J. Vis. Commun. Image R. 75 (2021) 103038

Contents lists available at ScienceDirect

Journal of Visual Communication and Image Representation

journal homepage: www.elsevier.com/locate/jvci

Real-time video super-resolution using lightweight depthwise separable group convolutions with channel shuffling☆

Zhijiao Xiao [a,1], Zhikai Zhang [a,1], Kwok-Wai Hung [b,*], Simon Lui [b]

[a] College of Computer Science and Software Engineering, Shenzhen University, China
[b] Tencent Music Entertainment, Shenzhen, China

ARTICLE INFO

Keywords:
Super-resolution
Lightweight alignment module
Channel shuffle
Residual networks

2000 MSC:
41A05
41A10
65D05
65D17

ABSTRACT

In recent years, convolutional neural networks (CNNs) have accelerated the developments of video super resolution (SR) for achieving higher image quality. However, the computational cost of existing CNN-based video super-resolution is too heavy for real-time applications. In this paper, we propose a new video super-resolution framework using lightweight frame alignment module and well-designed up-sampling module for real-time processing. Specifically, our framework, which is called as Lightweight Shuffle Video Super-Resolution Network (LSVSR), combines channel shuffling, depthwise convolution and pointwise group convolution to significantly reduce the computational burden during frame alignment and high-resolution frame reconstruction. On the public benchmark datasets, our proposed network outperforms the state-of-the-art lightweight video SR networks in terms of objective (PSNR and SSIM) and subjective evaluations, number of network parameters and floating-point operations. Our network can achieve real-time 540P to 2160P 4× super-resolution for more than 60fps using desktop GPUs or mobile phones with neural processing unit.

1. Introduction

Video super-resolution (VSR) is a hot topic of image processing and computer vision, of which its technology has been widely used in industry in recent years. In general, the VSR task aims at restoring high-resolution (HR) frame from its corresponding low-resolution (LR) frame. The main problem of VSR is how to use the correlated information between multiple consecutive frames. Moreover, the significant computation cost of deep convolution neural networks (CNNs) is required in existing approaches. If CNNs-based VSR model can achieve outstanding performance and requires a low computational cost, VSR can be further extended to many application scenarios. Recently, most of the state-of-the-art video super resolution methods are highly complex CNNs models [1–3]. These models have outstanding performance with high computational complexity, making them difficult to apply to real-world scenarios. In high-level vision tasks such as image classification and object recognition, some models with excellent performance and low complexity are proposed [4,5]. These approaches inspire us to reduce the computational complexity of the VSR model. In the VSR task, motion compensation plays an important role to the performance of the model. Several models [6,7] use traditional methods to calculate the optical flow field between adjacent frames, and complete the frame alignment by warping operations. Moreover, there are models [2,8] that use CNNs to replace traditional methods to accomplish optical flow estimation. However, no matter which method is used, estimating the optical flow will put high computational burden on the model. In addition, explicit motion compensation will often cause the problem of visual artifacts due to inaccurate motion estimation. Hence, a deep network is required to reduce the artifacts leading to high computation costs. In order to solve the problem of huge calculations of existing VSR models, we propose an end-to-end trainable model called Lightweight Shuffle Video Super-Resolution Network (LSVSR). In LSVSR, we use depthwise convolution and pointwise group convolution to factorizes the standard convolutions. Based on them, we designed a module called lightweight alignment module (LAM) that avoids explicit motion compensations. Hence, compared with other VSR methods that need to estimate optical flow, LSVSR greatly reduces the amount of model calculations. Moreover, a shallow restoration network is connected to the end of motion alignment for generating the final high-resolution frame. On the public benchmark datasets, LSVSR outperforms several

☆ This paper has been recommended for acceptance by Zicheng Liu.

* Corresponding author.
E-mail address: guoweihung@tencent.com (K.-W. Hung).

[1] Zhijiao Xiao and Zhikai Zhang contributes equally in this work.

https://doi.org/10.1016/j.jvcir.2021.103038
Received 8 June 2020; Received in revised form 13 November 2020; Accepted 16 January 2021
Available online 10 February 2021
1047-3203/© 2021 Elsevier Inc. All rights reserved.

real-time state-of-the-arts VSR models in terms of subjective visual or objective evaluations, and computational costs. In general, the contributions of our paper can be summarized as the following three points:

- We propose a new network architecture for video super-resolution, called as LSVSR, which uses depthwise convolution and pointwise group convolution to simplify the CNNs model for both frame alignment and frame generation, in order to greatly reduce the computational cost of the model.
- We propose a new lightweight frames alignment module to exploit spatial–temporal information without explicit motion compensation to save computations.
- We perform an extensive ablation studies to verify the benefits of the proposed framework and baselines under various settings.
- We show that the performance of LSVSR is better than other real-time state-of-the-arts method in terms of calculations, running time and number of parameters.

2. Related works

2.1. Deep learning in super-resolution

In recent years, due to the rapid developments of deep learning, especially the development of CNNs has brought significant impacts to super resolution [9] and image interpolation [10]. Dong et al. [11] firstly introduced the convolution neural network for single image super resolution (SISR) named SRCNN. Different from traditional methods [12–14], SRCNN proves that convolution neural networks can perform the task of mapping low resolution image (LR) to high resolution image (HR) well. Based on SRCNN, they proposed FSRCNN [15] that uses the deconvolution layer to reconstruct the HR image. FSRCNN avoided pre-interpolation of the image to reduce the amount of calculations. Shi et al. [16] proposed a sub-pixel convolution neural network which uses the pixel shuffle technique to accelerate the reconstruction of HR images. Kim et al. [17] proposed a 20-layer deep network named VDSR with global residual structure. After VDSR, more networks with higher performance and deeper layers were proposed, such as LapSRN, RDN, CARN, ESRGAN, DBPN and SAN [18,17,19–33] etc. With the rapid development of SISR, researchers also use deep learning to solve video super-resolution problems, such as [1,2,6,8,3,34–37]. Huang et al. proposed BRCN [38], which uses a recurrent neural network (RNN) for VSR. They used RNN to model long-term contextual information of temporal sequences well. DECN was proposed by Liao et al. [36] which uses non-iterative framework to reduce the computational load of motion estimation. DECN uses a hand-designed optical flow algorithm to generate SR drafts and uses a novel CNN framework for SR draft construction and final reconstruction. Sajjadi et al. [34] proposed FRVSR that uses a previously inferred HR estimate to help the estimation of the current frame. This framework encourages temporally consistent results. Xue et al. [2] proposed a motion compensation method called task-oriented flow, which results into an end-to-end trainable network that combines motion estimation and image processing. Muhammad et al. [1] proposed RBPN, which uses recurrent encoder-decoder module to gradually fuse temporal information. RBPN allows evaluation at a larger scale and considers videos in different motion regimes.

2.2. Motion compensation in video super-resolution

Based on the ESPCN [16] proposed by Shi et al., Caballero et al. introduced a sub-pixel up-sampling convolution layer for video super-resolution, and proposed a model named VESPCN [8]. Specifically, they proposed the spatial transformer motion compensation module, which explicitly completed motion compensation. This module aligns the neighboring frame information to let the network better restore the reference frame information. Kappeler et al. proposed VSRnet [6], which calculates the optical flow between two frames by the traditional methods [39] and aligns the neighboring frames explicitly with the reference frames. The final result is calculated by stacking the mutil-frames together into the designed convolution neural network. However, there are two problems with explicit motion compensation: If the optical flow field is inaccurate, explicit motion compensation will have a negative effect. Even the obtained optical flow field is ground true, due to lighting inconsistency and occlusion, the image-wrapping based motion compensation may produce doubling artifacts [2]. Calculating the optical flow and conducting explicit motion compensation will increase the computational cost of the model. Implicit motion compensation can solve the drawbacks of explicit motion compensation because it avoids the calculation of optical flow. Hence, the process of motion compensation is implicitly incorporated in the model. To avoid explicit motion compensation, Jo et al. proposed a network structure called DUF [40], which uses the local spatio-temporal neighborhood of each pixel to generate dynamic up-sampling filters and the HR residual image. The residual is added to the output of the previous up-sampling module. They prove that implicit dynamic compensation can also achieve high-level performance. Tian et al. proposed the TDAN [35] model that uses deformable convolution to perform adaptive implicit alignment of input frames. They align input frame at the feature level without computing optical flows. Then, TDAN uses the features to dynamically predict offsets of sampling convolution kernels and up-sample the feature by a reconstruction net finally. Tao et al. [41] proposed a sub-pixel motion compensation (SPMC) module that implements frame alignment and super-resolution simultaneously. SPMC not only completes frame alignment and super-resolution, but also uses Conv-LSTM to utilize inter-frame information and intra-frame information to complete frame fusion. Furthermore, Wang et al. proposed the EDVR [3] model, which also completes frame alignment implicitly and frame fusion. However, they proposed a new model architecture which uses Pyramid, Cascading and Deformable (PCD) modules to solve the alignment problem. In the fusion part, they proposed Temporal and Spatial Attention (TSA) module, which ultilizes a dual attention mechanism to complete the fusion of features. Overall, existing VSR methods either focus on explicit motion estimation or implicit motion estimation with high computational complexity, which is not suitable for real-time VSR.

2.3. Lightweight convolution network

Nowadays, with the development of deep learning, CNNs network structure becomes deeper and deeper, which makes the model difficult to be used on devices with low computing power. To solve this problem, Hinton et al. [42] proposed a method called knowledge distillation to reduce the size of the network. The main idea of knowledge distillation is to train a simple student network by a sophisticated teacher network. First, the training results are generated from the teacher network that has completed the pre-training process, then the student network learns from the output of the trained teacher network. Experiments have shown that the student network trained by knowledge distillation is better than that trained on the dataset alone, because the output of the teacher model provides more concise information than the original dataset. Depthwise separable convolution is proposed by Chollet et al. in the Xception [43] model. The depthwise separable convolution accelerates the inference speed of the model by decoupling the standard convolution kernel into depthwise convolution and pointwise convolution [44]. For the same amount of parameters, models using depthwise separable convolutions can achieve better results because of the larger number of channels. Howard et al. proposed Mobilenet [4] that uses a depthwise separable convolution kernel as the main unit of the model to implement a neural network model that can be run on mobile devices. They performed a detailed analysis of the depthwise separable convolution and reduced calculations of networks by the depth separable convolution. Jing et al. proposed a similar idea of using MobileNet based architecture to accelerate the inference speed in style transfer [45]. Zhang et al. proposed Shuffle Net [5], which uses pointwise group

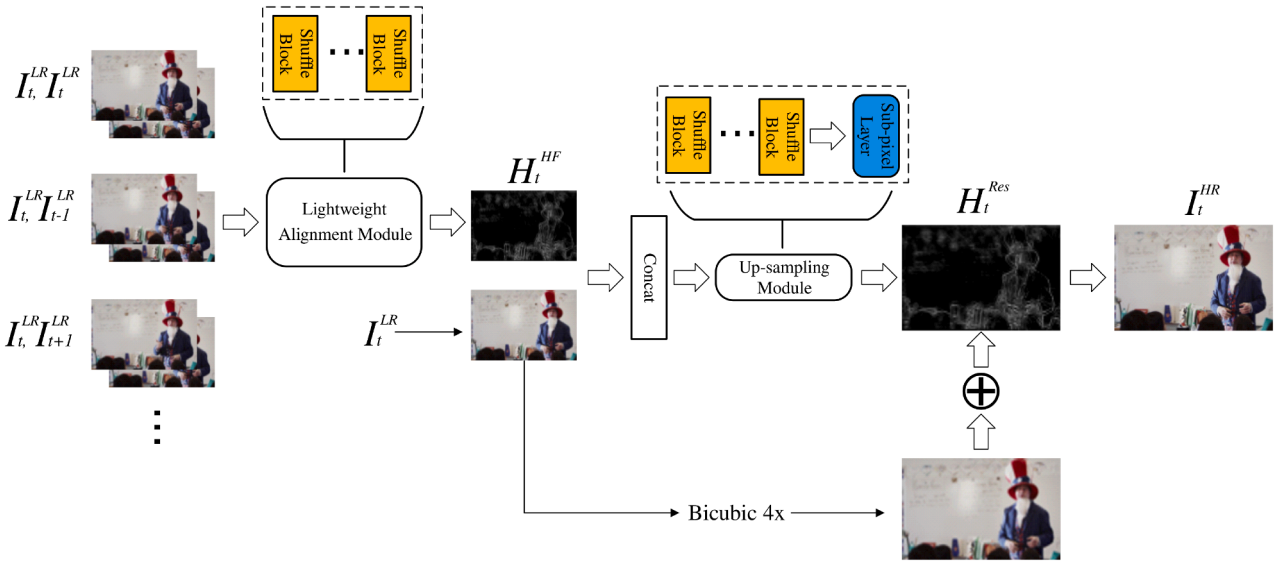


Fig. 1. The overall architecture of our proposed Lightweight Shuffle Video Super-Resolution Network (LSVSR).

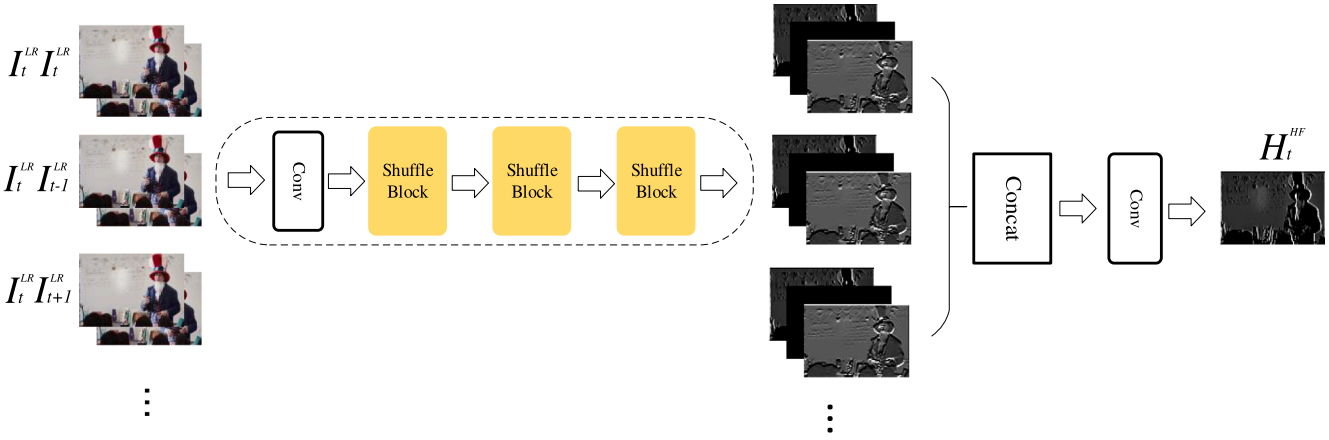


Fig. 2. Lightweight alignment module (LAM) of LSVSR.

convolution to reduce the computations of depthwise separable convolution. However, the use of pointwise group convolution will cause the problem of information exchange blocking between different groups of the convolution kernel. To alleviate this problem, they used channel shuffle technology. The aforementioned models are used in high-level vision tasks. Inspired by them, we propose Lightweight Shuffle Video Super-Resolution Network (LSVSR), which is designed to be lightweight, fast, and without explicit motion estimation and compensation steps. Base on depthwise convolution and pointwise group convolution, we also design a lightweight alignment to perform implicit motion compensation.

VESPCN uses a spatial transformer network to learn the optical flow fields between the neighboring frames. And, it uses huber loss to supervise warping, but this increases the model's learning burden. In our proposed model, as shown in Fig. 1, we inputs pairs of adjacent frames into the proposed LAM for estimating the high frequency information of adjacent frames into the up-sampling module. The overall complexity of proposed framework is extremely low for real-time applications. First, in Section 3.1, we define the VSR problem in detail, and then describe an overview of our proposed framework. Then, the proposed LAM is introduced in Section 3.2. Finally, an up-sampling module in the network for SR reconstruction is introduced in Section 3.3.

3. Proposed method

3.1. Overview

In VSR, let the t-th reference LR image be I_t^{LR} and the corresponding HR image be I_t^{HR}. Restoring HR images I_t^{HR} from I_t^{LR} and $2N$ neighboring frame $[I_{t-N}^{LR},\ldots,I_{t+N}^{LR}]$ is our goal. As shown in Fig. 1, in the LSVSR, we not only concatenate I_t^{LR} with each neighboring frame, but also concatenate I_t^{LR} with itself as the model input. First, we input $I_t^{LR}, I_{t-N}^{LR}, \ldots, (I_t^{LR}, I_{t+N}^{LR})$ into LAM, as shown in Eq. (1). In the equation, H_t^{HF} refers to the high frequency information output by the LAM module, which is used to help the Up-sampling module to restore I_t^{HR}. $F_{LAM}(.)$ refers to LAM, which is shown in Fig. 2 and it will be introduced in detail below:

$$H_t^{HF} = F_{LAM}((I_t^{LR}, I_{t-N}^{LR}), \ldots, (I_t^{LR}, I_{t+N}^{LR})). \tag{1}$$

H_t^{HF} and I_t^{LR} are concatented as input for F_{US}, where $F_{US}(.)$ refers to the Up-sampling module in LSVSR, and H_t^{Res} refers to the high frequency information of I_t^{HR} predicted by LSVSR. The structure of the Up-sampling module is shown in Fig. 5. $F_{Bicubic}(.)$ refers to Biucbic interpolation. Finally, H_t^{Res} is added with the global residual connection to get I_t^{HR}, Eqs. (2) and (3) illustrate the above process:

Table 1
Structure detail of LSVSR.

Module	Lightweight Shuffle Video Super-Resolution Network (LSVSR)	Parameters
LAM (Shared)	Conv(5,2,88)	4.4 k
	$ShuffleBlock\begin{cases}Pointwise-Conv(1,88,88)(Group=8)\\Depthwise-Conv(5,88,88)\\Pointwise-Conv(1,88,88)(Group=8)\end{cases}$	4.1 k
	$ShuffleBlock\begin{cases}Pointwise-Conv(1,88,88)(Group=8)\\Depthwise-Conv(5,88,88)\\Pointwise-Conv(1,88,88)(Group=8)\end{cases}$	4.1 k
	$ShuffleBlock\begin{cases}Pointwise-Conv(1,88,88)(Group=8)\\Depthwise-Conv(5,88,88)\\Pointwise-Conv(1,88,88)(Group=8)\end{cases}$	4.1 k
	Conv(3,264,1)	2.4 k
Up-sampling module	Conv(5,2,88)	4.4 k
	$ShuffleBlock\begin{cases}Pointwise-Conv(1,88,88)(Group=8)\\Depthwise-Conv(5,88,88)\\Pointwise-Conv(1,88,88)(Group=8)\end{cases}$	4.1 k
	$ShuffleBlock\begin{cases}Pointwise-Conv(1,88,88)(Group=8)\\Depthwise-Conv(5,88,88)\\Pointwise-Conv(1,88,88)(Group=8)\end{cases}$	4.1 k
	$ShuffleBlock\begin{cases}Pointwise-Conv(1,88,88)(Group=8)\\Depthwise-Conv(5,88,88)\\Pointwise-Conv(1,88,88)(Group=8)\end{cases}$	4.1 k
	$ShuffleBlock\begin{cases}Pointwise-Conv(1,88,88)(Group=8)\\Depthwise-Conv(5,88,88)\\Pointwise-Conv(1,88,88)(Group=8)\end{cases}$	4.1 k
	$Pointwise-Conv(1,88,1)$	0.088 k

$$H_t^{Res} = F_{US}(H_t^{HF}, I_t^{LR}), \tag{2}$$

$$I_t^{HR} = H_t^{Res} + F_{Bicubic}(I_t^{LR}). \tag{3}$$

The details of the LSVSR are shown in Table 1. We will introduce the Shuffle Block, LAM and up-sampling module in details as follow.

3.2. Lightweight alignment module

The description of the LAM is divided into three parts. The first part is depthwise convolution and pointwise group convolution, which are the main component of the shuffle block. Secondly, we give a comprehensive description of the LAM module. Finally, we introduce shuffle block, which is the main components of LAM.

3.2.1. Lightweight convolution

The concept of pointwise convolution comes from the depthwise separable convolution in MobileNet. The depthwise separable convolution splits the standard convolution kernel into depthwise convolution and pointwise convolution. The depthwise separable convolution reduces the computational complexity of the standard convolution by 7 to 9 times. The role of pointwise convolution is to compensate the feature channel correlation information lost by depthwise convolution. Although the convolution kernel size is 1×1, the amount of computation is still very large. We can quantify the depthwise convolution, the pointwise convolution, and their ratios into the following equation. Eq. (4) illustrates the amount of computation for depthwise convolution:

$$C_{depthwise} = k^2 * n * W * H, \tag{4}$$

where k is the size of the convolution kernel, n is the number of input channels, H and W is the height, width of the output feature map. Eq. (5) illustrates the amount of computation for pointwise convolution:

$$C_{pointwise} = m * n * W * H, \tag{5}$$

where m refers to the number of output channels, and we use the following equation to compare the calculation of the two convolutions:

$$P = C_{pointwise} / C_{depthwise} = m / k^2. \tag{6}$$

If we use a convolution kernel with a 3×3 kernel and 64 channel number, the computation of the pointwise convolution will be approximately 7 times to the depthwise convolution. Grouping the point convolutions is an effective way to reduce the computation. However, group convolution will cause problems with channel correlation information loss. To solve this problem, the idea of channel shuffling is proposed. The process of channel shuffling is shown in Fig. 4. Channel shuffling

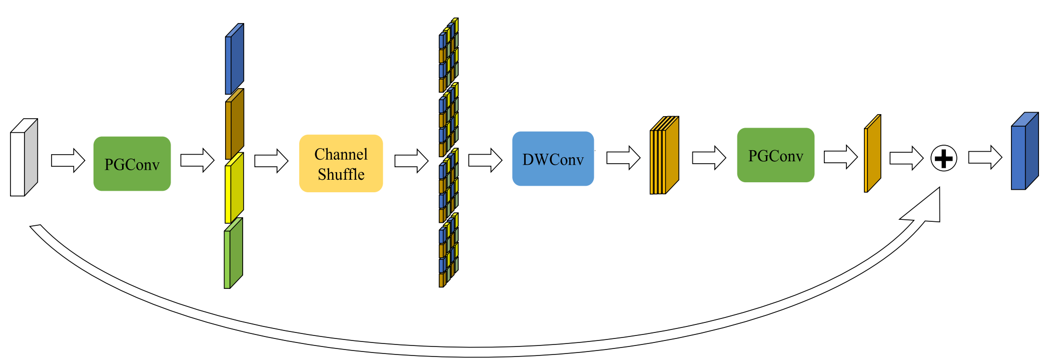


Fig. 3. Proposed shuffle block for frame alignment and up-sampling.

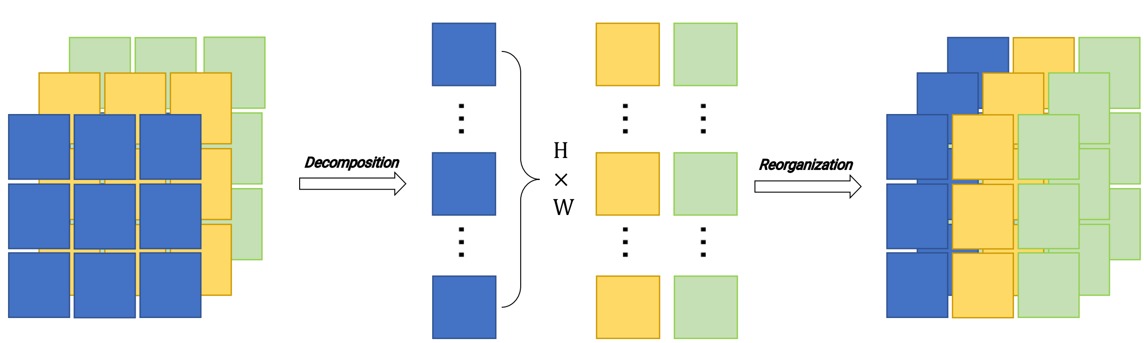


Fig. 4. Channel shuffle operation.

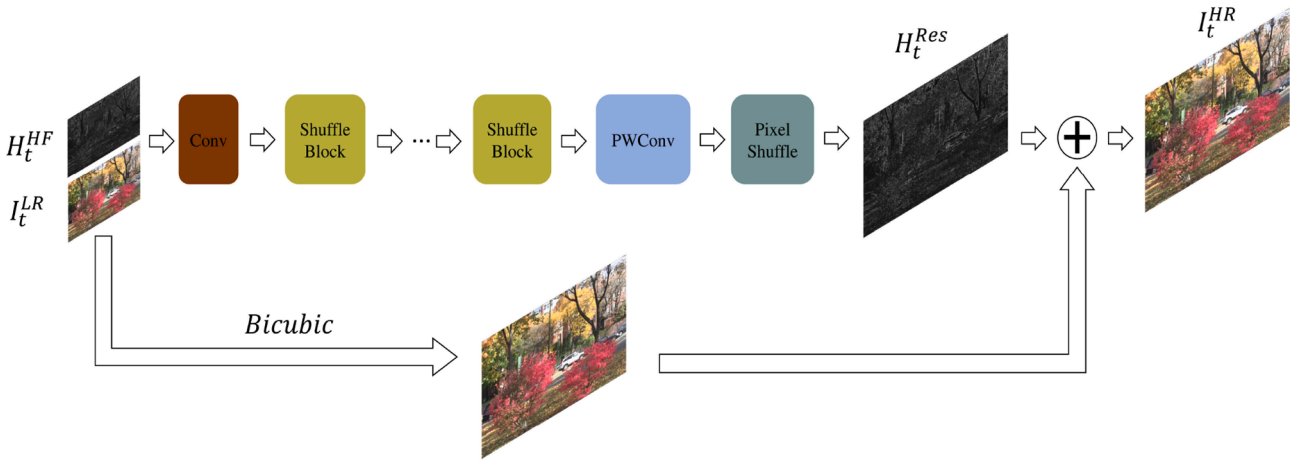


Fig. 5. Upsampling module in LSVSR. This module is responsible for combining high frequency information to complete image upsampling.

reorganizes the feature map of each channel at the pixel level, and the feature maps are evenly mixed. This mitigates the negative impact of channel correlation information loss.

3.2.2. Lightweight alignment module using shuffle block

The LAM module is mainly composed of three shuffle blocks, as shown in Fig. 2. The input method of LAM allows it to avoid learning the correlations between $[I_{t-N}^{LR}, \ldots, I_{t-1}^{LR}, I_{t+1}^{LR}, \ldots, I_{t+N}^{LR}]$. LAM only needs to estimate the differences between the two frames in each calculation, which greatly reduces the learning burden of the LAM. By sequentially inputting multiple sets of pictures into LAM, the LAM module can learn the high frequency information contained in different neighboring frames to help restore the reference frame. After obtaining the aligned feature map by a 3×3 standard convolution kernel, we converted the feature map of the number of $(2N+1)*c$ channels into a single-channel feature map containing the high-frequency details, where c is the number of output channels of the alignment module. In our model, we set the output channel of the shuffle block to 88. H_t^{HF} is the high frequency detail information obtained by combining the multiple neighboring frame and the reference frame, and H_t^{HF} will help the up-sampling module to better restore the HR image.

3.2.3. Shuffle block

Shuffle Net has achieved good performance in image classification tasks. Like Shuffle Net, in this paper, we use channel shuffle technology in the model. However, we used a depthwise convolution with a kernel size of 5×5 instead of the original 3×3 depthwise convolution, which increased the receptive field of the model. The structure of the shuffle block is shown in Fig. 3. In Fig. 3, we divide the pointwise convolution into four groups as an example. After channel shuffling operation, the features of each channel need to be shuffled in space. For this purpose, the features need to be processed with depthwise convolution. Finally, the shuffled feature map is processed using a pointwise group convolution, of which residual connections are used. First, we use the following equation to indicate the amount of shuffle block calculation:

$$C_{SB} = n*W*H*(\frac{2m}{g} + k^2), \tag{7}$$

where g is the number of groups in group pointwise convolution. The amount of calculation for the depthwise separable convolution can be explained by the following equation:

$$C_{DS} = n*W*H*(k^2 + m). \tag{8}$$

The amount of calculation of the standard convolution is calculated by Eq. (9):

$$C_{Conv} = n*m*W*H*k^2. \tag{9}$$

If the number of input channel is 64, the number of groups is 8, and the size of the kernel size is 3×3, the calculation of the depthwise separable convolution is about 3 times more than the shuffle block, and the calculation of the convolution is about 23 times more than the shuffle block. This illustrates the calculation complexity of the shuffle block is extremely low.

3.3. Up-sampling module

As shown in Fig. 5, the purpose of the module is to efficiently use the extra information obtained by the inter-frame correlation to perform super-resolution. The feature map obtained by LAM and the LR image are concatenated as the input of the up-sampling module. First, the feature is extracted by a 5×5 convolution kernel, and the feature map is put into several shuffle blocks for feature mapping. Then, a pointwise convolution is used to reduce the channel of the mapped feature map and change its output channel to $r \times r$. Finally, the LR image is enlarged using bicubic and added to the high frequency information feature map to obtain the final HR image.

4. Experimental results

In this section, we will explain the experiment of LSVSR in details. In our experiments, we mainly focus on $4\times$ SR factor. First, we will introduce the training set and test set used in the experiment, and introduce various experimental parameter settings. Next, we compare the proposed model with other the-state-of-the-art methods. Finally, we will perform ablation experiments on each part of the proposed model and carefully verify the role of each module.

4.1. Training and testing details

In our experiments, Vimeo90K is used as the training set for the model. Vimeo90K included a total of 90 K 7-frame videos. We use bicubic to down-sample the training set by a factor of 4, and cut a lot of 64×64 images from each frame for training. For each video, we select the third frame in each 7-frame videos as the training reference frame, and the rest of the frames are used as their neighboring frames. Before training, we filter the references frame of the original Vimeo90k video. If the variance of the references frame is less than 1600, the video is removed. After augmentation, about 65 K videos remain. Moreover, we performed data augmentation, including vertical flipping, horizontal flipping, and 180-degree rotation to expand the data to four times of the original. All training images are converted to YCbCr format, and only the

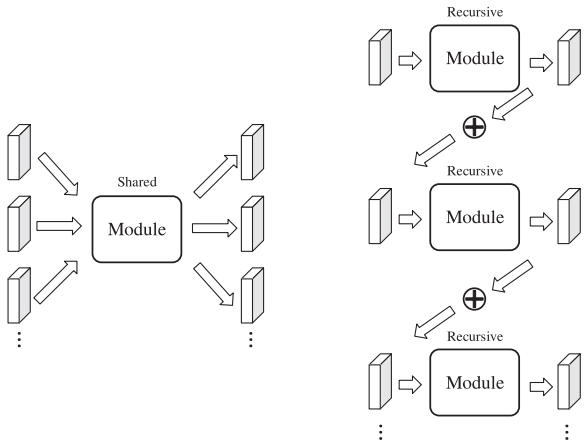
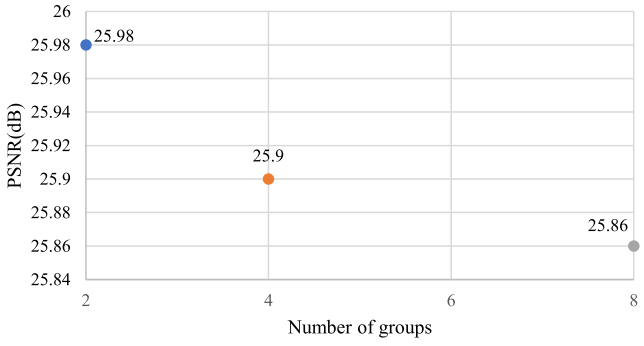
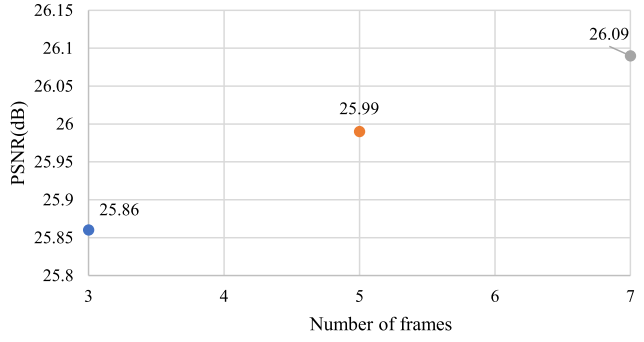


Fig. 6. Shared alignment module vs. Recursive alignment module.



(a) Different number of groups



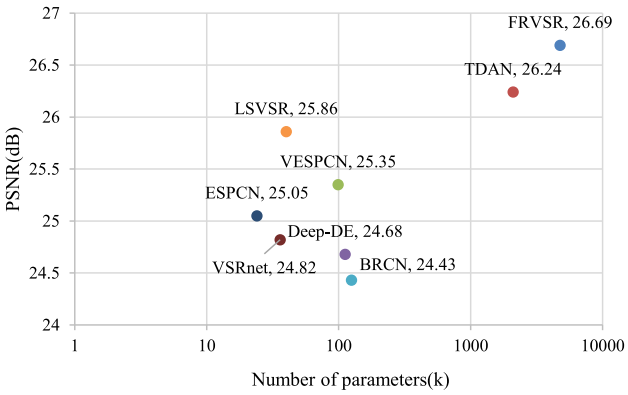
(b) Different number of frames

Fig. 7. Comparison experiments of setting different number of groups and frames in LSVSR.

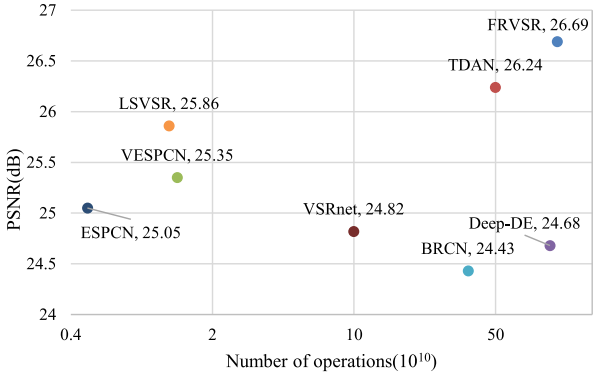
Y channel is used as input during training and testing PSNR and SSIM values. We chose Vid4 for testing, and Vid4 are 4 short video that includes city, walk, calendar, and foliage. We built our model on Ubuntu 16.04.4 using Pytorch version 0.11_5. During training, we set the batch size to 8, and fixed the learning rate to 0.001 for training 150 epochs. We use MSE as the loss function during training. Also, Adam is used as the model's optimizer with β_1=0.9, β_2=0.999 and $\epsilon = 10^{-8}$.

4.2. Comparision with state-of-the-art real-time SR methods

In this section, we compare our proposed LSVSR with other real-time state-of-the-arts methods. We only consider the model's multiplication operation when calculating the number of operations and assume that



(a) PSNR vs parameters on Vid4.



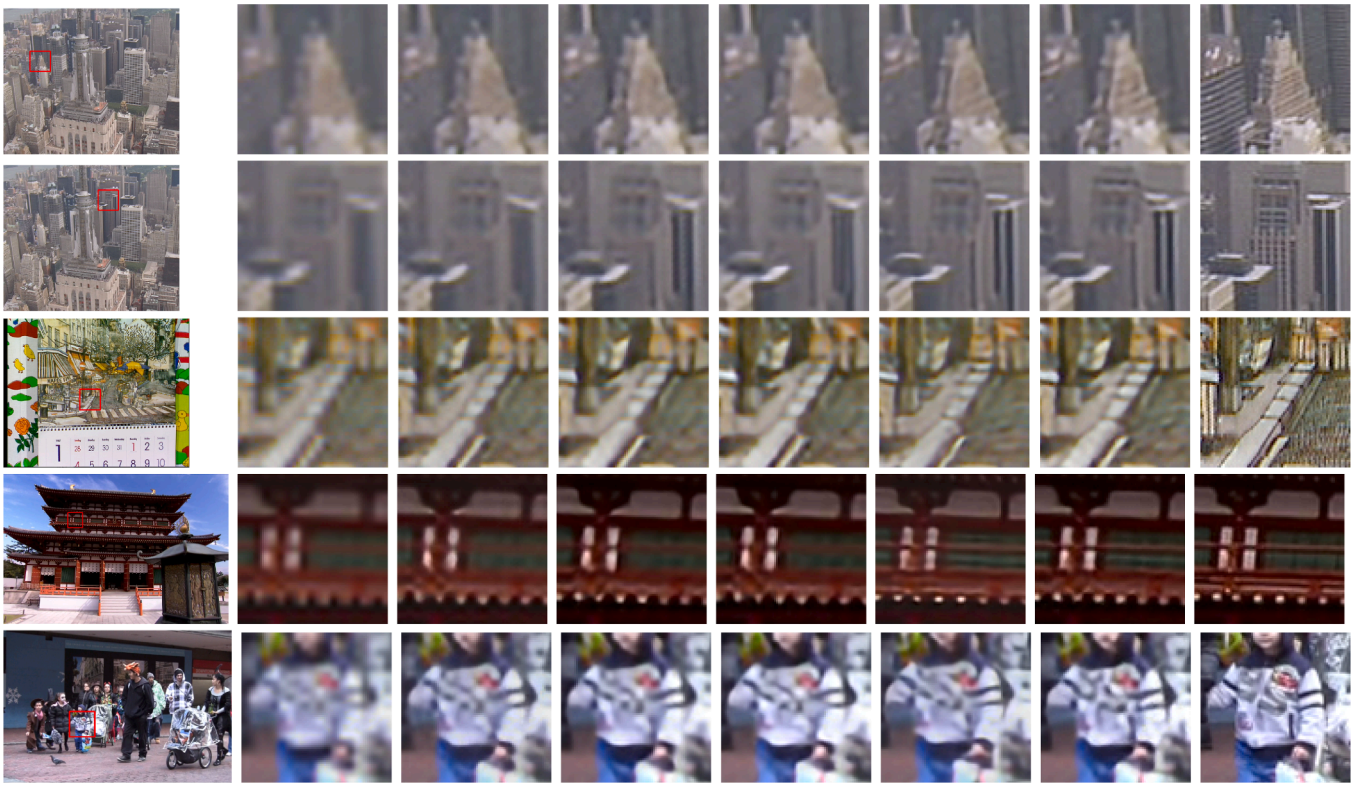
(b) PSNR vs operations on Vid4.

Fig. 8. Comparison experiments of multiple methods.

the size of I_t^{LR} is 512 × 383. The results of the comparison are shown in Table 4. For calculation, compared with other methods, LSVSR is superior to most methods except ESPCN. The reason for the small amount of calculation of ESPCN is that it only takes a single frame as its input, and does not consider the connection of the neighboring frame. Therefore, its performance is 0.75 dB lower than LSVSR as shown in Table 4. In terms of the number of parameters, in addition to ESPCN, the number of parameters of VSRnet is also lower than LSVSR. However, since it is necessary to interpolate the LR image in advance, the amount of calculation is large, and the performance is not satisfactory. The PSNR measured by VSRnet on Vid4 is 0.98 dB lower than LSVSR. As we can see from Table 4, the proposed LSVSR can still achieve excellent performance with low parameter count and low computational complexity. At the same time, we can see the results of different methods in Fig. 8. In Figs. 9 and 10, we show the resulting images from LSVSR and other methods. Obviously, LSVSR can better restore image details than other methods. Also, we list the PSNR of LSVSR and VESPCN for different scale in Table 5. It can be seen that in the case of 3× and 4× , the PSNR measured by LSVSR on Vid4 is 0.51 dB and 0.45 dB higher than VESPCN, respectively. In Table 5, we can see that when the input image is 720 × 576, we calculate the running speed and LSVSR is better than VESPCN.

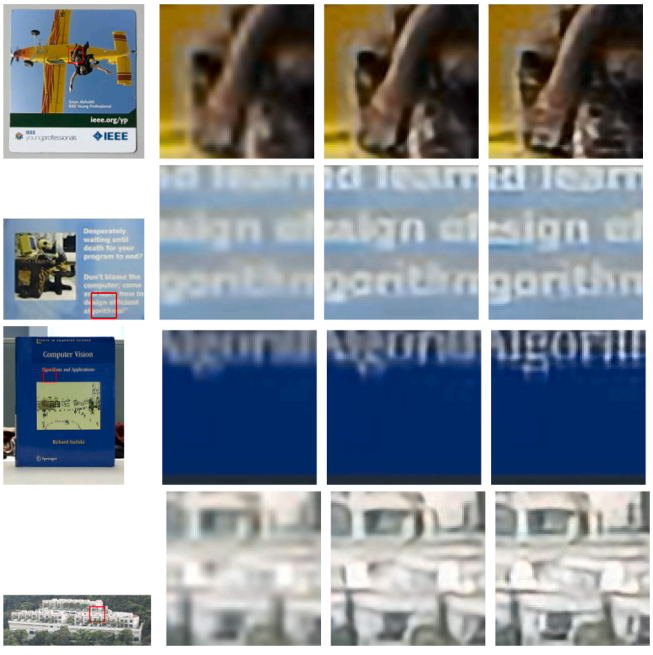
4.3. Ablation study

In this section, we will prove the role of each part of LSVSR. The first is a set of comparative experiments with VESPCN, which proves that the alignment module and up-sampling module of LSVSR are superior to VESPCN. Second, we verified the role of channel shuffle in the shuffle block. Then, we compare the performance of the shared convolution



(a) Image (b) Bicubic (c) ESPCN (d) VESPCN (e) VSRnet (f) DRVSR (g) Ours (h) Original

Fig. 9. 4x SR results in temple,walk and city dataset.



(a) Original (b) Bicubic (c) VESPCN (d) Ours

Fig. 10. 4x SR results of real-world video [41].

kernel with the recursive convolution kernel. Finally, we compare our method with other the state-of the arts.

4.3.1. Comparison with VESPCN

In this section, we will analyze and verify the effects of each component of the LSVSR in detail. We use VESPCN as the baseline of LSVSR. LSVSR finally uses 3 frames including reference frames as the input of the model, and the other two frames are the previous frame and the next frame of the reference frame. In order to make a reasonable comparison with VESPCN, we replace LAM of LSVSR with spatial transformer motion compensation module of VESPCN, and also use 3 frames as input, while the LSVSR up-sampling module remain unchanged. The purpose of this experiment is to certificate that the proposed alignment module is better than VESPCN. As can be seen in Table 2, using the LSVSR structure for the frame extraction and alignment, the PSNR on Vid4 is 0.46 dB higher than the motion compensation structure using VESPCN. This experiment shows that our model can better utilize the information of consecutive frames. For the up-sampling module, we design a set of comparison experiments. We keep the LAM parts unchanged, and replace the LSVSR up-sampling module with the same up-sampling module as VESPCN. This comparison experiment proves that the up-sampling module of our model has better

Table 2
Ablation Study of Proposed LSVSR. Up-sample means whether the VESPCN style up-sampling module is used instead of the Up-sampling Module in the LSVSR. Shuffle means whether not to use channel shuffle technology. Recursive means whether to use recursive LAM modules instead of shared structures. MC means whether the VESPCN motion compensation module is used instead of the LAM module.

Config	Up-sample	Shuffled	Recursive	MC	PSNR
1	√	×	×	×	25.45
2	×	√	√	×	25.60
3	×	×	√	×	25.77
4	×	×	×	√	25.40
Ours	×	×	×	×	25.86

Table 3
Experimental results of different frame numbers and group numbers. G2 means group = 2 and F3 means frame = 3.

Config	G2	G4	G8	F3	F5	F7	PSNR	SSIM
1	×	×	√	√	×	×	25.86	0.7561
2	×	×	√	×	√	×	25.99	0.7578
3	×	×	√	×	×	√	26.09	0.7626
4	×	√	×	√	×	×	25.90	0.7587
5	√	×	×	√	×	×	25.98	0.7666

Table 4
Complexity and performance of video SR Methods for 4× SR. Temple, Penguin are short video datasets from [41].

	Para.	Oper.	Vid4	Temple	Penguin
Bicubic	-	-	23.82	26.99	33.31
BRCN [38]	125 k	3.66×10^{11}	24.43	-	-
Bayesian	-	-	24.66	-	-
Deep-DE [36]	112 k	9.28×10^{11}	24.68	-	-
VSRnet [6]	36 k	9.96×10^{10}	24.82	27.17	33.82
ESPCN [16]	24 k	4.83×10^{9}	25.05	29.03	36.54
VESPCN [8]	99 k	1.34×10^{10}	25.35	30.10	36.88
Ours	40 k	1.22×10^{10}	25.86	30.78	36.96
DRVSR [41]	1081 k	3.27×10^{11}	25.88	-	-
TDAN [35]	2096 k	4.99×10^{11}	26.24	-	-
FRVSR [34]	4769 k	1.07×10^{11}	26.69	-	-
RBPN [1]	1814 k	2.34×10^{12}	27.12	-	-
TOF [2]	715 k	2.24×10^{12}	-	-	-
EDVR [3]	1731 k	3.40×10^{11}	27.35	-	-

performance than the up-sampling module of VESPCN. The LSVSR up-sampling module has more feature channels that allows the features to be fully mapped. However, the computational complexity of the LSVSR up-sampling module is lower than VESPCN.

4.3.2. *Channel shuffle*

In this experiment, we try to remove the channel shuffle operation used in all shuffle blocks in order to prove its effectiveness. In Table 2, we give the results of the model after removing the channel shuffle operation. It can be seen that the model has a 0.17 dB drop, which proves that the operation plays an important role in enhancing the feature channel correlation of the convolution.

4.3.3. *Shared vs. Recursive alignment module*

We find that using multiple frame information recursively is worse than shared. The difference between them can be seen in Fig. 6. Using a recursive structure is 0.09 dB lower than using a shared structure. Combining information from multiple neighboring frames will make network learning difficult. To prove this opinion, first, we extract the features from $[I_t, I_t]$ and input it into the alignment module, and add the result to the feature map from $[I_{t-1}, I_t]$. After that, put it into the alignment module again. Finally, by analogy, a recursive structure is formed. Similarly, we give the results of the model using the recursive structure on Vid4 in Table 2. Due to the multi-frames information is fused together, the learning of the network becomes more difficult, and the performance is not as good as the shared alignment module.

4.3.4. *Number of frames and groups*

In this section, multiple sets of comparison experiments are designed that input different numbers of frames and different numbers of groups in pointwise group convolution. First, number of input frames to 3, 5, and 7 frames respectively. As can be seen from Table 3, compared to the input of 3 frames, when the number of input frames is 5 frames and 7 frames, the performance improves, reaching 25.99 dB and 26.09 dB, respectively. Secondly, we conducted a ablation experiment on the number of group convolutions in the shuffle block, and set them to 2, 4, and 8, respectively. Different group numbers will affect the performance of the model. Fig. 7 shows the results of the comparison test. As the group increases, the performance of the model decreases. However, due to the channel shuffle operation, when we gradually increase the group, the performance does not sudden drop. When we set the group to 2 and 4, the model results are 25.98 dB and 25.90 dB respectively.

Table 5
Comparison of VESPCN and LSVSR under different scale on Vid4 and their runtime for 3× SR for 720 × 576 input on Nvidia 980 Ti.

Algorithm	Scale	PSNR(dB)	SSIM	Runtime
VESPCN	2×	-	-	33 ms
	3×	27.25	0.8447	
	4×	25.35	0.7557	
Ours	2×	32.53	0.9411	29 ms
	3×	27.76	0.8463	
	4×	25.86	0.7561	

4.3.5. *Real-time processing on desktop GPUs and mobile phones*

Modern desktop GPUs and mobile phones are capable to process tera operations per second (Tops). For example, Huawei mobile phones with Kirin 810/990 chipsets are incorporated with neural processing units with processing power up to 8 tera-ops or more. Our network requires 32.25 giga-ops (3.225×10^{10}) for 4× super-resolution of 540P to 2160P. Hence, our network runs at 248.05 frames per second on modern mobile phones with 8 tera-ops theoretically. Assuming a portion of performance loss due to bottlenecks of realizations, our network can run at more than 60 frames per second on modern mobile phones. Desktop GPUs with higher processing power than mobile phones can generate more frames per second and higher resolution outputs using our LSVSR.

5. Conclusion

Real-time super resolution algorithms have been applied in many applications in real-world scenarios, hence it is necessary to design a real-time super resolution algorithm which generates higher quality images. In this paper, we propose a real-time video super resolution algorithm called Lightweight Shuffle Video Super Resolution Network (LSVSR). Specifically, we propose a new frame alignment module named LAM based on the shared shuffle block to complete the implicit alignment between neighboring frames. Moreover, an up-sampling module based on shuffle block with depthwise and group convolution is proposed to perform high resolution image reconstruction. The proposed LSVSR model only has a 40 k parameters and requires minimal amount of computations, so it can be deployed on some computing resource-poor devices to be used in real-world scenarios. LSVSR still has rooms for computation reductions, of which we will investigate further. For example, knowledge distillation can be used to generate new models with lower complexity, so that the performence of the model will be further improved.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

This work is supported by NSFC No: 62002230.

References

[1] M. Haris, G. Shakhnarovich, N. Ukita, Recurrent back-projection network for video super-resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3897–3906.

[2] T. Xue, B. Chen, J. Wu, D. Wei, W.T. Freeman, Video enhancement with task-oriented flow, Int. J. Comput. Vision 127 (2019) 1106–1125.

[3] X. Wang, K.C. Chan, K. Yu, C. Dong, C. Change Loy, Edvr: Video restoration with enhanced deformable convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2019.

[4] A.G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, H. Adam, Mobilenets: Efficient convolutional neural networks for mobile vision applications, arXiv preprint arXiv:1704.04861 (2017).

[5] X. Zhang, X. Zhou, M. Lin, J. Sun, Shufflenet: An extremely efficient convolutional neural network for mobile devices, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6848–6856.

[6] A. Kappeler, S. Yoo, Q. Dai, A.K. Katsaggelos, Video super-resolution with convolutional neural networks, IEEE Trans. Comput. Imag. 2 (2016) 109–122.

[7] S.H. Keller, F. Lauze, M. Nielsen, Motion compensated video super resolution, in: International Conference on Scale Space and Variational Methods in Computer Vision, Springer, 2007, pp. 801–812.

[8] J. Caballero, C. Ledig, A. Aitken, A. Acosta, J. Totz, Z. Wang, W. 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, 2017, pp. 4778–4787.

[9] W.-C. Siu, Z.-S. Liu, J.-J. Huang, K.-W. Hung, Learning approaches for super-resolution imaging, in: Learning Approaches in Signal Processing, Jenny Stanford Publishing, 2018, pp. 283–352.

[10] K.-W. Hung, K. Wang, J. Jiang, Image interpolation using convolutional neural networks with deep recursive residual learning, Multimedia Tools Appl. 78 (2019) 22813–22831.

[11] C. Dong, C.C. Loy, K. He, X. Tang, Image super-resolution using deep convolutional networks, IEEE Trans. Pattern Anal. Machine Intell. 38 (2015) 295–307.

[12] K.-W. Hung, W.-C. Siu, Robust soft-decision interpolation using weighted least squares, IEEE Trans. Image Process. 21 (2011) 1061–1069.

[13] R. Timofte, V. De Smet, L. Van Gool, A+: Adjusted anchored neighborhood regression for fast super-resolution, in: Asian Conference on Computer Vision, Springer, 2014, pp. 111–126.

[14] C.-Y. Yang, C. Ma, M.-H. Yang, Single-image super-resolution: A benchmark, in: European Conference on Computer Vision, Springer, 2014, pp. 372–386.

[15] C. Dong, C.C. Loy, X. Tang, Accelerating the super-resolution convolutional neural network, in: European Conference on Computer Vision, Springer, 2016, pp. 391–407.

[16] W. Shi, J. Caballero, F. Huszár, J. Totz, A.P. Aitken, R. Bishop, D. Rueckert, Z. 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, 2016, pp. 1874–1883.

[17] J. Kim, J. Kwon Lee, K. Mu Lee, Accurate image super-resolution using very deep convolutional networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 1646–1654.

[18] W.-S. Lai, J.-B. Huang, N. Ahuja, M.-H. Yang, Deep laplacian pyramid networks for fast and accurate super-resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 624–632.

[19] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, Y. Fu, Residual dense network for image super-resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 2472–2481.

[20] N. Ahn, B. Kang, K.-A. Sohn, Fast, accurate, and lightweight super-resolution with cascading residual network, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018, pp. 252–268.

[21] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, Y. Qiao, C. Change Loy, Esrgan: Enhanced super-resolution generative adversarial networks, in: Proceedings of the European Conference on Computer Vision (ECCV), 2018.

[22] Q. Chang, K.-W. Hung, J. Jiang, Deep learning based image super-resolution for nonlinear lens distortions, Neurocomputing 275 (2018) 969–982.

[23] Z. Li, J. Yang, Z. Liu, X. Yang, G. Jeon, W. Wu, Feedback network for image super-resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 3867–3876.

[24] K.-W. Hung, W.-C. Siu, Novel dct-based image up-sampling using learning-based adaptive k-nn mmse estimation, IEEE Trans. Circuits Syst. Video Technol. 24 (2014) 2018–2033.

[25] T. Dai, J. Cai, Y. Zhang, S.-T. Xia, L. Zhang, Second-order attention network for single image super-resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2019, pp. 11065–11074.

[26] Q. Li, Z. Li, L. Lu, G. Jeon, K. Liu, X. Yang, Gated multiple feedback network for image super-resolution, arXiv preprint arXiv:1907.04253 (2019).

[27] J.-H. Choi, J.-H. Kim, M. Cheon, J.-S. Lee, Lightweight and efficient image super-resolution with block state-based recursive network, arXiv preprint arXiv: 1811.12546 (2018).

[28] J.-H. Kim, J.-H. Choi, M. Cheon, J.-S. Lee, Ram: Residual attention module for single image super-resolution, arXiv preprint arXiv:1811.12043 (2018).

[29] P. Liu, H. Zhang, K. Zhang, L. Lin, W. Zuo, Multi-level wavelet-cnn for image restoration, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2018, pp. 773–782.

[30] K. Zhang, W. Zuo, L. Zhang, Learning a single convolutional super-resolution network for multiple degradations, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 3262–3271.

[31] J. Yu, Y. Fan, J. Yang, N. Xu, Z. Wang, X. Wang, T. Huang, Wide activation for efficient and accurate image super-resolution, arXiv preprint arXiv:1808.08718 (2018).

[32] D. Liu, Z. Wang, Y. Fan, X. Liu, Z. Wang, S. Chang, X. Wang, T.S. Huang, Learning temporal dynamics for video super-resolution: A deep learning approach, IEEE Trans. Image Process. 27 (2018) 3432–3445.

[33] Y. Fan, H. Shi, J. Yu, D. Liu, W. Han, H. Yu, Z. Wang, X. Wang, T. S. Huang, Balanced two-stage residual networks for image super-resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops, 2017, pp. 161–168.

[34] M.S. Sajjadi, R. Vemulapalli, M. Brown, Frame-recurrent video super-resolution, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2018, pp. 6626–6634.

[35] Y. Tian, Y. Zhang, Y. Fu, C. Xu, Tdan: Temporally deformable alignment network for video super-resolution, arXiv preprint arXiv:1812.02898 (2018).

[36] R. Liao, X. Tao, R. Li, Z. Ma, J. Jia, Video super-resolution via deep draft-ensemble learning, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 531–539.

[37] K.-W. Hung, C. Qiu, J. Jiang, Video super resolution via deep global-aware network, IEEE Access 7 (2019) 74711–74720.

[38] Y. Huang, W. Wang, L. Wang, Bidirectional recurrent convolutional networks for multi-frame super-resolution, in: Adv. Neural Informat. Process. Syst., 2015, pp. 235–243.

[39] M. Drulea, S. Nedevschi, Total variation regularization of local-global optical flow, in: 2011 14th International IEEE Conference on Intelligent Transportation Systems (ITSC), IEEE, 2011, pp. 318–323.

[40] Y. Jo, S. Wug Oh, J. Kang, S. 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, 2018, pp. 3224–3232.

[41] X. Tao, H. Gao, R. Liao, J. Wang, J. Jia, Detail-revealing deep video super-resolution, in: Proceedings of the IEEE International Conference on Computer Vision, 2017, pp. 4472–4480.

[42] G. Hinton, O. Vinyals, J. Dean, Distilling the knowledge in a neural network, arXiv preprint arXiv:1503.02531 (2015).

[43] F. Chollet, Xception: Deep learning with depthwise separable convolutions, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1251–1258.

[44] K.-W. Hung, Z. Zhang, J. Jiang, Real-time image super-resolution using recursive depthwise separable convolution network, IEEE Access 7 (2019) 99804–99816.

[45] Y. Jing, X. Liu, Y. Ding, X. Wang, E. Ding, M. Song, S. Wen, Dynamic instance normalization for arbitrary style transfer, arXiv preprint arXiv:1911.06953 (2019).