.

[25] H. Jiang, D. Sun, V. Jampani, M.-H. Yang, E. Learned-Miller, and J. Kautz, "Super slomo: High quality estimation of multiple intermediate frames for video interpolation," in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, Jun. 2018, pp. 9000–9008.

[26] X. Zhang, R. Jiang, T. Wang, and J. Wang, "Recursive neural network for video deblurring," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 31, no. 8, pp. 3025–3036, Aug. 2020.

[27] J. Park, K. Ko, C. Lee, and C.-S. Kim, "BMBC: Bilateral motion estimation with bilateral cost volume for video interpolation," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 109–125.

[28] J. Lin, Y. Huang, and L. Wang, "FDAN: Flow-guided deformable alignment network for video super-resolution," 2021, *arXiv:2105.05640*.

[29] D. Fuoli, M. Danelljan, R. Timofte, and L. Van Gool, "Fast online video super-resolution with deformable attention pyramid," 2022, *arXiv:2202.01731*.

[30] J. Liang et al., "Recurrent video restoration transformer with guided deformable attention," 2022, *arXiv:2206.02146*.

[31] H. Lee, T. Kim, T.-Y. Chung, D. Pak, Y. Ban, and S. Lee, "AdaCoF: Adaptive collaboration of flows for video frame interpolation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 5316–5325.

[32] W. Bao, W.-S. Lai, X. Zhang, Z. Gao, and M.-H. Yang, "MEMC-Net: Motion estimation and motion compensation driven neural network for video interpolation and enhancement," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 43, no. 3, pp. 933–948, Mar. 2019.

[33] S. Niklaus, L. Mai, and F. Liu, "Video frame interpolation via adaptive separable convolution," in *Proc. IEEE Int. Conf. Comput. Vis. (ICCV)*, Venice, Italy, Oct. 2017, pp. 261–270.

[34] Z. Chen et al., "SiamBAN: Target-aware tracking with Siamese box adaptive network," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Aug. 2, 2022, doi: 10.1109/TPAMI.2022.3195759.

[35] W. Bao, W.-S. Lai, C. Ma, X. Zhang, Z. Gao, and M.-H. Yang, "Depth-aware video frame interpolation," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Long Beach, CA, USA, Jun. 2019, pp. 3698–3707.

[36] T. Xue, B. Chen, J. Wu, D. Wei, and W. T. Freeman, "Video enhancement with task-oriented flow," *Int. J. Comput. Vis.*, vol. 127, no. 8, pp. 1106–1125, 2019.

[37] C. Liu and D. Sun, "On Bayesian adaptive video super resolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 2, pp. 346–360, Feb. 2013.

[38] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[39] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.

[40] G. Xu, J. Xu, Z. Li, L. Wang, X. Sun, and M.-M. Cheng, "Temporal modulation network for controllable space-time video super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Oct. 2021, pp. 6388–6397.

[41] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Trans. Signal Process.*, vol. 45, no. 11, pp. 2673–2681, Nov. Jun. 1997.

[42] X. Wang, K. C. K. Chan, K. Yu, C. Dong, and C. C. Loy, "EDVR: Video restoration with enhanced deformable convolutional networks," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Long Beach, CA, USA, Jun. 2019, pp. 1954–1963.

[43] J. Dai et al., "Deformable convolutional networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, Jun. 2017, pp. 764–773.

[44] X. Zhu, H. Hu, S. Lin, and J. Dai, "Deformable ConvNets V2: More deformable, better results," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2019, pp. 9308–9316.

[45] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," 2016, *arXiv:1608.03983*.

[46] P. Yi, Z. Wang, K. Jiang, J. Jiang, and J. Ma, "Progressive fusion video super-resolution network via exploiting non-local spatio-temporal correlations," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, Oct. 2019, pp. 3106–3115.

[47] W. Li, X. Tao, T. Guo, L. Qi, J. Lu, and J. Jia, "MuCAN: Multi-correspondence aggregation network for video super-resolution," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 335–351.

[48] A. Kappeler, S. Yoo, Q. Dai, and A. K. Katsaggelos, "Video super-resolution with convolutional neural networks," *IEEE Trans. Comput. Imaging*, vol. 2, no. 2, pp. 109–122, Jun. 2016.

[49] B. Bare, B. Yan, C. Ma, and K. Li, "Real-time video super-resolution via motion convolution kernel estimation," *Neurocomputing*, vol. 367, pp. 236–245, Nov. 2019.

[50] Y. Zheng, X. Yu, M. Liu, and S. Zhang, "Single-image deraining via recurrent residual multiscale networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 33, no. 3, pp. 1310–1323, Mar. 2020.

[51] H. Wang, D. Su, C. Liu, L. Jin, X. Sun, and X. Peng, "Deformable non-local network for video super-resolution," *IEEE Access*, vol. 7, pp. 177734–177744, 2019.

[52] K. C. K. Chan, X. Wang, K. Yu, C. Dong, and C. C. Loy, "BasicVSR: The search for essential components in video super-resolution and beyond," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2021, pp. 4947–4956.

[53] Y. Jo, S. W. Oh, J. Kang, and S. J. Kim, "Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation," in *Proc. IEEE Conf. Comput. Vis. pattern Recognit.*, Jun. 2018, pp. 3224–3232.

[54] S. Li, F. He, B. Du, L. Zhang, Y. Xu, and D. Tao, "Fast spatio-temporal residual network for video super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 10522–10531.

[55] L. Wang, Y. Guo, Z. Lin, X. Deng, and W. An, "Learning for video super-resolution through hr optical flow estimation," in *Proc. Asian Conf. Comput. Vis.* Cham, Switzerland: Springer, 2018, pp. 514–529.

[56] M. Haris, G. Shakhnarovich, and N. Ukita, "Recurrent back-projection network for video super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2019, pp. 3897–3906.

[57] Y. Tian, Y. Zhang, Y. Fu, and C. Xu, "TDAN: Temporally-deformable alignment network for video super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 3360–3369.

[58] X. Ying, L. Wang, Y. Wang, W. Sheng, W. An, and Y. Guo, "Deformable 3D convolution for video super-resolution," *IEEE Signal Process. Lett.*, vol. 27, pp. 1500–1504, 2020.

[59] T. Isobe et al., "Video super-resolution with temporal group attention," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, Jun. 2020, pp. 8008–8017.

[60] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *Proc. Eur. Conf. Comput. Vis. (ECCV)*. Munich, Germany: Springer, 2018, pp. 294–310.

[61] X. Cheng and Z. Chen, "Multiple video frame interpolation via enhanced deformable separable convolution," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 44, no. 10, pp. 7029–7045, Oct. 2021.

[62] X. Shi, Z. Chen, H. Wang, D.-Y. Yeung, W.-K. Wong, and W.-C. Woo, "Convolutional LSTM network: A machine learning approach for precipitation nowcasting," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 1–9.

[63] M. Choi, H. Kim, B. Han, N. Xu, and K. M. Lee, "Channel attention is all you need for video frame interpolation," in *Proc. AAAI Conf. Artif. Intell.*, New York, NY, USA, 2020, pp. 10663–10671.

[64] E. Shechtman, Y. Caspi, and M. Irani, "Increasing space-time resolution in video," in *Proc. Eur. Conf. Comput. Vis.* Copenhagen, Denmark: Springer, 2002, pp. 753–768.

[65] O. Shahar, A. Faktor, and M. Irani, *Space-Time Super-Resolution From a Single Video*. Springs, CO, USA: IEEE, 2011.

[66] B. Zhao and X. Li, "Edge-aware network for flow-based video frame interpolation," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jun. 8, 2022, doi: 10.1109/TNNLS.2022.3178281.

[67] J. Kang, Y. Jo, S. W. Oh, P. Vajda, and S. J. Kim, "Deep space-time video upsampling networks," in *Proc. Eur. Conf. Comput. Vis.* Cham, Switzerland: Springer, 2020, pp. 701–717.

[68] Z. Geng, L. Liang, T. Ding, and I. Zharkov, "RSTT: Real-time spatial temporal transformer for space-time video super-resolution," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2022, pp. 17441–17451.

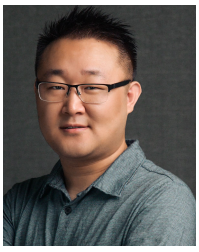
[69] M. Haris, G. Shakhnarovich, and N. Ukita, "Space-time-aware multi-resolution video enhancement," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2020, pp. 2859–2868.

Hai Wang (Member, IEEE) received the B.E. degree in electronic engineering from Xidian University, Xi'an, China, in 2019, and the M.E. degree in electronic engineering from Tsinghua University, Beijing, China, in 2022. He is currently pursuing the Ph.D. degree in statistical science with University College London, London, U.K.

His research interests include generative models, and video super-resolution and enhancement.

Xiaoyu Xiang received the B.E. degree in engineering physics from Tsinghua University, Beijing, China, in 2015, and the Ph.D. degree from the Department of Electrical and Computer Engineering, Purdue University, West Lafayette, IN, USA, in 2021.

She is currently a Research Scientist with the Meta Reality Labs, Menlo Park, CA, USA. Her primary area of research has been image and video restoration, novel view synthesis, and generative models.



Yapeng Tian (Member, IEEE) received the B.E. degree in electronic engineering from Xidian University, Xi'an, China, in 2013, the M.E. degree in electronic engineering from Tsinghua University, Beijing, China, in 2017, and the Ph.D. degree in computer science from the University of Rochester, Rochester, NY, USA, in 2022.

He has published more than 30 papers in leading journals and international conferences/workshops. His research interests are computer vision, computer audition, and machine learning.

Dr. Tian was selected for the 2023 AAAI New Faculty Highlights Program.

Wenming Yang (Senior Member, IEEE) received the Ph.D. degree in information and communication engineering from Zhejiang University, Hangzhou, China, in 2006.

He is currently an Associate Professor with the Shenzhen International Graduate School and the Department of Electronic Engineering, Tsinghua University, Beijing, China. His research interests include image processing, computer vision, pattern recognition, deep learning, and their applications.

Qingmin Liao (Senior Member, IEEE) received the B.S. degree in radio technology from the University of Electronic Science and Technology of China, Chengdu, China, in 1984, and the M.S. and Ph.D. degrees in signal processing and telecommunications from the University of Rennes 1, Rennes, France, in 1990 and 1994, respectively.

He is currently a Professor with the Shenzhen International Graduate School and the Department of Electronic Engineering, Tsinghua University, Beijing, China. His research interests include image/video processing, analysis, biometrics, and their applications.