

# Zooming Slow-Mo: Fast and Accurate One-Stage Space-Time Video Super-Resolution

Xiaoyu Xiang\*  
Purdue University  
xiang43@purdue.edu

Yun Fu  
Northeastern University  
yunfu@ece.neu.edu

Yapeng Tian\*  
University of Rochester  
yapengtian@rochester.edu

Jan P. Allebach†  
Purdue University  
allebach@ecn.purdue.edu

Yulun Zhang  
Northeastern University  
yulun100@gmail.com

Chenliang Xu†  
University of Rochester  
chenliang.xu@rochester.edu

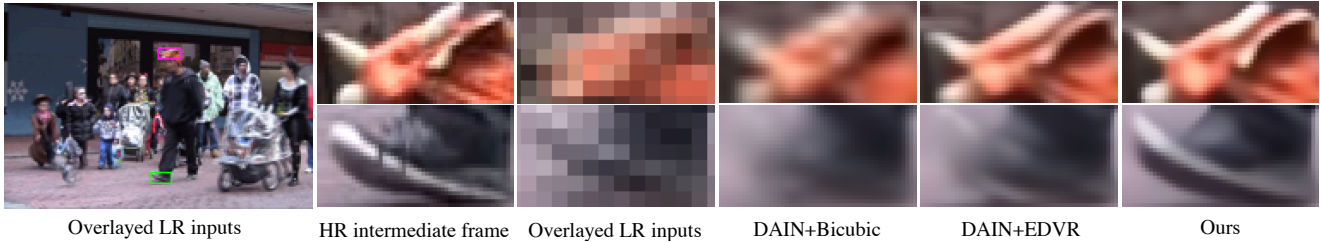


Figure 1: **Example of space-time video super-resolution.** We propose a one-stage space-time video super-resolution (STVSR) network to directly predict high frame rate (HFR) and high-resolution (HR) frames from the corresponding low-resolution (LR) and low frame rate (LFR) frames without explicitly interpolating intermediate LR frames. A HR intermediate frame  $t$  and its neighboring low-resolution frames:  $t - 1$  and  $t + 1$  as an overlaid image are shown. Compare to a state-of-the-art two-stage method: DAIN [1]+EDVR [37] on the HR intermediate frame  $t$ , our method is more capable of handling visual motions and therefore restores more accurate image structures and sharper edges. In addition, our network is more than 3 times faster on inference speed with a 4 times smaller model size than the DAIN+EDVR.

## Abstract

In this paper, we explore the space-time video super-resolution task, which aims to generate a **high-resolution (HR) slow-motion video from a low frame rate (LFR), low-resolution (LR) video**. A simple solution is to split it into two sub-tasks: **video frame interpolation (VFI) and video super-resolution (VSR)**. However, temporal interpolation and spatial super-resolution are intra-related in this task. **Two-stage methods cannot fully take advantage of the natural property**. In addition, state-of-the-art VFI or VSR networks require a large frame-synthesis or reconstruction module for predicting high-quality video frames, which makes the two-stage methods have large model sizes and thus be time-consuming. To overcome the problems, we propose a **one-stage space-time video super-resolution framework, which directly synthesizes an HR slow-motion video from an LFR, LR video**. Rather than synthesizing missing LR video frames

as VFI networks do, we **firstly temporally interpolate LR frame features in missing LR video frames capturing local temporal contexts by the proposed feature temporal interpolation network**. Then, we propose a **deformable ConvLSTM to align and aggregate temporal information simultaneously for better leveraging global temporal contexts**. Finally, a **deep reconstruction network is adopted to predict HR slow-motion video frames**. Extensive experiments on benchmark datasets demonstrate that the proposed method not only achieves better quantitative and qualitative performance but also is more than three times faster than recent two-stage state-of-the-art methods, e.g., DAIN+EDVR and DAIN+RBPN.

## 1. Introduction

Space-Time Video Super-Resolution (STVSR) [30] aims to automatically generate a photo-realistic video sequence with a high space-time resolution from a low-resolution and

\*Equal contribution; †Equal advising.

low frame rate input video. Since HR slow-motion videos are more visually appealing containing fine image details and clear motion dynamics, they are desired in rich applications, such as film making and high-definition television.

To tackle the problem, most existing works in previous literatures [30, 22, 33, 28, 6, 14] usually adopt hand-crafted regularization and make strong assumptions. For example, space-time directional smoothness prior is adopted in [30], and [22] assumes that there is no significant change in illumination for the static pixels. However, these strong constraints make the methods have limited capacity in modeling various and diverse space-time visual patterns. Besides, the optimization for these methods is usually computationally expensive (*e.g.*,  $\sim 1$  hour for 60 frames in [22]).

In recent years, deep convolutional neural networks have shown promising efficiency and effectiveness in various video restoration tasks, such as video frame interpolation (VFI) [24], video super-resolution (VSR) [4], and video deblurring [32]. To design an STVSR network, one straightforward way is by directly combining a video frame interpolation method (*e.g.*, SepConv [25], ToFlow [40], DAIN [1] *etc.*) and a video super-resolution method (*e.g.*, DUF [11], RBPN [8], EDVR [37] *etc.*) in a two-stage manner. It firstly interpolates missing intermediate LR video frames with VFI and then reconstructs all HR frames with VSR. However, temporal interpolation and spatial super-resolution in STVSR are intra-related. The two-stage methods splitting them into two individual procedures cannot make full use of this natural property. In addition, to predict high-quality video frames, both state-of-the-art VFI and VSR networks require a big frame reconstruction network. Therefore, the composed two-stage STVSR model will contain a large number of parameters and is computationally expensive.

To alleviate the above issues, we propose a unified one-stage STVSR framework to learn temporal interpolation and spatial super-resolution simultaneously. We propose to adaptively learn a deformable feature interpolation function for temporally interpolating intermediate LR frame features rather than synthesizing pixel-wise LR frames as in two-stage methods. The learnable offsets in the interpolation function can aggregate useful local temporal contexts and help the temporal interpolation handle complex visual motions. In addition, we introduce a new deformable ConvLSTM model to effectively leverage global contexts with simultaneous temporal alignment and aggregation. HR video frames can be reconstructed from the aggregated LR features with a deep SR reconstruction network. To this end, the one-stage network can learn end-to-end to map an LR, LFR video sequence to its HR, HFR space in a sequence-to-sequence manner. Experimental results show that the proposed one-stage STVSR framework outperforms state-of-the-art two-stage methods even with much fewer parameters. An example is illustrated in Figure 1.

The contributions of this paper are three-fold: (1) We propose a one-stage space-time super-resolution network that can address temporal interpolation and spatial SR simultaneously in a unified framework. Our one-stage method is more effective than two-stage methods taking advantage of the intra-relatedness between the two sub-problems. It is also computationally more efficient since only one frame reconstruction network is required rather than two large networks as in state-of-the-art two-stage approaches. (2) We propose a frame feature temporal interpolation network leveraging local temporal contexts based on deformable sampling for intermediate LR frames. We devise a novel deformable ConvLSTM to explicitly enhance temporal alignment capacity and exploit global temporal contexts for handling large motions in videos. (3) Our one-stage method achieves state-of-the-art STVSR performance on both Vid4 [17] and Vimeo [40]. It is 3 times faster than the two-stage network: DAIN [1] + EDVR [37] while having a nearly  $4\times$  reduction in model size. **The source code is released in <https://github.com/Mukosame/Zooming-SlowMo-CVPR-2020>.**

## 2. Related Work

In this section, we discuss works on three related topics: video frame interpolation (VFI), video super-resolution (VSR), and space-time video super-resolution (STVSR).

**Video Frame Interpolation** The target of video frame interpolation is to synthesize non-existent intermediate frames in between the original frames. Meyer *et al.* [21] introduced a phase-based frame interpolation method, which generates intermediate frames through per-pixel phase modification. Long *et al.* [19] predicted intermediate frames directly with an encoder-decoder CNN. Niklaus *et al.* [24, 25] regarded the frame interpolation as a local convolution over the two input frames and used a CNN to learn a spatially-adaptive convolution kernel for each pixel for high-quality frame synthesis. To explicitly handle motions, there are also many flow-based video interpolation approaches [10, 18, 23, 2, 1]. These methods usually have inherent issues with inaccuracies and missing information from optical flow results. In our one-stage STVSR framework, rather than synthesizing the intermediate LR frames as current VFI methods do, we interpolate features from two neighboring LR frames to directly synthesize LR feature maps for missing frames without requiring explicit supervision.

**Video Super-Resolution** Video super-resolution aims to reconstruct an HR video frame from the corresponding LR frame (reference frame) and its neighboring LR frames (supporting frames). **One key problem for VSR is how to temporally align the LR supporting frames with the reference frame.** Several VSR methods [4, 34, 26, 36, 40] use

参考系

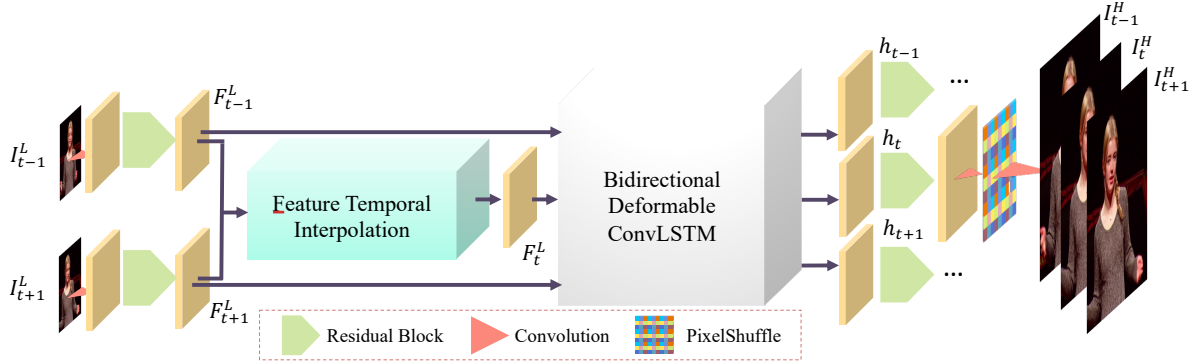


Figure 2: Overview of our one-stage STVSR framework. It directly reconstructs consecutive HR video frames without synthesizing LR intermediate frames  $I_t^L$ . Feature temporal interpolation and bidirectional deformable ConvLSTM are utilized to leverage local and global temporal contexts for better exploiting temporal information and handling large motions. Note that we only show two input LR frames from a long sequence in this figure for a better illustration.

optical flow for explicit temporal alignment, which first estimates motions between the reference frame and each supporting frame with optical flow and then warps the supporting frame using the predicted motion map. Recently, RBPV proposes to incorporate the single image and multi-frame SR for VSR in which flow maps are directly concatenated with LR video frames. However, it is difficult to obtain accurate flow; and flow warping also introduces artifacts into the aligned frames. To avoid this problem, DUF [11] with dynamic filters and TDAN [35] with deformable alignment were proposed for implicit temporal alignment without motion estimation. EDVR [37] extends the deformable alignment in TDAN by exploring multiscale information. However, most of the above methods are many-to-one architectures, and they need to process a batch of LR frames to predict only one HR frame, which makes the methods computationally inefficient. Recurrent neural networks, such as convolutional LSTMs [39] (ConvLSTM), can ease sequence-to-sequence (S2S) learning; and they are adopted in VSR methods [15, 9] for leveraging temporal information. However, without explicit temporal alignment, the RNN-based VSR networks have limited capability in handling large and complex motions within videos. To achieve efficient yet effective modeling, unlike existing methods, we propose a novel ConvLSTM structure embedded with an explicit state updating cell for space-time video super-resolution.

Rather than simply combining a VFI network and a VSR network to solve STVSR, we propose a more efficient and effective one-stage framework that simultaneously learns temporal feature interpolation and spatial SR without accessing to LR intermediate frames as supervision.

**Space-Time Video Super-Resolution** Shechtman *et al.* [29] firstly proposed to extend SR to the space-time domain. Since pixels are missing in LR frames and

even several entire LR frames are unavailable, STVSR is a highly ill-posed inverse problem. To increase video resolution both in time and space, [29] combines information from multiple video sequences of dynamic scenes obtained at sub-pixel and sub-frame misalignments with a directional space-time smoothness regularization to constrain the ill-posed problem. Mudanagudi [22] posed STVSR as a reconstruction problem using the Maximum a posteriori-Markov Random Field [7] with graph-cuts [3] as the solver. Takeda *et al.* [33] exploited local orientation and local motion to steer spatio-temporal regression kernels. Shahar *et al.* [28] proposed to exploit a space-time patch recurrence prior within natural videos for STVSR. However, these methods have limited capacity to model rich and complex space-time visual patterns, and the optimization for these methods is usually computationally expensive. To address these issues, we propose a one-stage network to directly learn the mapping between partial LR observations and HR video frames and to achieve fast and accurate STVSR.

### 3. Space-Time Video Super-Resolution

Given an LR, LFR video sequence:  $\mathcal{I}^L = \{I_{2t-1}^L\}_{t=1}^{n+1}$ , our goal is to generate the corresponding high-resolution slow-motion video sequence:  $\mathcal{I}^H = \{I_t^H\}_{t=1}^{2n+1}$ . To intermediate HR frames  $\{I_{2t}^H\}_{t=1}^n$ , there are no corresponding LR counterparts in the input sequence. To fast and accurately increase resolution in both space and time domains, we propose a one-stage space-time super-resolution framework: Zooming Slow-Mo as illustrated in Figure 2. The framework mainly consists of four parts: *feature extractor*, *frame feature temporal interpolation module*, *deformable ConvLSTM*, and *HR frame reconstructor*.

We first use a feature extractor with a convolutional layer and  $k_1$  residual blocks to extract feature maps:  $\{F_{2t-1}^L\}_{t=1}^{n+1}$  from input video frames. Taking the feature maps as input,

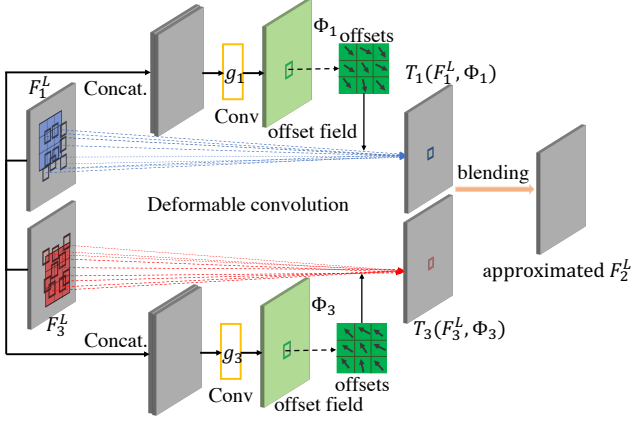


Figure 3: Frame feature temporal interpolation based on deformable sampling. Since approximated  $F_2^L$  will be used to predict the corresponding HR frame, it will implicitly enforce the learnable offsets to capture accurate local temporal contexts and be motion-aware.

we then synthesize the LR feature maps:  $\{F_{2t}^L\}_{t=1}^n$  of intermediate frames with the proposed frame feature interpolation module. Furthermore, to better leverage temporal information, we use a deformable ConvLSTM to process the consecutive feature maps:  $\{F_t^L\}_{t=1}^{2n+1}$ . Unlike vanilla ConvLSTM, the proposed deformable ConvLSTM can simultaneously perform temporal alignment and aggregation. Finally, we reconstruct the HR slow-mo video sequence from the aggregated feature maps.

### 3.1. Frame Feature Temporal Interpolation

Given extracted feature maps:  $F_1^L$  and  $F_3^L$  from input LR video frames:  $I_1^L$  and  $I_3^L$ , we want to synthesize the feature map  $F_2^L$  corresponding to the missing intermediate LR frame  $I_2^L$ . Traditional video frame interpolation networks usually perform temporal interpolation on pixel-wise video frames, which will lead to a two-stage STVSR design. Unlike previous methods, we propose to learn a feature temporal interpolation function  $f(\cdot)$  to directly synthesize the intermediate feature map  $F_2^L$  (see Fig. 3). A general form of the interpolation function can be formulated as:

$$F_2^L = f(F_1^L, F_3^L) = H(T_1(F_1^L, \Phi_1), T_3(F_3^L, \Phi_3)) , \quad (1)$$

where  $T_1(\cdot)$  and  $T_3(\cdot)$  are two sampling functions and  $\Phi_1$  and  $\Phi_3$  are the corresponding sampling parameters;  $H(\cdot)$  is a blending function to aggregate sampled features.

For generating accurate  $F_2^L$ , the  $T_1(\cdot)$  should capture forward motion information between  $F_1^L$  and  $F_2^L$ , and the  $T_3(\cdot)$  should capture backward motion information between  $F_3^L$  and  $F_2^L$ . However, the  $F_2^L$  is not available for computing forward and backward motion information in this task.

To alleviate this problem, we use motion information between  $F_1^L$  and  $F_3^L$  to approximate forward and backward

motion information. Inspired by recent deformable alignment in [35] for VSR, we propose to use deformable sampling functions to implicitly capture motion information for frame feature temporal interpolation. With exploring rich local temporal contexts by deformable convolutions in sampling functions, our feature temporal interpolation can even handle very large motions in videos.

The two sampling functions share the same network design but have different weights. For simplicity, we use the  $T_1(\cdot)$  as an example. It takes LR frame feature maps  $F_1^L$  and  $F_3^L$  as input to predict an offset for sampling the  $F_1^L$ :

$$\Delta p_1 = g_1([F_1^L, F_3^L]) , \quad (2)$$

where  $\Delta p_1$  is a learnable offset and also refers to the sampling parameter:  $\Phi_1$ ;  $g_1$  denotes a general function of several convolution layers;  $[\cdot]$  denotes the channel-wise concatenation. With the learned offset, the sampling function can be performed with a deformable convolution [5, 42]:

$$T_1(F_1^L, \Phi_1) = DCConv(F_1^L, \Delta p_1) . \quad (3)$$

Similarly, we can learn an offset  $\Delta p_3 = g_3([F_3^L, F_1^L])$  as the sampling parameter:  $\Phi_3$  and then obtain sampled features  $T_3(F_3^L, \Phi_3)$  with a deformable convolution.

To blend the two sampled features, we use a simple linear blending function  $H(\cdot)$ :

$$F_2^L = \alpha * T_1(F_1^L, \Phi_1) + \beta * T_3(F_3^L, \Phi_3) , \quad (4)$$

where  $\alpha$  and  $\beta$  are two learnable  $1 \times 1$  convolution kernels and  $*$  is a convolution operator. Since the synthesized LR feature map  $F_2^L$  will be used to predict the intermediate HR frame  $I_2^L$ , it will enforce the synthesized LR feature map to be close to the real intermediate LR feature map. Therefore, the two offsets  $\Delta p_1$  and  $\Delta p_3$  will implicitly learn to capture the forward and backward motion information, respectively.

Applying the designed deformable temporal interpolation function to  $\{F_{2t-1}^L\}_{t=1}^{n+1}$ , we can obtain intermediate frame feature maps  $\{F_{2t}^L\}_{t=1}^n$ .

### 3.2. Deformable ConvLSTM

Now we have consecutive frame feature maps:  $\{F_t^L\}_{t=1}^{2n+1}$  for generating the corresponding HR video frames, which will be a sequence-to-sequence mapping. It has been proved in previous video restoration tasks [40, 34, 37] that temporal information is vital. Therefore, rather than reconstructing HR frames from the corresponding individual feature maps, we aggregate temporal contexts from neighboring frames. ConvLSTM [39] is a popular 2D sequence data modeling method and we can adopt it to perform temporal aggregation. **At the time step  $t$ , the ConvLSTM updates hidden state  $h_t$  and cell state  $c_t$  with:**

$$h_t, c_t = ConvLSTM(h_{t-1}, c_{t-1}, F_t^L) . \quad (5)$$



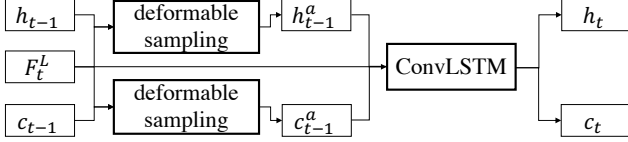


Figure 4: Deformable ConvLSTM for better exploiting global temporal contexts and handling fast motion videos. At time step  $t$ , we introduce state updating cells to learn deformable sampling to adaptively align hidden state  $h_{t-1}$  and cell state  $c_{t-1}$  with current input feature map:  $F_t^L$ .

From its state updating mechanism [39], we can learn that the ConvLSTM can only implicitly capture motions between previous states:  $h_{t-1}$  and  $c_{t-1}$  and the current input feature map with small convolution receptive fields. Therefore, ConvLSTM has limited ability to handle large motions in natural videos. If a video has large motions, there will be a severe temporal mismatch between previous states and  $F_t^L$ . Then,  $h_{t-1}$  and  $c_{t-1}$  will propagate mismatched “noisy” content rather than useful global temporal contexts into  $h_t$ . Consequently, the reconstructed HR frame  $I_t^H$  from  $h_t$  will suffer from annoying artifacts.

To tackle the large motion problem and effectively exploit global temporal contexts, we explicitly embed a state-updating cell with deformable alignment into ConvLSTM (see Fig. 4):

$$\begin{aligned}
 \Delta p_t^h &= g^h([h_{t-1}, F_t^L]) , \\
 \Delta p_t^c &= g^c([c_{t-1}, F_t^L]) , \\
 h_{t-1}^a &= DConv(h_{t-1}, \Delta p_t^h) , \\
 c_{t-1}^a &= DConv(c_{t-1}, \Delta p_t^c) , \\
 h_t, c_t &= ConvLSTM(h_{t-1}^a, c_{t-1}^a, F_t^L) ,
 \end{aligned} \tag{6}$$

where  $g^h$  and  $g^c$  are general functions of several convolution layers,  $\Delta p_t^h$  and  $\Delta p_t^c$  are predicted offsets, and  $h_{t-1}^a$  and  $c_{t-1}^a$  are aligned hidden and cell states, respectively. Compared with vanilla ConvLSTM, we explicitly enforce the hidden state  $h_{t-1}$  and cell state  $c_{t-1}$  to align with the current input feature map  $F_t^L$  in our deformable ConvLSTM, which makes it more capable of handling motions in videos. Besides, to fully explore temporal information, we use the Deformable ConvLSTM in a bidirectional manner [27]. We feed temporally reversed feature maps into the same Deformable ConvLSTM and concatenate hidden states from forward pass and backward pass as the final hidden state  $h_t^2$  for HR frame reconstruction.

### 3.3. Frame Reconstruction

To reconstruct HR video frames, we use a temporally shared synthesis network, which takes individual hidden

<sup>2</sup>We use  $h_t$  to denote final hidden state, but it will refer to a concatenated hidden state in the Bidirectional Deformable ConvLSTM.

state  $h_t$  as input and outputs the corresponding HR frame. It has  $k_2$  stacked residual blocks [16] for learning deep features and utilizes a sub-pixel upscaling module with PixelShuffle as in [31] to reconstruct HR frames  $\{I_t^H\}_{t=1}^{2n+1}$ . To optimize our network, we use a reconstruction loss function:

$$l_{rec} = \sqrt{\|I_t^{GT} - I_t^H\|^2 + \epsilon^2} , \tag{7}$$

where  $I_t^{GT}$  refers to the  $t$ -th ground-truth HR video frame, Charbonnier penalty function [13] is used as the loss term, and  $\epsilon$  is empirically set to  $1 \times 10^{-3}$ . Since the space and time SR problems are intra-related in STVSR, our model is end-to-end trainable and can simultaneously learn this spatio-temporal interpolation with only supervision from HR video frames.

### 3.4. Implementation Details

In our implementation,  $k_1 = 5$  and  $k_2 = 40$  residual blocks are used in feature extraction and HR frame reconstruction modules, respectively. We randomly crop a sequence of down-sampled image patches with the size of  $32 \times 32$  and take out the odd-indexed 4 frames as LFR and LR inputs, and the corresponding consecutive 7-frame sequence of  $4 \times 3$  size as supervision. Besides, we perform data augmentation by randomly rotating  $90^\circ$ ,  $180^\circ$  and  $270^\circ$ , and horizontal-flipping. We adopt a Pyramid, Cascading and Deformable (PCD) structure in [37] to employ deformable alignment and apply Adam [12] optimizer, where we decay the learning rate with a cosine annealing for each batch [20] from  $4e - 4$  to  $1e - 7$ . The batch size is set to be 24 and trained on 2 Nvidia Titan XP GPUs.

## 4. Experiments and Analysis

### 4.1. Experimental Setup

**Datasets** We use Vimeo-90K as the training set [40], including more than 60,000 7-frame training video sequences. The dataset is widely used in previous VFI and VSR works [2, 1, 35, 8, 37]. Vid4 [17] and Vimeo testset [40] are used as the evaluation datasets. To measure the performance of different methods under different motion conditions, we split the Vimeo testset into fast motion, medium motion, and slow motion sets as in [8], which include 1225, 4977 and 1613 video clips, respectively. We remove 5 video clips from the original medium motion set and 3 clips from the slow motion set, which have consecutively all-black background frames that will lead to infinite values on PSNR. We generate LR frames by bicubic with a downsampling factor 4 and use odd-indexed LR frames as input to predict the corresponding consecutive HR and HFR frames.

<sup>3</sup>Considering recent state-of-the-art methods (e.g., EDVR [37] and RBPN [8]) use only 4 as the upscaling factor, we adopt the same practice.

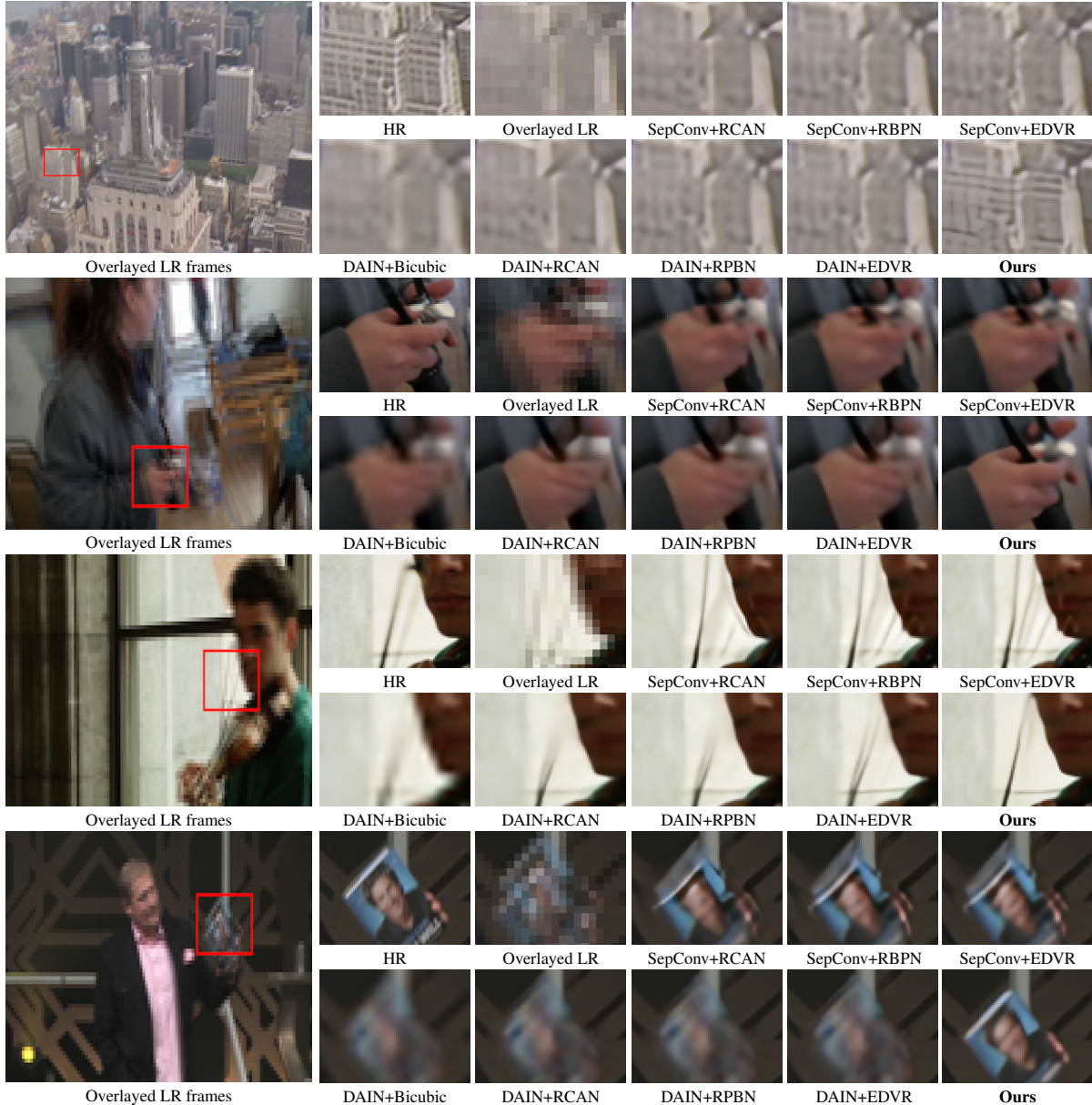


Figure 5: Visual comparisons of different methods on video frames from Vid4 and Vimeo datasets. Our one-stage Zooming SlowMo model can reconstruct more visually appealing HR video frames with more accurate image structures and fewer blurring artifacts.

**Evaluation** Peak Signal-to-Noise Ratio (PSNR) and Structural Similarity Index (SSIM) [38] are adopted to evaluate STVSR performance of different methods. To measure the efficiency of different networks, we also compare the model sizes and inference time of the entire Vid4 [17] dataset measured on one Nvidia Titan XP GPU.

## 4.2. Comparison to State-of-the-art Methods

We compare the performance of our one-stage Zooming SlowMo network to two-stage methods composed of state-of-the-art (SOTA) VFI and VSR networks. Three recent

SOTA VFI approaches, SepConv [25], Super-SloMo<sup>4</sup> [10], and DAIN [1], are compared. To achieve STVSR, three SOTA SR models, including single-image SR model, RCAN [41], and two recent VSR models, RBPN [8] and EDVR [37], are used to generate HR frames from both original LR and interpolated LR frames.

Quantitative results are shown in Table 1. From the table, we can learn the following facts: (1) DAIN+EDVR is

<sup>4</sup>Since there is no official source code released, we used an unofficial PyTorch implementation from <https://github.com/avinashpaliwal/Super-SloMo>.

Table 1: Quantitative comparison of our results and two-stage VFI and VSR methods on testsets. The best two results are highlighted in red and blue colors, respectively. The total runtime is measured on the entire Vid4 dataset [17]. Note that we omit the baseline models with Bicubic when comparing in terms of runtime.

VFI Method	SR Method	Vid4		Vimeo-Fast		Vimeo-Medium		Vimeo-Slow		Parameters (Million)	Runtime-VFI (s)	Runtime-SR (s)	Total Runtime (s)	Average Runtime (s/frame)
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM					
SuperSloMo [10]	Bicubic	22.84	0.5772	31.88	0.8793	29.94	0.8477	28.37	0.8102	19.8	0.28	-	-	-
SuperSloMo [10]	RCAN [41]	23.80	0.6397	34.52	0.9076	32.50	0.8884	30.69	0.8624	19.8+16.0	0.28	68.15	68.43	0.4002
SuperSloMo [10]	RBPB [8]	23.76	0.6362	34.73	0.9108	32.79	0.8930	30.48	0.8584	<b>19.8+12.7</b>	0.28	82.62	82.90	0.4848
SuperSloMo [10]	EDVR [37]	24.40	0.6706	35.05	0.9136	33.85	0.8967	30.99	0.8673	19.8+20.7	0.28	24.65	<b>24.93</b>	<b>0.1458</b>
SepConv [25]	Bicubic	23.51	0.6273	32.27	0.8890	30.61	0.8633	29.04	0.8290	21.7	2.24	-	-	-
SepConv [25]	RCAN [41]	24.92	0.7236	34.97	0.9195	33.59	0.9125	32.13	0.8967	21.7+16.0	2.24	68.15	70.39	0.4116
SepConv [25]	RBPB [8]	26.08	0.7751	35.07	0.9238	34.09	0.9229	32.77	0.9090	21.7+12.7	2.24	82.62	84.86	0.4963
SepConv [25]	EDVR [37]	25.93	0.7792	35.23	0.9252	34.22	0.9240	32.96	0.9112	21.7+20.7	2.24	24.65	26.89	0.1572
DAIN [1]	Bicubic	23.55	0.6268	32.41	0.8910	30.67	0.8636	29.06	0.8289	24.0	8.23	-	-	-
DAIN [1]	RCAN [41]	25.03	0.7261	35.27	0.9242	33.82	0.9146	32.26	0.8974	24.0+16.0	8.23	68.15	76.38	0.4467
DAIN [1]	RBPB [8]	25.96	0.7784	35.55	0.9300	34.45	0.9262	32.92	0.9097	24.0+12.7	8.23	82.62	90.85	0.5313
DAIN [1]	EDVR [37]	<b>26.12</b>	<b>0.7836</b>	<b>35.81</b>	<b>0.9323</b>	<b>34.66</b>	<b>0.9281</b>	<b>33.11</b>	<b>0.9119</b>	24.0+20.7	8.23	24.65	32.88	0.1923
Ours		<b>26.31</b>	<b>0.7976</b>	<b>36.81</b>	<b>0.9415</b>	<b>35.41</b>	<b>0.9361</b>	<b>33.36</b>	<b>0.9138</b>	<b>11.10</b>	-	-	<b>10.36</b>	<b>0.0606</b>

the best performing two-stage approach among the compared 12 methods; (2) VFI matters, especially for fast motion videos. Although RBPB and EDVR perform much better than RCAN for VSR, however, when equipped with more advanced VFI network DAIN, DAIN+RCAN can achieve comparable or even better performance than SepConv+RBPB and SepConv+EDVR on the Vimeo -Fast set; (3) VSR also matters. For example, with the same VFI network: DAIN, EDVR consistently achieves better STVSR performance than other VSR methods. In addition, we can see that our network outperforms the DAIN+EDVR by 0.19dB on Vid4, 0.25dB on Vimeo-Slow, 0.75dB on Vimeo-Medium, and 1dB on Vimeo-Fast in terms of PSNR. The significant improvements obtained on videos with fast motions demonstrate that our one-stage network with simultaneously leveraging local and global temporal contexts is more capable of handling diverse spatio-temporal patterns, including challenging large motions in videos than two-stage methods.

Moreover, we also investigate model sizes and runtime of different networks in Table 1. For synthesizing high-quality frames, SOTA VFI and VSR networks usually have very large frame reconstruction modules. Thus, the composed two-stage SOTA STVSR networks will contain a huge number of parameters. With only one frame reconstruction module, our one-stage model has much fewer parameters than the SOTA two-stage networks. From Table 1, we can see that it is more than 4× and 3× smaller than the DAIN+EDVR and DAIN+RBPB, respectively. The small model size makes our network more than 3× faster than the DAIN+EDVR and 8× faster than DAIN+RBPB. Compared to two-stage methods with a fast VFI network: SuperSloMo, our method is still more than 2× faster.

Visual results of different methods are illustrated in Figure 5. We see that our method achieves noticeably visual improvements over other two-stage methods. Clearly, the proposed network can synthesize visually appealing HR video frames with more fine details, more accurate



Figure 6: Ablation study on feature interpolation. The naive feature interpolation model without deformable sampling will obtain overly smooth results for videos with fast motions. With the proposed deformable feature interpolation (DFI), our model can well exploit local contexts in adjacent frames, thus is more effective in handling large motions.

structures, and fewer blurring artifacts even for challenging fast motion video sequences. We also observe that current SOTA VFI methods: SepConv and DAIN fail to handle large motions. Consequently, two-stage networks tend to generate HR frames with severe motion blurs. In our one-stage framework, we simultaneously learn temporal and spatial SR with exploring the natural intra-relatedness. Even with a much smaller model, our network can well address the large motion issue in temporal SR.

### 4.3. Ablation Study

We have already shown the superiority of our one-stage framework over two-stage networks. To further demonstrate the effectiveness of different modules in our network, we make a comprehensive ablation study.

**Effectiveness of Deformable Feature Interpolation** To investigate the proposed deformable feature interpolation





Figure 7: Ablation study on Deformable ConvLSTM (DConvLSTM). ConvLSTM will fail when meeting videos with fast motions. Embedded with state updating cells, the proposed DConvLSTM is more capable of leveraging global temporal contexts for reconstructing more accurate visual content even for fast motion videos.

Table 2: Ablation study on the proposed modules. Proposed deformable feature interpolation network and deformable ConvLSTM can effectively handle motions and improve STVSR performance, while the vanilla ConvLSTM performs worse when meeting large motions in videos.

Method	(a)	(b)	(c)	(d)	(e)
Naive feature interpolation					
Deformable feature interpolation (DFI)	✓				
ConvLSTM			✓		
Deformable ConvLSTM (DConvLSTM)				✓	
Bidirectional DConvLSTM					✓
Vid4 (slow motion)	25.18	25.34	25.68	26.18	26.31
Vimeo-Fast (fast motion)	34.93	35.66	35.39	36.56	36.81

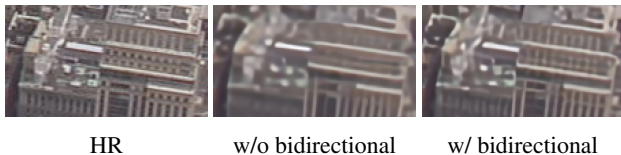


Figure 8: Ablation study on the bidirectional mechanism in DConvLSTM. Adding the bidirectional mechanism into DConvLSTM, the model can leverage both previous and future contexts, and therefore reconstructs more visually appealing frames with finer image details, especially for video frames at the first time step, which can not access any temporal information from other frames.

(DFI) module, we introduce two baselines: (a) and (b), where the model (a) only uses convolutions to blend LR features without deformable sampling functions as in model (b). In addition, neither (a) or (b) has ConvLSTM or DConvLSTM. From Table 2, we find that (b) outperforms (a) by 0.16dB on Vid4 with slow motions and 0.73dB on Vimeo-Fast with fast motions in terms of PSNR. Figure 6 shows a visual comparison. We can see that (a) produces a face with severe motion blur, while the proposed deformable feature interpolation with exploiting local temporal contexts can effectively address the large motion issue and help the model (b) generate a frame with more clear face structures and details. The superiority of the proposed DFI module demonstrates that the learned offsets in the deformable sampling

functions can effectively exploit local temporal contexts and successfully capture forward and backward motions even without any explicit supervision.

**Effectiveness of Deformable ConvLSTM** To validate the effect of the proposed Deformable ConvLSTM (DConvLSTM), we compare four different models: (b), (c), (d), and (e), where (c) adds a vanilla ConvLSTM structure into (b), (d) utilizes the proposed DConvLSTM, and (e) adopts a DConvLSTM in a bidirectional manner.

From Table 2, we can see that (c) outperforms (b) on Vid4 with slow motion videos while it is worse than (b) on Vimeo-Fast with fast motion sequences. The results validate that vanilla ConvLSTM can leverage useful global temporal contexts for slow motion videos, but cannot handle large motions in videos. Moreover, we observe that (d) is significantly better than both (b) and (c), which demonstrates that our DConvLSTM can successfully learn the temporal alignment between previous states and the current feature map. Therefore, it can better exploit global contexts for reconstructing visually pleasing frames with more details. Visual results in Figure 7 further support our findings.

In addition, we compare (e) and (d) in Table 2 and Figure 8 to verify the bidirectional mechanism in DConvLSTM. From Table 2, we can see that (e) can further improve STVSR performance over (d) on both slow motion and fast motion testing sets. The visual results in Figure 8 further shows that our full model with a bidirectional mechanism can restore more visual details by making full use of global temporal information for all input video frames.

## 5. Conclusion

In this paper, we propose a one-stage framework for space-time video super-resolution to directly reconstruct high-resolution and high frame rate videos without synthesizing intermediate low-resolution frames. To achieve this, we introduce a deformable feature interpolation network for feature-level temporal interpolation. Furthermore, we propose a deformable ConvLSTM for aggregating temporal



information and handling motions. With such a one-stage design, our network can well explore intra-relatedness between temporal interpolation and spatial super-resolution in the task. It enforces our model to adaptively learn to leverage useful local and global temporal contexts for alleviating large motion issues. Extensive experiments show that our one-stage framework is more effective yet efficient than existing two-stage networks, and the proposed feature temporal interpolation network and deformable ConvLSTM are capable of handling very challenging fast motion videos.

## Acknowledgements

The work was partly supported by NSF 1741472, 1813709, and 1909912. This article solely reflects the opinions and conclusions of its authors but not the funding agents.

## References

- [1] Wenbo Bao, Wei-Sheng Lai, Chao Ma, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. Depth-aware video frame interpolation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3703–3712, 2019.
- [2] Wenbo Bao, Wei-Sheng Lai, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. Memc-net: Motion estimation and motion compensation driven neural network for video interpolation and enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2019.
- [3] Yuri Boykov, Olga Veksler, and Ramin Zabih. Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(11):1222–1239, 2001.
- [4] Jose Caballero, Christian Ledig, Andrew Aitken, Alejandro Acosta, Johannes Totz, Zehan Wang, and Wenzhe Shi. Real-time video super-resolution with spatio-temporal networks and motion compensation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4778–4787, 2017.
- [5] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 764–773, 2017.
- [6] Esmaeil Faramarzi, Dinesh Rajan, and Marc P Christensen. Space-time super-resolution from multiple-videos. In *2012 11th International Conference on Information Science, Signal Processing and their Applications (ISSPA)*, pages 23–28. IEEE, 2012.
- [7] Stuart Geman and Donald Geman. Stochastic relaxation, gibbs distributions, and the bayesian restoration of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (6):721–741, 1984.
- [8] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Recurrent back-projection network for video super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3897–3906, 2019.
- [9] Yan Huang, Wei Wang, and Liang Wang. Video super-resolution via bidirectional recurrent convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):1015–1028, 2017.
- [10] Huaizu Jiang, Deqing Sun, Varun Jampani, Ming-Hsuan Yang, Erik Learned-Miller, and Jan Kautz. Super slo-mo: High quality estimation of multiple intermediate frames for video interpolation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9000–9008, 2018.
- [11] Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, and Seon Joo Kim. Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3224–3232, 2018.
- [12] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- [13] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 624–632, 2017.
- [14] Tao Li, Xiaohai He, Qizhi Teng, Zhengyong Wang, and Chao Ren. Space-time super-resolution with patch group cuts prior. *Signal Processing: Image Communication*, 30:147–165, 2015.
- [15] Bee Lim and Kyoung Mu Lee. Deep recurrent resnet for video super-resolution. In *2017 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 1452–1455. IEEE, 2017.
- [16] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 136–144, 2017.
- [17] Ce Liu and Deqing Sun. A bayesian approach to adaptive video super resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 209–216. IEEE, 2011.
- [18] Ziwei Liu, Raymond A Yeh, Xiaoou Tang, Yiming Liu, and Aseem Agarwala. Video frame synthesis using deep voxel flow. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4463–4471, 2017.
- [19] Gucan Long, Laurent Kneip, Jose M Alvarez, Hongdong Li, Xiaohu Zhang, and Qifeng Yu. Learning image matching by simply watching video. In *European Conference on Computer Vision*, pages 434–450. Springer, 2016.
- [20] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.
- [21] Simone Meyer, Oliver Wang, Henning Zimmer, Max Grosse, and Alexander Sorkine-Hornung. Phase-based frame interpolation for video. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1410–1418, 2015.

- [22] Uma Mudenagudi, Subhashis Banerjee, and Prem Kumar Kalra. Space-time super-resolution using graph-cut optimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(5):995–1008, 2010.
- [23] Simon Niklaus and Feng Liu. Context-aware synthesis for video frame interpolation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1701–1710, 2018.
- [24] Simon Niklaus, Long Mai, and Feng Liu. Video frame interpolation via adaptive convolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 670–679, 2017.
- [25] Simon Niklaus, Long Mai, and Feng Liu. Video frame interpolation via adaptive separable convolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 261–270, 2017.
- [26] Mehdi SM Sajjadi, Raviteja Vemulapalli, and Matthew Brown. Frame-recurrent video super-resolution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6626–6634, 2018.
- [27] Mike Schuster and Kuldip K Paliwal. Bidirectional recurrent neural networks. *IEEE Transactions on Signal Processing*, 45(11):2673–2681, 1997.
- [28] Oded Shahar, Alon Faktor, and Michal Irani. *Space-time super-resolution from a single video*. IEEE, 2011.
- [29] Eli Shechtman, Yaron Caspi, and Michal Irani. Increasing space-time resolution in video. In *European Conference on Computer Vision*, pages 753–768. Springer, 2002.
- [30] Eli Shechtman, Yaron Caspi, and Michal Irani. Space-time super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, (4):531–545, 2005.
- [31] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1874–1883, 2016.
- [32] Shuochen Su, Mauricio Delbracio, Jue Wang, Guillermo Sapiro, Wolfgang Heidrich, and Oliver Wang. Deep video deblurring for hand-held cameras. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1279–1288, 2017.
- [33] Hiroyuki Takeda, Peter Van Beek, and Peyman Milanfar. Spatiotemporal video upscaling using motion-assisted steering kernel (mask) regression. In *High-Quality Visual Experience*, pages 245–274. Springer, 2010.
- [34] Xin Tao, Hongyun Gao, Renjie Liao, Jue Wang, and Jiaya Jia. Detail-revealing deep video super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4472–4480, 2017.
- [35] Yapeng Tian, Yulun Zhang, Yun Fu, and Chenliang Xu. Tdan: Temporally deformable alignment network for video super-resolution. *arXiv preprint arXiv:1812.02898*, 2018.
- [36] Longguang Wang, Yulan Guo, Zaiping Lin, Xinpu Deng, and Wei An. Learning for video super-resolution through hr optical flow estimation. In *Asian Conference on Computer Vision*, pages 514–529. Springer, 2018.
- [37] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 0–0, 2019.
- [38] Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.
- [39] SHI Xingjian, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. Convolutional lstm network: A machine learning approach for precipitation nowcasting. In *Advances in Neural Information Processing Systems*, pages 802–810, 2015.
- [40] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision*, 127(8):1106–1125, 2019.
- [41] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European Conference on Computer Vision*, pages 286–301, 2018.
- [42] Xizhou Zhu, Han Hu, Stephen Lin, and Jifeng Dai. Deformable convnets v2: More deformable, better results. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 9308–9316, 2019.

# Appendices

## Network Architecture

We further illustrate the feature temporal interpolation network in Figure 9 and the proposed STVSR framework in Figure 10 to help readers better understand the overall structure of our proposed network.

To make our paper be concise and easy to follow, we use a simple version of deformable sampling to introduce the proposed feature temporal interpolation and deformable ConvLSTM. However, in our implementation, as stated in Section 3.4 of the paper, we adopt a Pyramid, Cascading and Deformable (PCD) structure<sup>5</sup> as in [37] to implement the deformable sampling, which can exploit multi-scale contexts with a feature pyramid.

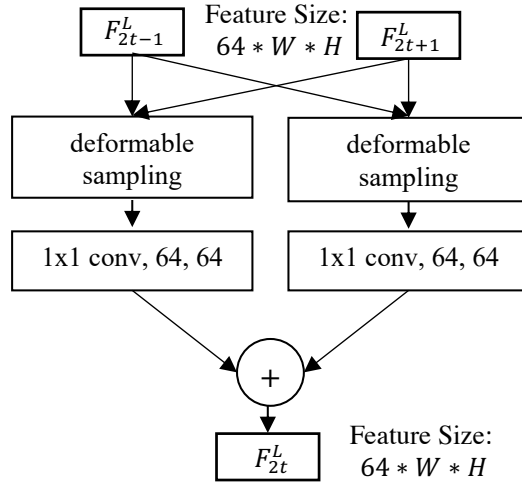


Figure 9: Feature temporal interpolation for intermediate LR frames. It will predict an intermediate LR frame feature map  $F_{2t}^L$  from two neighboring feature maps:  $F_{2t-1}^L$  and  $F_{2t+1}^L$ , where  $t = 1, 2, \dots, n$ . Note that the deformable sampling module on the left samples features from  $F_{2t-1}^L$  with generated sampling parameters from both  $F_{2t-1}^L$  and  $F_{2t+1}^L$ ; on the contrary, the deformable sampling module on the right samples features from  $F_{2t+1}^L$ .

<sup>5</sup>The official PyTorch implementation of the PCD can be found in <https://github.com/xinntao/EDVR>.

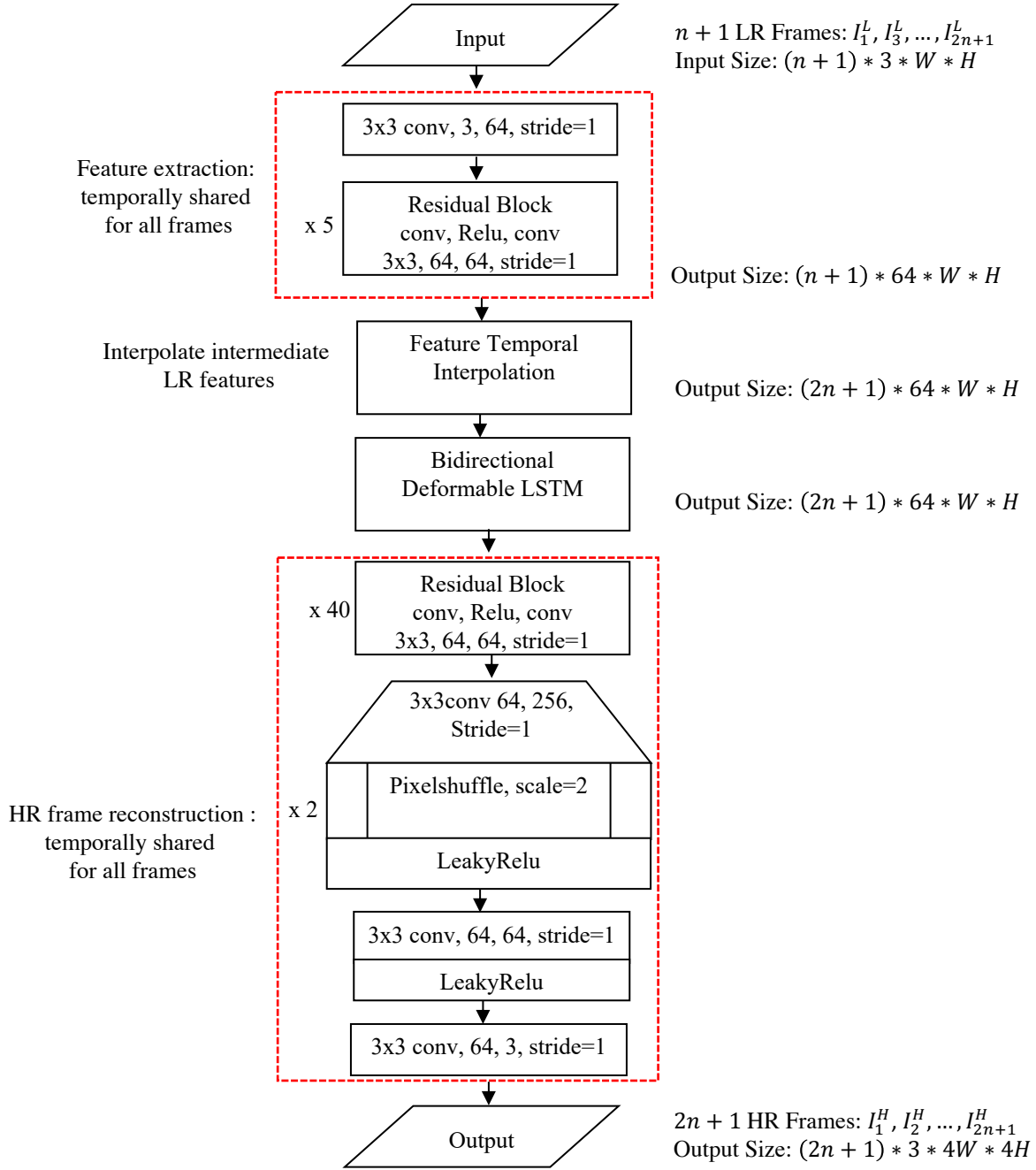


Figure 10: Flowchart of the proposed one-stage STVSR framework. The feature extraction and HR frame reconstruction networks are temporally shared for all frames, in which different frames are processed independently.