How Video Super-Resolution and Frame Interpolation Mutually Benefit

Chengcheng Zhou
Tsinghua University
zhoucc19@mails.tsinghua.edu.cn

Zongqing Lu
Tsinghua University
luzq@sz.tsinghua.edu.cn

Linge Li
Huawei Technologies Co., Ltd.
lilinge@huawei.com

Qiangyu Yan
Huawei Technologies Co., Ltd.
yanqiangyu1@huawei.com

Jing-Hao Xue
University College London
jinghao.xue@ucl.ac.uk

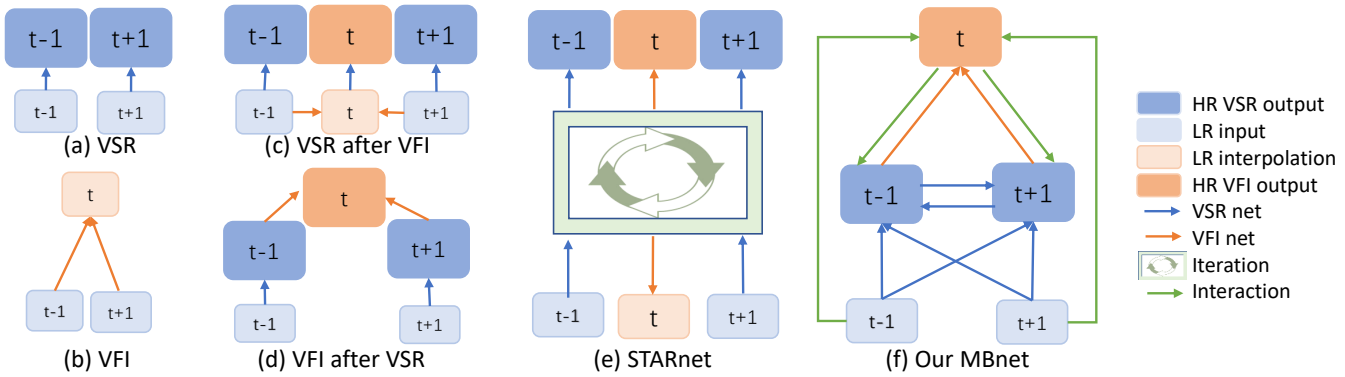

Figure 1: Various types of combinations of VSR and VFI. Our MBnet (f) is built on the interaction of VSR and VFI to exploit better the space-time inter-dependence and make VSR and VFI mutually benefit each other. Compared with STARnet [13], our MBnet feeds results of VFI back to VSR only once and achieves competitive results with much fewer parameters.

ABSTRACT

Video super-resolution (VSR) and video frame interpolation (VFI) are inter-dependent for enhancing videos of low resolution and low frame rate. However, most studies treat VSR and temporal VFI as independent tasks. In this work, we design a spatial-temporal super-resolution network based on exploring the interaction between VSR and VFI. The main idea is to improve the middle frame of VFI by the super-resolution (SR) frames and feature maps from VSR. In the meantime, VFI also provides extra information for VSR and thus, through interacting, the SR of consecutive frames of the original video can also be improved by the feedback from the generated middle frame. Drawing on this, our approach leverages a simple interaction of VSR and VFI and achieves state-of-the-art performance on various datasets. Due to such a simple strategy, our approach is universally applicable to any existing VSR or VFI networks for effectively improving their video enhancement performance.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.
ACM MM '21, October 20-24, 2021, Chengdu, China
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-8651-7/21/10…$15.00
https://doi.org/10.1145/3474085.3475672

CCS CONCEPTS

• **Computing methodologies** → **Computational photography**.

KEYWORDS

Video super-resolution; video frame interpolation; spatial-temporal inter-dependence

ACM Reference Format:
Chengcheng Zhou, Zongqing Lu, Linge Li, Qiangyu Yan, and Jing-Hao Xue. 2021. How Video Super-Resolution and Frame Interpolation Mutually Benefit. In *ACM MM '21: 29th ACM International Conference on Multimedia, October 20-24, 2021, Chengdu, China.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3474085.3475672

1 INTRODUCTION

To suit high-resolution and high-frame-rate displays such as UHD 4K TVs and players, video super-resolution (VSR) [29] (Fig.1(a)) and video-frame interpolation (VFI) [2] (Fig.1(b)) are often used to achieve the conversion from a low-frame-rate (LFR) and low-resolution (LR) video to a high-frame-rate (HFR) and high-resolution (HR) video.

At present, most VSR networks (*e.g.* RBPN [12], TDAN [29], EDVR [32]) and VFI methods (*e.g.* SepConv [23], AdaCoF [20], DAIN [2]) are independently studied. However, the essence of both VSR and VFI tasks is to supplement and generate pixels based on adjacent pixel information in the same frame or from the adjacent

reference frames. On account of the similarity between pixels of the same object both in the time domain and the space domain, if VSR and VFI tasks are processed separately, it will not be able to make full use of the temporal and spatial inter-dependence.

Space-time video super-resolution (STVSR) [13, 33], strongly required in film slow-motion making and high-resolution television, aims to automatically generate videos of high space-time resolution from LFR and LR input videos. To design an STVSR network, one straightforward way is to cascade VSR and VFI networks in a two-stage manner, either first enhancing spatial resolution and then temporally interpolating HR frames (Fig.1(d)), or first interpolating missing intermediate LR frames and then reconstructing all HR frames (Fig.1(c)). However, such a simple cascaded connection of VSR and VFI will introduce spatial and temporal irregularity without alignment.

Therefore, this study focuses on the joint enhancement of VSR and VFI. Different from STARnet [13], our MBnet (Fig.1(f)) is designed on the basis of interaction between VSR and VFI at the pixel level and does not need many iterations, enabling MBnet to be conveniently combined with any existing independent VSR and VFI networks. We find that the VSR network pays more attention to the low-level context feature while the VFI network lays emphasis on high-level motion information, so they can be mutually beneficial: VSR can benefit from the VFI feedback for higher frame rate and more accurate motion estimation and compensation, and VFI can get more context details from the VSR features. Drawing on this, our approach leverages a simple interaction of VSR and VFI and achieves state-of-the-art performance on various datasets. Moreover, our design is a universally applicable structure that can be easily combined with any independent VSR and VFI networks to effectively improve their video enhancement results.

The main contributions of our work are four-fold:

1) We deeply explore why VSR and VFI are better to be joint and how they mutually benefit from each other. Especially, our VSR features are more suitable to enhance VFI than the features extracted by the independent ResNet that used in [2, 22].

2) We propose an effective interaction structure between VSR and VFI; it can also be conveniently used to combine most independent VSR and VFI networks for STVSR, *e.g.* RBPN [12]. Moreover, our proposed MBnet achieves competitive results on several popular datasets compared to state-of-the-art methods.

3) In order to reduce the huge computational costs of the flow-based VSR and VFI, we adopt a lightweight flow network on LR frames and a flow refinement network on HR frames, which achieves better performance with fewer parameters than typical flow nets (*e.g.* PWCNet [27]) widely used in VSR and VFI tasks [13, 19, 34]

4) We get the flow to the middle frame by the Flow Refine Net instead of directly use the interpolation of adjacent flows in [3, 16], and we try a new supervised loss of flow, which adopts a latest optical flow network [28] to get the referential flow for supervision and achieves better perceptional experience.

2 RELATED WORK

Our work is mainly related to deep learning based methods for three video enhancement topics: VSR, VFI, and STVSR.

2.1 Video Super-Resolution (VSR)

Video super-resolution aims to reconstruct an HR video frame from the corresponding LR frame (reference frame) and its neighboring LR frames (supporting frames). In recent years, a lot of image super-resolution methods based on deep learning [4, 11, 36] have emerged and gradually became the mainstream. However, simply adapting image super-resolution into video super-resolution will bring temporal inconsistency and artifacts due to the motion between frames. A common solution is to introduce optical flow [9, 27] and then warp the supporting frame to the reference frame by using the predicted flow map, *e.g.* RBPN [12], ToFlow [34], SOF-VSR [31]. Other methods adapt dynamic filters [17], non-local spatio-temporal correlations [30, 35], channel attention[15] or deformable ConvLSTM [29, 32] to make implicit motion compensation for alignment. However, the cost of explicit flow calculation in HR frames is huge if starting from scratch.

2.2 Video Frame Interpolation (VFI)

The aim of VFI is to synthesize non-existent intermediate frames between original adjacent frames. Kernel-based VFI nets [2, 20] adopt spatially adaptive separable convolution to enlarge the reception field, but they have limited ability when the displacement is out of the kernel area. Similarly to VSR, most VFI networks [2, 14, 20, 22, 34] adopt optical flow for explicit motion compensation and temporal alignment. Some methods [2, 20] combine kernel with optical flow, and DAIN [2] also adopts the depth information to optimize optical flow. Most methods use typical flow nets [9, 27] to get the adjacent flow, and then synthesis the flow to the middle frame by interpolation, which induces huge computation costs and may bring in errors. Thus, we adopt a lightweight optical flow network [31] on LR frames and a flow refine network [26] to get the middle flow on HR frames, and we try a new supervised flow loss to achieve better perception. Recently, meta-learning is also introduced into frame interpolation [7]; CAIN [8] adapts channel attention into VFI; and EDSC [6] uses ConvLSTM to learn motion offset for implicit motion compensation.

2.3 Space-Time Video Super-Resolution

STVSR was firstly proposed to extend SR to the space-time domain in [24]. Recent studies [13, 33] show that single-stage STVSR can be better than most two-stage concatenations of independent VSR and VFI nets. Zooming-low-mo [33] used deformable ConvLSTM to combine local context information with motion offset. FISR [19] put forward a multi-scale temporal loss to make use of the alignment information between several frames based on flow and warping. STVUN [18] fuse features from different frames without explicit motion information compensation, but its feature interpolation still relies on warping input frames by optical flow calculated from PWC-Net [27]. STARnet [13] (Fig.1(e)), similarly to our work, achieved joint learning of VSR and VFI via iterations between multi-scale features from adjacent frames. It also adopts PWCNet [27] for flow estimation. However, our network is based on the iterations between VSR and VFI at the pixel level, which does not need many iterations and can be conveniently combined with most existing independent VSR and VFI networks.

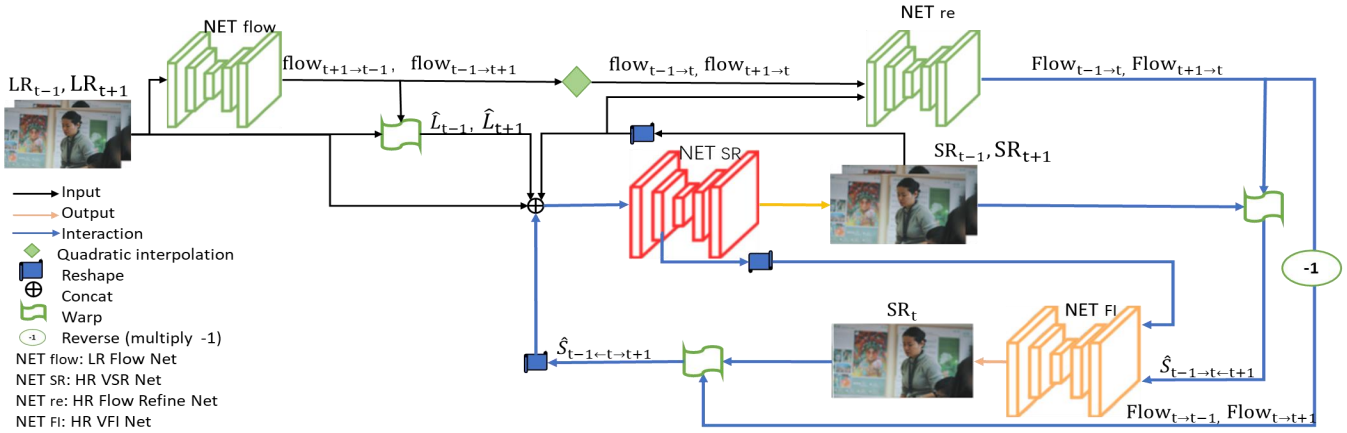

Figure 2: Diagram of our MBnet. It contains four sub-networks: NET_{flow}, NET_{SR}, NET_{re} and NET_{FI}. It uses odd LR frames LR_{t-1} and LR_{t+1} as input to generate the odd HR frames SR_{t-1} and SR_{t+1} (NET_{SR}) and the even HR frame SR_t (NET_{FI}) between them as output. It designs an interaction structure to exploit the inter-dependence between VSR and VFI to mutually benefit them and thus improve STVSR.

3 PROPOSED METHOD

3.1 Overall Network Architecture

In this paper we propose a new network called MBnet (Fig.2) for STVSR. We implement MBnet by designing a weighted shared interaction structure to better meld spatial and temporal information between VSR and VFI coarse-to-fine. Previous studies [13] show that doing VSR before VFI can achieve better results than the other way around. Hence, we feed the results of the VSR net to the VFI net, and feedback the interpolated HR frames to the VSR net to refine the results, and we adopt a lightweight optical flow network OFRnet [31] on LR frames and a flow refinement network PACnet [26] on HR frames to substantially reduce computational costs.

Taking two LR frames as input and generating three HR frames as an example, when given LR and LFR video frame sequence LR_{t-1} and LR_{t+1} with size of $h \times w$, our goal is to generate HR frames SR_{t-1}, SR_t and SR_{t+1} with size of $(h \times scale) \times (w \times scale)$ with up-sampling scale, *e.g.* scale = 4. Specifically, as shown in Fig.2, our MBnet consists of four sub-networks: NET_{flow} (Sec.3.2) for LR optical flow estimation between input LR frames LR_{t-1} and LR_{t+1}; NET_{re} (Sec.3.2) for HR flow refinement of HR frames from adjacent frames SR_{t-1} and SR_{t+1} to middle frame SR_t; NET_{SR} (Sec.3.3) for VSR to generate HR frames SR_{t-1} and SR_{t+1}; and NET_{FI} (Sec.3.4) for VFI to get HR middle frame SR_t.

3.2 LR Flow Net: NET_{flow} and HR Flow Refinement Net: NET_{re}

NET_{flow} is designed to estimate the bidirectional optical flow from two input LR frames. Typical optical flow networks usually need a huge number of parameters to estimate dense flow maps, *e.g.* PWCNet [27] has more than 8M parameters, while our NET_{flow} has only 0.76M parameters. Considering the efficiency requirement by STVSR, we adopt the light weight flow net OFRnet. OFRnet is a net for coarse-to-fine multi-scale optical flow estimation with shared

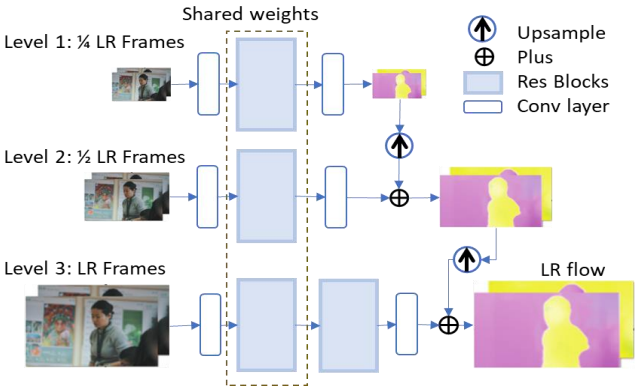

Figure 3: Structure of LR Flow Net NET_{flow}. The input frames are downsampled by 4 to get a coarse flow, and then the coarse flow will be refined in our multiscale pyramid structure step by step. The output flows of three levels will be restrained by multi-scale loss $\mathcal{L}^{flow}$ in Sec.3.5, thus our NET_{flow} can deal with large displacements with no more than 1/10 parameters of typical PWCnet [27].

weights (as shown in Fig.3), which can deal with large displacements and complex scenes while keeping lightweight.

NET_{flow} estimates the bidirectional dense motion flow maps $flow_{t-1 \leftrightarrow t+1}$ of size $4(channel) \times h \times w$ between LR frames LR_{t-1} and LR_{t+1}:

$$flow_{t-1 \leftrightarrow t+1} = NET_{flow}[LR_{t-1}, LR_{t+1}]. \quad (1)$$

Here $flow_{t-1 \leftrightarrow t+1}$ includes $flow_{t-1 \rightarrow t+1}$ ($2 \times h \times w$ flow from frame LR_{t-1} to LR_{t+1}) and $flow_{t-1 \leftarrow t+1}$ ($2 \times h \times w$ flow from frame LR_{t+1} to LR_{t-1}).

The LR flows are used to warp corresponding adjacent LR frames for VSR. However, VFI need HR flows from adjacent frames to the

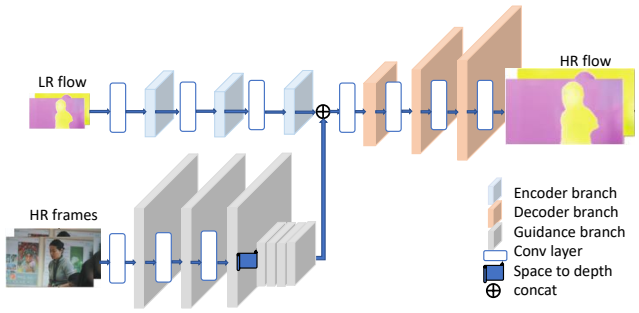

Figure 4: Detailed structure of Flow Refinement Net NET_{re}. Apart from the encoder and decoder branch, we add a guidance branch to introduce more detailed information from HR frames to refine the HR flow.

middle frame. Re-computing HR optical flow from scratch requires huge computational costs, and existent optical flow network needs HR middle frame S_t for calculation. Previous work use quadratic interpolation [16, 22] or Flow Projection Layer [2, 3] to get the middle flow, however, these methods will bring in artifacts and errors especially at the edge of objects. Simply up-sampling LR flow maps and interpolating to the middle frame can hardly generate accurate HR optical flow. Considering these problems, we design a NET_{re} to get and refine HR flows to the middle frame after quadratic interpolation.

Given bidirectional LR optical flows $flow_{t-1\leftrightarrow t+1}$ from Eq.1, we first use quadratic interpolation of the bidirectional LR flows to obtain $flow_{t-1\rightarrow t}$ (flow from frame $t-1$ to frame t) and $flow_{t\leftarrow t+1}$ (flow from frame $t+1$ to frame t):

$$flow_{t-1\rightarrow t} = (1-\widetilde{t})^2 flow_{t-1\rightarrow t+1} - \widetilde{t}(1-\widetilde{t}) flow_{t-1\leftarrow t+1}, \quad (2)$$

$$flow_{t\leftarrow t+1} = -\widetilde{t}(1-\widetilde{t}) flow_{t-1\rightarrow t+1} + \widetilde{t}^2 flow_{t-1\leftarrow t+1}. \quad (3)$$

Here $\widetilde{t}$ is the interpolation time and $\widetilde{t}$=0.5 for the middle frame t.

Then by using HR frames $SR_{t\pm1}$ (concatenation of SR_{t-1} and SR_{t+1} from Eq.5) in VSR as guidance, our Flow Refinement Net NET_{re} is aimed to refine and up-sample the LR flow $flow_{t-1\rightarrow t}$ from Eq.2 and $flow_{t\leftarrow t+1}$ from Eq.3 to get bidirectional HR dense flow maps $Flow_{t-1\rightarrow t}$ (flow from frame SR_{t-1} to SR_t) and $Flow_{t\leftarrow t+1}$ (flow from frame SR_t to SR_{t-1}):

$$[Flow_{t-1\rightarrow t}, Flow_{t\leftarrow t+1}] = NET_{re}[flow_{t-1\rightarrow t}, flow_{t\leftarrow t+1}, SR_{t\pm1}]. \quad (4)$$

Our Flow Refinement Net NET_{re} is a small encoder-decoder net with extra guidance branch, which extracts information from HR images to guide input LR flow to generate HR optical flow. The detailed structure is showed in Fig.4. Moreover, we try a new loss to get better flows to help VFI produce better perception results, and the details of new loss is explained in Sec.3.5.

3.3 LR VSR Net: NET_{SR}

NET_{SR} uses odd LR frames $LR_{t\pm1}$ (concatenation of LR_{t-1} and LR_{t+1}) and their warped LR frames $\hat{LR}_{t\mp1\rightarrow t\pm1}$, the HR frames $SR_{t\pm1}^{(i-1)}$ from NET_{SR} of former iteration and warped HR frames $\hat{SR}_{t\rightarrow t\pm1}$ to generate corresponding odd HR frames $SR_{t+1}^{(i)}$ and $SR_{t-1}^{(i)}$ (in the first iteration $i=1$, HR frames are simply the bilinear up-samples of LR frames):

$$[SR_{t+1}^{(i)}, SR_{t-1}^{(i)}] = NET_{SR}[LR_{t\pm1}, SR_{t\pm1}^{(i-1)}, \hat{LR}_{t\mp1\rightarrow t\pm1}, \hat{SR}_{t\rightarrow t\pm1}^{(i-1)}], \quad (5)$$

where warped LR frames $\hat{LR}_{t\mp1\rightarrow t\pm1}$ is the concatenation of $\hat{LR}_{t-1\rightarrow t+1}$ and $\hat{LR}_{t-1\leftarrow t+1}$, with $\hat{LR}_{t-1\rightarrow t+1} = \text{warp}(LR_{t-1}, flow_{t-1\rightarrow t+1})$ (to warp LR_{t-1} to time t+1) and $\hat{LR}_{t-1\leftarrow t+1} = \text{warp}(LR_{t+1}, flow_{t-1\leftarrow t+1})$ (to warp LR_{t+1} to time $t-1$). Here function *warp* means moving all pixels of adjacent LR frames alongside the direction of flow from Eq.1 to align to the current frame. Differently, warped HR frames $\hat{SR}_{t\rightarrow t\pm1}$ is the concatenation of $\hat{SR}_{t-1\leftarrow t}$ and $\hat{SR}_{t\rightarrow t+1}$ and then reshaped into LR by space to depth, and $\hat{SR}_{t-1\leftarrow t} = \text{warp}(SR_t, Flow_{t-1\leftarrow t})$ (to warp SR_t to time $t-1$), $\hat{SR}_{t\rightarrow t+1} = \text{warp}(SR_t, Flow_{t\rightarrow t+1})$ (to warp SR_t to time $t+1$). In order to warp HR frames from Eq.6 in VFI back to VSR, we reverse the direction of HR flow in Eq.4 to approximate the optical flow from time t to time $t-1$ and time $t+1$. Specifically, we warp the result of VFI back to the adjacent frames to refine the HR frames, because the feedback of VFI can alleviate the spacial relevance degeneration of VSR by temporal alignment. Compared with feeding back the warped frames of VSR, our refined flow brings in reliable warped frames of VFI, which are closer to the adjacent frames and have more relevant information, thus the VSR results are improved, especially when there is large displacement or fine texture. We only iterate twice to mutually benefit VSR and VFI, and more iterations cannot bring obvious further improvement. Finally, we get all the output SR frames of VSR net and VFI net ($SR_{t-1}^{(2)}$, $SR_t^{(2)}$, $SR_{t+1}^{(2)}$) as the output of the whole network.

In our baseline, we simply adopt RDNnet [36] for NET_{SR}, whose numbers of residual dense blocks and layers in each block can be adjusted to control the depth and parameters of the network, as complex scenes usually require deeper and more complex networks. This module can be replaced by another VSR net, *e.g.* RBPN [12]; that is, our structure could be easily combined with other VSR nets.

3.4 HR VFI Net: NET_{FI}

Previous work [2, 22] has indicated that adding features from a pretrained network can help the VFI net to better reconstruct the middle frame, because only using adjacent frames will result in context degeneration in VFI. However, their pretrained feature extraction network comes from other semantic tasks (for example, semantic segmentation). Unlike them, VSR and VFI are both related to pixels generation with tighter feature correlation, thus features from VSR can better help VFI. We also use HR frames from the VSR net to refine the HR flow of the middle frame, which could help the VFI net to reduce artifacts through more precise flow estimation and temporal alignment.

Apart from the feature from VSR, we feed the warped frames of the adjacent HR frames $\hat{SR}_{t\pm1\rightarrow t}$ and the reshaped VSR feature into a VFI net NET_{FI} to synthesize the even middle frame between them at time t:

$$SR_t^{(i)} = NET_{FI}[\hat{SR}_{t\pm1\rightarrow t}, feature], \quad (6)$$

where $\hat{SR}_{t\pm1\rightarrow t}$ is the concatenation of warped HR frames $\hat{SR}_{t-1\rightarrow t}$ and $\hat{SR}_{t\leftarrow t+1}$. Specifically, $\hat{SR}_{t-1\rightarrow t} = \text{warp}(SR_{t-1}, Flow_{t-1\rightarrow t})$ (to warp HR frame SR_{t-1} to the time t); $\hat{SR}_{t\leftarrow t+1} = \text{warp}(SR_{t+1}, Flow_{t\leftarrow t+1})$

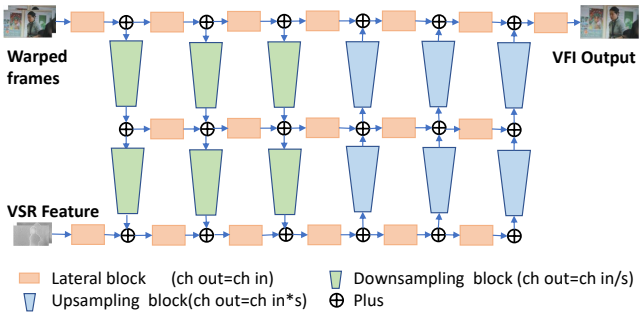

Figure 5: Structure of HR VFI Net NET_{FI}. Each block consists of two convolution layers followed by two ReLU layers with different input and output channels, and another bilinear up-sample layer in the upsampling block. For spacial SR scale = 4, we set s = scale / 2 in the downsampling and upsampling blocks.

(to warp HR frame SR_{t+1} to time t); that is, HR frames from VSR (Eq.5) are warped alongside the direction of flow from Eq.4 to align to the middle frame. The context feature *feature* is the output of the second convolution layer in the VSR net NET_{SR}. It provides more detailed context information for the VFI net, because the feature of VSR is strongly associated with the reconstruction of pixels and can remedy the lack of detailed context and edge information in VFI, thus alleviating blur and producing more sharp-edged frames. On the other hand, the result of VFI SR_t will be fed back to VSR (Eq.5) for one more time to help VSR.

Considering temporal alignment of motion, we use a variant of U-Net, GridNet [10] of three rows and six columns, for interpolation frame synthesis. In order to fully use the context information from the VSR net to better deal with large motion and complex scene, we change the input of each row in GridNet into multi-scale varying from LR to HR. The detailed structure is shown in Fig.5. This part is also flexible: it can be enlarged or replaced by other VFI nets to fit for user purposes in practice.

3.5 Loss Functions

Our MBnet includes four sub-networks, and we train them jointly end to end. Hence, we need to consider three kinds of loss functions: loss $\mathcal{LR}^{SR}$ for frame reconstruction, unsupervised loss $\mathcal{LR}^{flow}$ and supervised loss $\mathcal{LR}^{FLOW}$ for optical flow.

For the supervised loss, we adopt the L1 loss between the HR ground truth frames $\{HR_t\}_{t=1}^{2T+1}$ and our predicted SR frames $\{SR_t\}_{t=1}^{2T+1}$, in which T frames are interpolated by VFI and $T+1$ frames are generated by VSR (in our net $T=1$):

$$\mathcal{LR}^{SR} = \sum_{t=1}^{2T+1} \|SR_t - HR_t\|_1. \tag{7}$$

For the unsupervised loss of the optical flow of LR frames and the refined flow of SR frames, we calculate $\mathcal{L}^{flow}$ as

$$\mathcal{L}^{flow} = \sum_{t=1}^{2T+1} \frac{\lambda_1 \mathcal{L}_t^{SR} + \lambda_2 \mathcal{L}_t^{LR}}{T}, \tag{8}$$

where the SR flow loss $\mathcal{L}_t^{SR}$ is

$$\mathcal{L}_t^{SR} = \|\text{warp}(SR_{t-1}, Flow_{t-1\to t+1}) - SR_{t+1}\|_1 + 0.1\|\Delta Flow_{t-1\to t+1}\|_1,$$

and the LR flow loss $\mathcal{L}_t^{LR}$ is

$$\mathcal{L}_t^{LR} = \|\text{warp}(LR_{t-1}, flow_{t-1\to t+1}) - LR_{t+1}\|_1 + 0.1\|\Delta flow_{t-1\to t+1}\|_1,$$

here LR flows of three levels (Fig3) will all be restrained step by step with corresponding downsampling frames. The $\|\Delta Flow_{t-1\to t+1}\|_1$ and $\|\Delta flow_{t-1\to t+1}\|_1$ denote the L1 regularization term for the smoothness of optical flow, and Δ means finite difference of flow field.

In addition, we try a new supervised flow loss $\mathcal{L}^{FLOW}$ during training. We use the pre-trained optical flow network RAFT [28] to calculate the flow of ground truth frames $HF_{t-1\to t}$ (flow from H_{t-1} to H_t) and $HF_{t\leftarrow t+1}$ (flow from H_{t+1} to H_t),

$$HF_{t-1\to t} = \text{RAFT}(HR_{t-1}, HR_t),\ HF_{t\leftarrow t+1} = \text{RAFT}(HR_{t+1}, HR_t), \tag{9}$$

and then use L1 loss to supervise our HR flow $Flow_{t-1\to t}$ and $Flow_{t\leftarrow t+1}$:

$$\mathcal{L}^{FLOW} = \|Flow_{t-1\to t} - HF_{t-1\to t}\|_1 + \|Flow_{t\leftarrow t+1} - HF_{t\leftarrow t+1}\|_1. \tag{10}$$

Hence the total loss is

$$\mathcal{L} = \mathcal{L}^{SR} + \lambda_a \mathcal{L}^{flow} + \lambda_b \mathcal{L}^{FLOW}. \tag{11}$$

4 EXPERIMENTS AND ANALYSIS

4.1 Experimental Setup

We use four publicly available datasets, Vimoe90K [34], Middlebury [1], UCF101 [25] and Vid4 [21], in the experiments. We set $\lambda_1 = 1.0$, $\lambda_2 = 0.5$, $\lambda_a = \lambda_b = 0.01$ to balance different losses (detailed experiments are shown in Table 5). We calculate the Peak Signal-to-Noise Ratio (PSNR), the Structural Similarity Index (SSIM) and Natural Image Quality Evaluator (NIQE) on each test set for evaluation.

In order to compare with other video enhancement methods, we focus on S×4 SR factor and T×2 interpolation in temporal domain (scale = 4, $\widetilde{t} = 0.5$). For our baseline, we use 8 dense blocks with 8 layers for each block if no extra notice. The input frames are cropped into patches of size 32×32 and augmented by rotating and flipping randomly (same as in [33]). The Adam optimizer is used for training with an initial learning rate of 1e-4, and the batch size is set to 16 on Nvidia 1080Ti.

4.2 Ablation Studies

In the ablation studies, we train our baseline on the training set of Vimeo-90K [34], and then test them on the same test folder 00001 of Vimeo-90K in the ablation studies, which includes 58 different scenes and totally 406 frames.

Effect of interaction. We set VSR first and VFI second without feedback as the control group without interaction. In order to eliminate the influence of network depth, we also double the numbers of RDN blocks (D) and layers of each block (C), whose depth and parameters are also doubled, and retrain our networks with different sizes and test them on the same test set. The results in Table 2 indicate that the feedback is beneficial to both VSR and VFI in all networks of different sizes, and the PSNR of nets with feedback is

Table 1: Ablation studies on different features (VSR or ResNet) for VFI, on w/ and w/o feedback for VSR, on different flow nets, on different VSR nets and on different VFI nets. The best results are shown in bold. VSR feature and VFI feedback help a lot in our MBnet, which indicates that VSR and VFI can mutually benefit.

	VSR feature	ResNet feature	VFI feedback	OFR net	PWC net	RDN net	RBPN net	GridNet	Unet	VSR PSNR	VSR SSIM	VFI PSNR	VFI SSIM	Average PSNR	Average SSIM
1	√	×	√	√	×	√	×	√	×	**33.48**	**0.922**	**30.98**	**0.906**	**32.41**	**0.915**
2	×	√	√	√	×	√	×	√	×	33.24	0.919	30.72	0.902	32.16	0.912
3	×	×	√	√	×	√	×	√	×	33.04	0.919	30.56	0.901	31.98	0.911
4	√	×	×	√	×	√	×	√	×	33.18	0.918	30.88	0.904	32.20	0.912
5	√	×	√	×	√	√	×	√	×	33.23	0.919	30.83	0.903	32.20	0.912
6	√	×	×	√	×	×	√	√	×	32.64	0.911	30.15	0.894	31.57	0.904
7	√	×	√	√	×	×	√	√	×	33.13	0.918	30.27	0.897	31.91	0.909
8	√	×	√	√	×	√	×	×	√	33.46	0.922	24.16	0.775	29.53	0.859

even higher than doubled layers nets without feedback. We also compare the convergence with or without interaction when trained from scratch. As shown in Fig.6, using interaction, from which more effective information can be extracted, would lead to higher performance, and the convergence of net with interaction is faster and more stable. All these results suggest that VSR and VFI can benefit from their interaction through our feedback structure.

Table 2: Ablation studies on the benefit of interaction (red for the best results and blue for the second best). 'D' denotes the number of RDN blocks and 'C' denotes the layer number in each block. The models with interaction outperforms a lot those models without interaction.

	VSR PSNR	VFI PSNR	Average PSNR
D8C4 w/o interaction	32.29	29.88	31.26
D8C4 with interaction	32.78	30.26	31.70
D8C8 w/o interaction	32.53	30.06	31.47
D8C8 with interaction	33.25	30.74	32.17
D16C8 w/o interaction	33.02	30.31	31.85
D16C8 with interaction	33.87	31.17	32.72

Mutual benefit. Previous studies [2, 22] extract context feature to help VFI through a pre-trained ResNet, while our MBnet collects the output of the second convolution layer in the VSR net as context feature and feeds it into the VFI net. From the results in Table 1, we can make the following observations. First, the first three rows of results show that, if we replace our VSR features with the features from an extra ResNet as [2], the results (row 2) are better than using no features (row 3) but worse than using our VSR features (row 1). This indicates our features from VSR can provide useful information for VFI than those from an independent feature extraction net; that is, VFI can benefit from VSR to get more context details.

Secondly, if we warp the HR results $S_{t\pm1}$ of VSR itself instead of the VFI result S_t back to VSR (row4, row6), the last two rows of results show that the PSNR of VSR declines about 0.3dB compared with the net with the VFI feedback. This indicates that VFI can provide more accurate motion information to VSR. This can be attributed to the fact that the distance from frame t to frame $t-1$ or frame $t+1$ is shorter than that from frame $t-1$ to frame $t+1$,

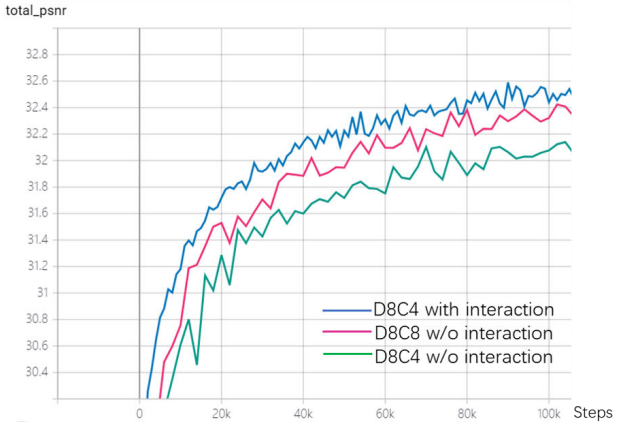

Figure 6: Convergence studies on w/ or w/o interaction. Interaction mechanism leads to not only higher performance but also faster and more stable convergence during training.

thus smaller motion and more similar pixels between frames could provide more information to reconstruct the HR frames.

Effect of LR Flow Net NET_{flow}. We compare our NET_{flow} OFRnet with the typical flow network PWCNet [27]. We replace our OFRnet with a pre-trained PWCNet and fine-tune it with the loss $\mathcal{L}_t^{LR}$ end to end. The STVSR result of PWCNet is worse, because PWCNet is based on synthetic datasets *e.g.* Flying Chairs [9] and Sintel [5], thus it performs not well when handing complex motion in real world. Compared with fine-tuning pre-trained models, initializing our flow net from scratch has a great advantage over PWCNet in terms of both performance and parameter amount.

Effect of LR VSR Net NET_{SR}. We replace the RDN blocks in our NET_{SR} with RBPN blocks [12]. For fair comparison, we change the RBPN to two LR frames as input and two HR frames as output and retrain the whole networks from scratch. The results of VSR and VFI without interaction (in row6) are worse than RDN blocks in row4, then we introduce the feedback mechanism into the network and the results of VSR clearly improves with 0.49 PSNR(in row7), and the frame of VFI also benefits from interaction, that is, our interaction structure can be applied to other VSR methods and bring distinct improvement.

Table 3: Comparison with other methods (S×4, T×2).We use red for the best results and blue for the second best.

Method	UCF101			Middlebury-other			Vid4		Vimeo90K			VSR	VFI	Parameters
	PSNR	SSIM	NIQE	PSNR	SSIM	NIQE	PSNR	SSIM	PSNR	SSIM	NIQE	PSNR	PSNR	(Million)
DBPN [11]+ToFlow[34]	28.112	0.902	8.630	26.012	0.808	5.901	-	-	29.867	0.915	7.120	-	-	10.5+1.1
DBPN [11]+DAIN [2]	28.175	0.902	8.755	26.268	0.809	5.869	-	-	30.021	0.918	7.223	-	-	10.5+24.0
DAIN [2]+RBPN [12]	27.631	0.909	8.932	25.744	0.811	5.814	-	-	29.422	0.916	7.253	-	-	24.0+12.7
RBPN [12]+DAIN [2]	28.729	0.919	8.769	26.766	0.821	5.522	-	-	30.455	0.926	7.081	-	-	12.7+24.0
RBPN+DAIN-joint	28.856	0.920	8.799	26.923	0.823	5.444	-	-	30.623	0.927	7.183	-	-	36.7
TDAN [29]+AdaCoF [20]	30.515	0.895	8.572	28.859	0.852	5.361	24.593	0.788	32.582	0.923	7.132	33.866	30.869	-
STARnet [13](SOTA)	29.111	0.924	8.787	27.115	0.827	5.423	-	-	30.830	0.929	7.154	32.349	30.704	111.6
Zooming-slo-mo [33](SOTA)	30.733	0.911	8.805	28.396	0.855	5.375	24.418	0.775	32.919	0.926	7.165	34.213	31.194	11.1
Our MBnet	30.852	0.937	8.739	29.045	0.857	5.359	24.505	0.883	33.048	0.928	7.077	34.361	31.297	30.8

Effect of HR VFI Net NET_{FI}. We also replace the GridNet of our NET_{FI} with a Unet of three layers, and the PSNR and SSIM of VFI results both decline a lot (row8). The GridNet structure has a big enough receptive field and multi-stream to well exploit the information from different scales, which is important for VFI.

Figure 7: Visual comparison on w/ and w/o $\mathcal{L}^{FLOW}$. Our supervised flow loss $\mathcal{L}^{FLOW}$ could effectively alleviate the artifacts at the edge of texture.

Table 4: Ablation studies on the flow loss (red for the best results and blue for the second best). Our flow loss $\mathcal{L}^{FLOW}$ leads to a little decline on PSNR (dB) and SSIM, but it improves the image quality with reduced NIQE.

loss	PSNR ↑	SSIM ↑	NIQE ↓
$\mathcal{L}^{SR}$	30.757	0.934	8.714
$\mathcal{L}^{SR} + 0.01\mathcal{L}^{flow}$	30.852	0.937	8.739
$\mathcal{L}^{SR} + 0.01\mathcal{L}^{flow} + 0.01\mathcal{L}^{FLOW}$	30.772	0.936	8.692

Effect of supervised flow loss $\mathcal{L}^{FLOW}$. All the models are tested on the UCF101 dataset. As shown in Table 4, our proposed supervised flow loss $\mathcal{L}^{FLOW}$ could improve the image quality with the lowest NIQE score. Although it brings in a decline in PSNR and SSIM compared with $\mathcal{L}^{SR} + 0.01\mathcal{L}^{flow}$, its results are still better than using L1 loss $\mathcal{L}^{SR}$ only. The visual comparison is illustrated in Fig.7: our proposed supervised flow loss $\mathcal{L}^{FLOW}$ could effectively alleviate the artifacts at the edge of texture. Because $\mathcal{L}^{FLOW}$ works on the flow net to get a more accurate flow estimation for motion compensation instead of supervising the final frame reconstruction directly, it leads to better perceptional experience rather than higher PSNR scores.

Table 5: Ablation studies on the loss weight (red for the best results and blue for the second best). The λ_1 and λ_2 are weights of HR and LR optical flows, and smaller LR optical flow weights λ_2 could improve the results of VFI. λ_a and λ_b are the weights of unsupervised and supervised optical flow losses. Too small flow loss weights bring severe artifacts on VFI frames, and too large flow loss weights result in more unstable training and slower convergence.

λ_1	λ_2	λ_a	λ_b	VSR PSNR	VFI PSNR	Average PSNR
1.0	0.5	0.01	0.01	33.73	30.94	32.80
1.0	1.0	0.01	0.01	33.8	30.76	32.79
1.0	0.5	0.001	0.001	33.79	30.66	32.75
1.0	0.5	0.1	0.1	33.76	30.72	32.75
1.0	0.5	1.0	1.0	32.79	30.74	32.11

Effect of different loss weights. Because the HR flow has bigger influence than LR flow especially on the middle frame from VFI, we set $\lambda_1 = 1.0$, $\lambda_2 = 0.5$. As shown in Table 5, the coarser LR flow could be better refined to HR with smaller LR optical flow weights λ_2, thus smaller LR optical flow weights could improve the results of VFI because of better HR flow estimation to the middle fame, although result in a little decline on VSR frames. The initial flow loss is almost 100 times bigger than the pixels loss, so the loss balance weights are set to $\lambda_a = \lambda_b = 0.01$. In detail, if the weights of unsupervised optical flow and supervised optical flow, λ_a and λ_b, are nearly same or larger than as the pixels loss of frames, the training becomes more unstable and the network converges more slowly, and, on the other hand, too small optical flow loss weights will bring severe artifacts on the VFI results.

4.3 Comparison with state-of-the-arts

We compare our MBnet with other end-to-end STVSR methods (Zooming-slo-mo [33] and STARnet [13]) and some combinations of VSR and VFI nets, *e.g.* TDAN [29] + AdaCoF [20]. Table 3 shows the quantitative results of S×4 and T×2 upsampling. For STARnet, we use the STAR-ST-Lr version from the paper [13], because we use the L1 loss to reconstruct frames only without the feature loss $\mathcal{L}^{vgg}$

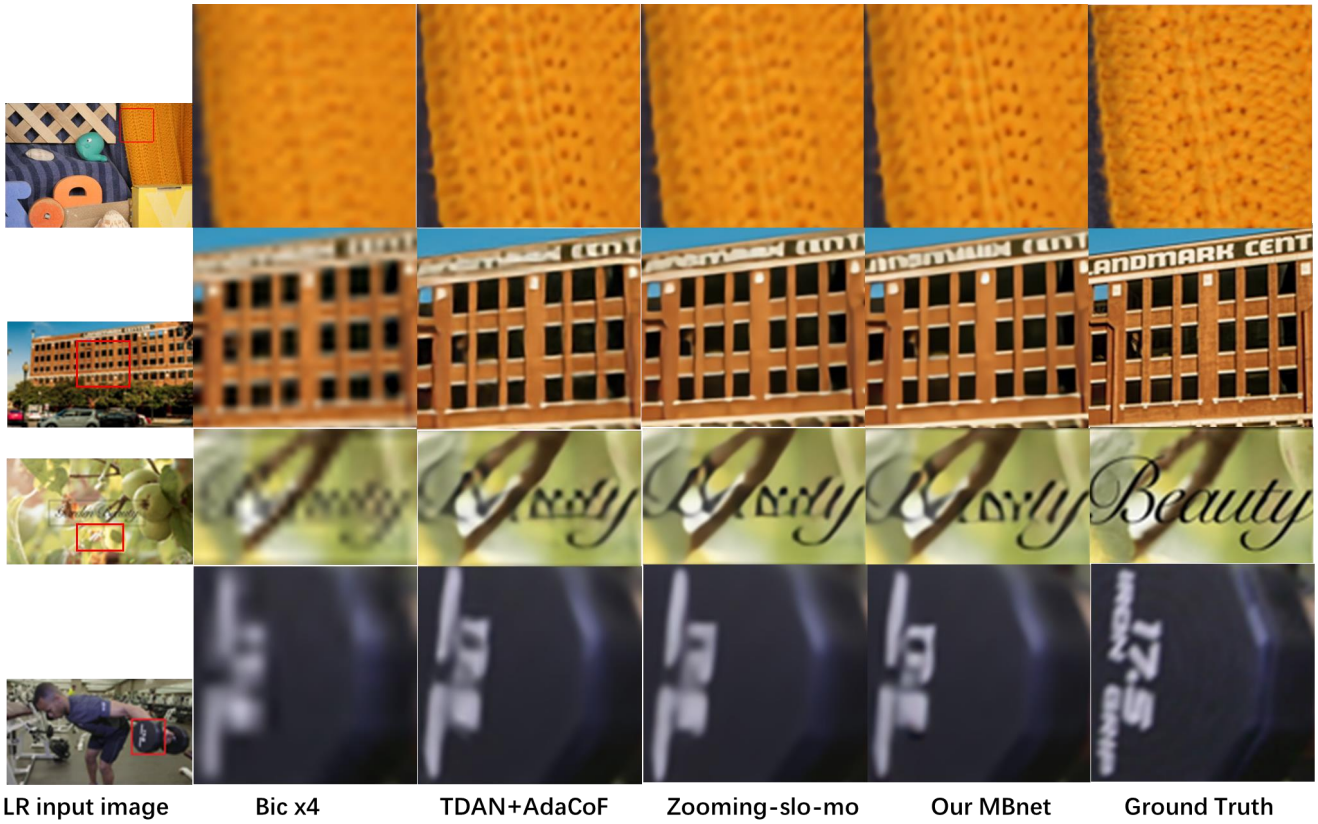

Figure 8: Results of S×4 and T×2 upsampling on Middlebury-other and Vimeo 90K (frame04 of RubberWhale, img2 of 00076/0172, img3 of 00076/0171, and im4 of 00006/0808). Our MBnet effectively mitigates motion blur, especially in the areas near the letters.

in [13]. For TDAN, AdaCoF and Zooming-so-mo, we retest them on the same test set with their pre-trained models, and the input frame number of Zooming-slo-mo is set to two for fair comparison.

The PSNR of our MBnet performs the best on most datasets. In addition, our MBnet outperforms the Zooming-slo-mo by 0.13dB on Vimeo-90K on average, and provides 0.15dB improvement on VSR and 0.1dB on VFI. Compared to STARnet, we achieve competitive results on all datasets with about 60% fewer parameters. This is mainly benefited from our two light-weighted flow nets, LR flow net NET_{flow} and HR flow refinement net NET_{re}. We observe that joint fine-tuning of VSR and VFI nets, *e.g.* RBPN+DAIN, could improve the final results but the PSNR is promoted not more than 0.2dB. The combination of state-of-the-art VSR and VFI networks, TDAN and AdaCoF, are better than the end-to-end STARnet and comparable to the state-of-the-art STVSR method Zooming-slo-mo on most datasets, but Zooming-slo-mo has limited ability in dealing with details, for example, the texture of orange sweater as shown in Fig.8 (the first row). Moreover, it will induce severe artifacts and motion blur (lower rows in Fig.8). Our MBnet can better reconstruct the context details with more accurate structures and alleviate motion blur with fewer artifacts, through the effective interactions between VSR and VFI.

5 CONCLUSION

This study focuses on the joint enhancement of VSR and VFI based on weighted shared interaction structure. We find that the VSR feature is beneficial to VFI and the VFI feedback can also provide more motion information for VSR. Thanks to such a simple strategy, our approach achieves state-of-the-art performance on various datasets and is universally applicable to embrace any existing VSR or VFI networks for effectively improving their video enhancement performance. In response to the difficulty and huge computation cost of flow estimation, we build two light flow nets and adopt a coarse-to-fine refinement strategy. In addition, we put forward a new supervised loss of flow, which adopts a latest optical flow network to get the referential flow for supervision and achieves better perceptional experience. In the future, we will further investigate our model by combining it with more VSR and VFI lightweight networks and considering higher up-sampling scales in the space and time domains.

ACKNOWLEDGMENTS

CZ thanks Tencent Video Cloud and Huawei. This work was partly supported by the Special Foundation for the Development of Strategic Emerging Industries of Shenzhen (JCYJ20170817161056260).

REFERENCES

[1] Simon Baker, Daniel Scharstein, JP Lewis, Stefan Roth, Michael J Black, and Richard Szeliski. 2011. A database and evaluation methodology for optical flow. *International Journal of Computer Vision* 92, 1 (2011), 1–31.

[2] Wenbo Bao, Wei-Sheng Lai, Chao Ma, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. 2019. Depth-aware video frame interpolation. In *CVPR*. 3703–3712.

[3] Wenbo Bao, Wei-Sheng Lai, Xiaoyun Zhang, Zhiyong Gao, and Ming-Hsuan Yang. 2021. MEMC-Net: Motion estimation and motion compensation driven neural network for video interpolation and enhancement. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 43, 3 (2021), 933–948.

[4] Wenbo Bao, Xiaoyun Zhang, Shangpeng Yan, and Zhiyong Gao. 2017. Iterative convolutional neural network for noisy image super-resolution. In *ICIP*. 4038–4042.

[5] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. 2012. A naturalistic open source movie for optical flow evaluation. In *ECCV*. 611–625.

[6] Xianhang Cheng and Zhenzhong Chen. 2020. Video Frame Interpolation via Deformable Separable Convolution.. In *AAAI*. 10607–10614.

[7] Myungsub Choi, Janghoon Choi, Sungyong Baik, Tae Hyun Kim, and Kyoung Mu Lee. 2020. Scene-Adaptive Video Frame Interpolation via Meta-Learning. In *CVPR*. 9444–9453.

[8] Myungsub Choi, Heewon Kim, Bohyung Han, Ning Xu, and Kyoung Mu Lee. 2020. Channel Attention Is All You Need for Video Frame Interpolation.. In *AAAI*. 10663–10671.

[9] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. 2015. FlowNet: Learning optical flow with convolutional networks. In *CVPR*. 2758–2766.

[10] Damien Fourure, Rémi Emonet, Elisa Fromont, Damien Muselet, Alain Tremeau, and Christian Wolf. 2017. Residual conv-deconv grid network for semantic segmentation. *arXiv preprint arXiv:1707.07958* (2017).

[11] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. 2018. Deep back-projection networks for super-resolution. In *CVPR*. 1664–1673.

[12] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. 2019. Recurrent back-projection network for video super-resolution. In *CVPR*. 3897–3906.

[13] Muhammad Haris, Greg Shakhnarovich, and Norimichi Ukita. 2020. Space-Time-Aware Multi-Resolution Video Enhancement. In *CVPR*. 2859–2868.

[14] Zhewei Huang, Tianyuan Zhang, Wen Heng, Boxin Shi, and Shuchang Zhou. 2020. RIFE: Real-Time Intermediate Flow Estimation for Video Frame Interpolation. *arXiv preprint arXiv:2011.06294* (2020).

[15] Takashi Isobe, Songjiang Li, Xu Jia, Shanxin Yuan, Gregory Slabaugh, Chunjing Xu, Ya-Li Li, Shengjin Wang, and Qi Tian. 2020. Video super-resolution with temporal group attention. In *CVPR*. 8008–8017.

[16] Huaizu Jiang, Deqing Sun, Varun Jampani, Ming-Hsuan Yang, Erik Learned-Miller, and Jan Kautz. 2018. Super SloMo: High quality estimation of multiple intermediate frames for video interpolation. In *CVPR*. 9000–9008.

[17] Younghyun Jo, Seoung Wug Oh, Jaeyeon Kang, and Seon Joo Kim. 2018. Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In *CVPR*. 3224–3232.

[18] Jaeyeon Kang, Younghyun Jo, Seoung Wug Oh, Peter Vajda, and Seon Joo Kim. 2020. Deep Space-Time Video Upsampling Networks. *arXiv preprint arXiv:2004.02432* (2020).

[19] Soo Ye Kim, Jihyong Oh, and Munchurl Kim. 2020. FISR: Deep Joint Frame Interpolation and Super-Resolution with a Multi-Scale Temporal Loss.. In *AAAI*. 11278–11286.

[20] Hyeongmin Lee, Taeoh Kim, Tae-young Chung, Daehyun Pak, Yuseok Ban, and Sangyoun Lee. 2020. AdaCoF: Adaptive Collaboration of Flows for Video Frame Interpolation. In *CVPR*. 5316–5325.

[21] Ce Liu and Deqing Sun. 2011. A Bayesian approach to adaptive video super resolution. In *CVPR*. 209–216.

[22] Simon Niklaus and Feng Liu. 2018. Context-aware synthesis for video frame interpolation. In *CVPR*. 1701–1710.

[23] Simon Niklaus, Long Mai, and Feng Liu. 2017. Video frame interpolation via adaptive separable convolution. In *CVPR*. 261–270.

[24] Eli Shechtman, Yaron Caspi, and Michal Irani. 2002. Increasing space-time resolution in video. In *ECCV*. 753–768.

[25] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. 2012. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402* (2012).

[26] Hang Su, Varun Jampani, Deqing Sun, Orazio Gallo, Erik Learned-Miller, and Jan Kautz. 2019. Pixel-adaptive convolutional neural networks. In *CVPR*. 11166–11175.

[27] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. 2018. PWC-Net: CNNs for optical flow using pyramid, warping, and cost volume. In *CVPR*. 8934–8943.

[28] Zachary Teed and Jia Deng. 2020. RAFT: Recurrent all-pairs field transforms for optical flow. In *European Conference on Computer Vision*. 402–419.

[29] Yapeng Tian, Yulun Zhang, Yun Fu, and Chenliang Xu. 2020. TDAN: Temporally-Deformable Alignment Network for Video Super-Resolution. In *CVPR*. 3360–3369.

[30] Hua Wang, Dewei Su, Chuangchuang Liu, Longcun Jin, Xianfang Sun, and Xinyi Peng. 2019. Deformable Non-Local Network for Video Super-Resolution. *IEEE Access* 7 (2019), 177734–177744.

[31] Longguang Wang, Yulan Guo, Li Liu, Zaiping Lin, Xinpu Deng, and Wei An. 2020. Deep video super-resolution using HR optical flow estimation. *IEEE Transactions on Image Processing* 29 (2020), 4323–4336.

[32] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. 2019. EDVR: Video restoration with enhanced deformable convolutional networks. In *CVPRW*. 1954–1963.

[33] Xiaoyu Xiang, Yapeng Tian, Yulun Zhang, Yun Fu, Jan P Allebach, and Chenliang Xu. 2020. Zooming Slow-Mo: Fast and Accurate One-Stage Space-Time Video Super-Resolution. In *CVPR*. 3370–3379.

[34] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. 2019. Video enhancement with task-oriented flow. *International Journal of Computer Vision* 127, 8 (2019), 1106–1125.

[35] Peng Yi, Zhongyuan Wang, Kui Jiang, Junjun Jiang, and Jiayi Ma. 2019. Progressive fusion video super-resolution network via exploiting non-local spatio-temporal correlations. In *CVPR*. 3106–3115.

[36] Yulun Zhang, Yapeng Tian, Yu Kong, Bineng Zhong, and Yun Fu. 2018. Residual dense network for image super-resolution. In *CVPR*. 2472–2481.