Fast Spatio-Temporal Residual Network for Video Super-Resolution

Sheng Li[1], Fengxiang He[2], Bo Du*[1], Lefei Zhang*[1], Yonghao Xu[3], and Dacheng Tao[2]

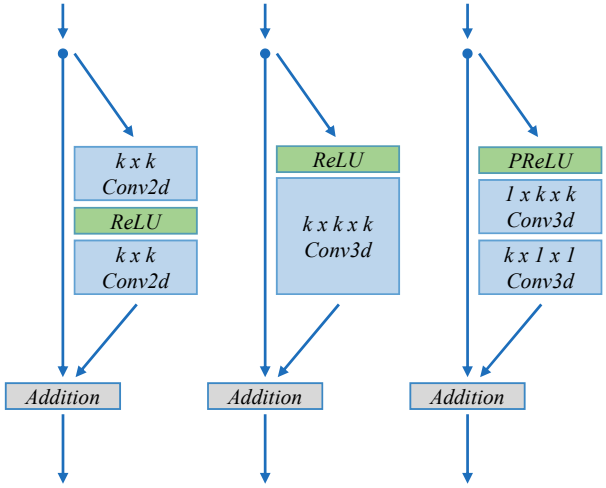
[1]School of Computer Science, Wuhan University, China
[2]UBTECH Sydney AI Centre, SCS, FEIT, the University of Sydney, Australia
[3]The State Key Laboratory of Information Engineering in Surveying, Mapping, and Remote Sensing, Wuhan University, China
{shli, remoteking, zhanglefei}@whu.edu.cn {fengxiang.he, dacheng.tao}@sydney.edu.au
yonghaoxu@ieee.org

Abstract

Recently, deep learning based video super-resolution (SR) methods have achieved promising performance. To simultaneously exploit the spatial and temporal information of videos, employing 3-dimensional (3D) convolutions is a natural approach. However, straight utilizing 3D convolutions may lead to an excessively high computational complexity which restricts the depth of video SR models and thus undermine the performance. In this paper, we present a novel fast spatio-temporal residual network (FSTRN) to adopt 3D convolutions for the video SR task in order to enhance the performance while maintaining a low computational load. Specifically, we propose a fast spatio-temporal residual block (FRB) that divide each 3D filter to the product of two 3D filters, which have considerably lower dimensions. Furthermore, we design a cross-space residual learning that directly links the low-resolution space and the high-resolution space, which can greatly relieve the computational burden on the feature fusion and up-scaling parts. Extensive evaluations and comparisons on benchmark datasets validate the strengths of the proposed approach and demonstrate that the proposed network significantly outperforms the current state-of-the-art methods.

1. Introduction

Super-resolution (SR) addresses the problem of estimating a high-resolution (HR) image or video from its low-resolution (LR) counterpart. SR is wildly used in various computer vision tasks, such as satellite imaging [4] and surveillance imaging [17]. Recently, deep learning based methods have been a promising approach to solve SR problem [5, 20, 27, 29, 30, 45]. A straight idea for video SR is to perform single image SR frame by frame. However, it ignores the temporal correlations among frames, the output HR videos usually lack the temporal consistency, which may emerge as spurious flickering artifacts [33].



(a) EDSR (b) 3D residual block (c) Proposed FRB

Figure 1: Comparison of (a) residual block in EDSR[27], (b) single C3D residual block, and (c) the proposed FRB.

Most existing methods for the video SR task utilize the temporal fusion techniques to extract the temporal information in the data, such as motion compensation [3, 39], which usually need manually deigned structure and much more computational consumption. To automatically and simultaneously exploit the spatial and temporal information, it is natural to employ 3-dimensional (3D) filters to

*Corresponding author.

Figure 2: Visually observations on the orginal frames and the SR results on the Dancing video at $\times 4$ SR, it is noticeable that the proposed FSTRN approach not only achieves the highest PSNR and SSIM values, but also restores the finest texture with the fewest artifacts.

replace 2-dimensional (2D) filters. However, the additional dimension would bring much more parameters and lead to an excessively heavy computational complexity. This phenomenon severely restricts the depths of the neural network adopted in the video SR methods and thus undermine the performance [15].

Since there are considerable similarities between the input LR videos and the desired HR videos, the residual connection is widely involved in various SR networks [20, 25, 27], fully demonstrating the residual connection advantages. However, the residual identity mapping for SR task are beyond sufficient usage, it is either applied on HR space [20, 37], largely increasing the computational complexity of the network, or applied on the LR space to fully retain the information from the original LR inputs [47], imposing heavy burdens on the feature fusion and upscaling stage at the final part of networks.

To address these problems, we propose fast spatio-temporal residual network (FSTRN) (Fig. 3) for video SR. It's difficult and impractical to build a very deep spatio-temporal network directly using original 3D convolution (C3D) due to high computational complexity and memory limitations. So we propose fast spatio-temporal residual block (FRB) (Fig. 1c) as the building module for FSTRN, which consists of skip connection and spatio-temporal factorized C3Ds. The FRB can greatly reduce computational complexity, giving the network the ability to learn spatio-temporal features simultaneously while guaranteeing computational efficiency. Also, global residual learning (GRL) are introduced to utilize the similarities between the input LR videos and the desired HR videos. On the one hand, we adopt to use LR space residual learning (LRL) in order to boost the feature extraction performance. On the other hand, we further propose a cross-space residual connection (CRL) to link the LR space and HR space directly. Through CRL, LR videos are employed as an "anchor" to retain the spatial information in the output HR videos.

Theoretical analyses of the proposed method provide a generalization bound $\mathcal{O}(1/\sqrt{n})$ with no explicitly dependence on the network size (n is the sample size), which guarantees the feasibility of our algorithm on unseen data. Thorough empirical studies on benchmark datasets evaluation validate the superiority of the proposed FSTRN over existing algorithms.

In summary, the main contributions of this paper are threefold:

- We propose a novel framework fast spatio-temporal residual network (FSTRN) for high-quality video SR. The network can exploit spatial and temporal information simultaneously. By this way, we retain the temporal consistency and ease the problem of spurious flickering artifacts.

- We propose a novel fast spatio-temporal residual block (FRB), which divides each 3D filter to the product of two 3D filters which have significantly lower dimensions. By this way, we significantly reduce the computing load while enhance the performance through deeper neural network architectures.

- We propose to employ global residual learning (GRL) which consist of LR space residual learning (LRL) and

cross-space residual learning (CRL) to utilize the considerable similarity between the input LR videos and the output HR videos, which significantly improve the performance.

2. Related work

2.1. Single-image SR with CNNs

In recent years, convolutional neural networks (CNNs) have achieved significant success in many computer vision tasks [13, 23, 24, 34, 36], including the super-resolution (SR) problem. Dong *et al*. pioneered a three layer deep fully convolutional network known as the super-resolution convolutional neural network (SRCNN) to learn the non-linear mapping between LR and HR images in the end-to-end manner [5, 6]. Since then, many research has been presented, which are usually based on deeper network and more advanced techniques.

As the network deepens, residual connections have been a promising approach to relieve the optimization difficulty for deep neural networks [13]. Combining residual learning, Kim *et al*. propose a very deep convolutional network [20] and a deeply-recursive convolutional network (DRCN) [21]. These two models significantly boost the performance, which demonstrate the potentials of the residual learning in the SR task. Tai *et al*. present a deep recursive residual network (DRRN) with recursive blocks and a deep densely connected network with memory blocks [37], which further demonstrates the superior performance of residual learning.

All the above methods work on interpolated upscaled input images. However, directly feeding interpolated images into neural networks can result in a significantly high computational complexity. To address this problem, an efficient sub-pixel convolutional layer [33] and transposed convolutional layer [7] are proposed in order to upscale the feature maps to a fine resolution at the end of the network.

Other methods employing residual connections include EDSR [27], SRResNet [25], SRDenseNet [42] to RDN [47]. However, residual connections are limited within the LR space. These residuals can enhance the performance of feature extraction but would put a excessively heavy load on the up-scaling and fusion parts of the network.

2.2. Video SR with CNNs

Based on image SR methods and further to grasp the temporal consistency, most existing methods employ a sliding frames window [3, 18, 19, 26, 39]. To handle spatio-temporal information simultaneously, existing methods usually utilize temporal fusion techniques, such as motion compensation [3, 19, 26, 39], bidirectional recurrent convolutional networks (BRCN) [14], long short-term memory networks (LSTM) [10]. Sajjadi *et al*. use a different way by using a frame-recurrent approach where the previous estimated SR frames are also redirected into the network, which encourages more temporally consistent results [32].

A more natural approach to learn spatio-temporal information is to employ 3D convolutions (C3D), which has shown superior performances in video learning [16, 43, 44]. Caballero *et al*. [3] mentioned the slow fusion can also be seen as C3D. In addition, Huang *et al*. [15] improved BRCN using C3D, allowing the model to flexibly obtain access to varying temporal contexts in a natural way, but the network is still shallow. In this work, we aimed to build a deep end-to-end video SR network with C3D and maintain high efficiency of computational complexity.

3. Fast spatio-temporal residual network

3.1. Network structure

In this section, we describe the structure details of the proposed fast spatio-temporal residual network (FSTRN). As shown in Fig. 3, FSTRN mainly consists of four parts: LR video shallow feature extraction net (LFENet), fast spatio-temporal residual blocks (FRBs), LR feature fusion and up-sampling SR net (LSRNet), and global residual learning (GRL) part composing by LR space residual learning (LRL) and cross-space residual learning (CRL).

LFENet simply uses a C3D layer to extract features from the LR videos. Let's denote the input and output of the FSTRN as I_{LR} and I_{SR} and the target output I_{HR}, the LFENet can be represented as:

$$F_0^L = H_{LFE}\left(I_{LR}\right), \tag{3.1}$$

where F_0^L is the output of extracted feature-maps, and $H_{LFE}\left(\cdot\right)$ denotes C3D operation in the LFENet. F_0^L is then used for later LR space global residual learning and also used as input to FRBs for further feature extraction.

FRBs are used to extract spatio-temporal features on the LFENet output. Assuming that D of FRBs are used, the first FRB performs on the LFENet output, and the subsequent FRB further extract features on the previous FRB output, so the output F_d^L of the d-th FRB can be expressed as:

$$\begin{aligned} F_d^L &= H_{FRB,d}\left(F_{d-1}^L\right) \\ &= H_{FRB,d}\left(H_{FRB,d-1}\left(\cdots\left(H_{FRB,1}\left(F_0^L\right)\right)\cdots\right)\right), \end{aligned} \tag{3.2}$$

where $H_{FRB,d}$ denotes the operations of the d-th FRB, more details about the FRB will be shown in Section 3.2.

Along with the FRBs, LR space residual learning (LRL) is conducted to further improve feature learning in LR space. LRL makes fully use of feature from the preceding layers and can be obtained by

$$F_{LRL}^L = H_{LRL}\left(F_D^L, F_0^L\right), \tag{3.3}$$

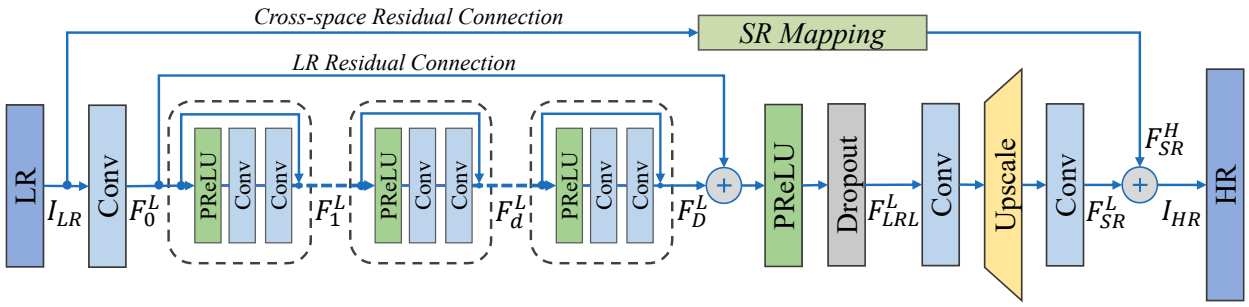


Figure 3: The architecture of our proposed fast spatio-temporal residual network (FSTRN).

where F_{LRL}^L is the output feature-maps of LRL by utilizing a composite function H_{LRL}. More details will be presented in Section 3.3.

LSRNet is applied to obtain super-resolved video in HR space after the efficient feature extraction of LRL. Specifically, we use a C3D for feature fusion followed by a deconvolution [8] for upscaling and again a C3D for feature-map channels tuning in the LSRNet. The output F_{SR}^L can be formulated as:

$$F_{SR}^L = H_{LSR}\left(F_{LRL}^L\right), \tag{3.4}$$

where $H_{LSR}(\cdot)$ denotes the operations of LSRNet.

At last, the network output is composed of the F_{SR}^L from the LSRNet and an additional LR to HR space global residual, forming a cross-space residual learning (CRL) in HR space. The detail of the CRL is also given in Section 3.3. So denote a SR mapping of input from LR space to HR space be F_{SR}^H, the output of FSTRN can be obtained as

$$I_{SR} = H_{FSTRN}\left(I_{LR}\right) = F_{SR}^L + F_{SR}^H, \tag{3.5}$$

where H_{FSTRN} represents the function of the proposed FSTRN method.

3.2. Fast spatio-temporal residual blocks

Now we present details about the proposed fast spatio-temporal residual block (FRB), which is shown in Fig. 1.

Residual blocks have been proven to show excellent performances in computer vision, especially in the low-level to high-level tasks [20, 25]. Lim *et al*. [27] proposed a modified residual block by removing the batch normalization layers from the residual block in SRResNet, as shown in Figure 1a, which showed a great improvement in single-image SR tasks. To apply residual blocks to multi-frame SR, we simply reserve only one convolutional layer, but inflate the 2D filter to 3D, which is similar to [16]. As shown in Figure 1b, the $k \times k$ square filter is expanded into a $k \times k \times k$ cubic filter, endowing the residual block with an additional temporal dimension.

After the inflation, the ensuing problems are obvious, in that it takes much more parameters than 2D convolution, accompanied by more computations. To solve this, we propose a novel fast spatio-temporal residual block (FRB) by factorizing the C3D on the above single 3D residual block into two step spatio-temporal C3Ds, *i.e*., we replace the inflated $k \times k \times k$ cubic filter with a $1 \times k \times k$ filter followed by a $k \times 1 \times 1$ filter, which has been proven to perform better, in both training and test loss [44, 46], as shown in Figure 1c. Also, we change the rectified linear unit (ReLU) [9] to its variant PReLU, in which the slopes of the negative part are learned from the data rather than predefined [12]. So the FRB can be formulated as:

$$F_d^L = F_{d-1}^L + W_{d,t}\left(W_{d,s}\left(\sigma\left(F_{d-1}^L\right)\right)\right), \tag{3.6}$$

where σ denoted the PReLU [12] activation function. $W_{d,s}$ and $W_{d,t}$ correspond to weights of the spatial convolution and the temporal convolution in FRB, respectively, where the bias term is not shown. In this way, the computational cost can be greatly reduced, which will be shown in Section 5.2. Consequently, we can build a larger, C3D-based model to directly video SR under limited computing resources with better performance.

3.3. Global residual learning

In this section, we describe the proposed global residual learning (GRL) on both LR and HR space. For SR tasks, input and output are highly correlated, so the residual connection between the input and output is wildly employed. However, previous works either perform residual learning on amplified inputs, which would lead to high computational costs, or perform residual connection directly on the input-output LR space, followed by upscaling layers for feature fusion and upsamping, which puts a lot of pressure on these layers.

To address these problems, we come up with global residual learning (GRL) on both LR and HR space, which mainly consists of two parts: LR space residual learning (LRL) and cross-space residual learning (CRL).

LR space residual learning (LRL) is introduced along with the FRBs in LR space. We apply a residual connection

with a followed parametric rectified linear unit (PReLU) [12] for it. Considering the high similarities between input frames, we also introduced a dropout [35] layer to enhance the generalization ability of the network. So the output F_{LRL}^L of LRL can be obtained by:

$$F_{LRL}^L = H_{LRL}\left(F_D^L, F_0^L\right) = \sigma_L\left(F_D^L + F_0^L\right), \quad (3.7)$$

where σ_L denoted the combination function of PReLU activation and dropout layer.

Cross-space residual learning (CRL) uses a simple SR mapping to directly map the LR video to HR space, and then adds to the LSRNet result F_{SR}^L, forming a global residual learning in HR space. Specifically, CRL introduces a interpolated LR to the output, which can greatly alleviate the burden on the LSRNet, helping improve the SR results. The LR mapping to HR space can be represented as:

$$F_{SR}^H = H_{CRL}\left(I_{LR}\right), \quad (3.8)$$

where F_{SR}^H is a super-resolved input mapping on HR space. H_{CRL} denotes the operations of the mapping function. The mapping function is selected to be as simple as possible so as not to introduce too much additional computational cost, including bilinear, nearest, bicubic, area, and deconvolution based interpolations.

The effectiveness of GRL and the selection of SR mapping method is demonstrated in Section 5.3.

3.4. Network learning

In training, we use l_1 loss function for training. To deal with the l_1 norm, we use the Charbonnier penalty function $\rho(x) = \sqrt{x^2 + \varepsilon^2}$ for the approximation.

Let θ be the parameters of network to be optimized, I_{SR} be the network outputs. Then the objective function is defined as:

$$\mathcal{L}\left(I_{SR}, I_{HR}; \theta\right) = \frac{1}{N}\sum_{n=1}^{N}\rho\left(I_{HR}^n - I_{SR}^n\right) \quad (3.9)$$

where N is the batch size of each training. Here we empirically set $\varepsilon = 1e-3$. Note that although the network produces the same frames as the input, we focus on the reconstruction of the center frame from the input frames in this work. As a result, our loss function is mainly related to the center frame of the input frames.

4. Theoretical analysis

In learning theory, we usually use generalization error to express the generalization capability of an algorithm, which is defined as the difference between the expected risk $\mathcal{R}$ and the empirical risk $\hat{\mathcal{R}}$ of the algorithm. In this section, we study the generalization ability of FSTRN. Specifically, we first give an upper bound for the covering number $\mathcal{N}(\mathcal{H})$ (covering bound) of the hypothesis space $\mathcal{H}$ induced by FSTRN. This covering bound constrain the complexity of FSTRN. Then we obtain an $O\left(\sqrt{\frac{1}{n}}\right)$ upper bound for the generalization error (generalization bound) of FSTRN. This generalization bound gives a theoretical guarantee to our proposed algorithms.

As Fig. 1c shows, FRB is obtained by adding an identity mapping to a chain-like neural network with one PReLU and two convolutional layers. Bartlett *et al*. proves that most standard nonlinearities are Lipschitz-continuous (including PReLU) [1]. Suppose the affine transformations introduced by the two convolutional operators can be respectively expressed by weight matrices A_1^i and A_2^i. Expect all FRBs, from the input end of the stem to the output end, there are 1 convolutional layer, 1 PReLU, 1 upscale, and 1 convolutional layer (we don't consider dropout here). They can be respectively expressed by weight matrix A_1, nonlinearity σ_1, weight matrix A_2, and weight matrix A_3. As Fig. 3 shows, LR residual learning is an identity mapping and HR residual learning can be expressed by a weight matrix A_{HR}. We can further obtain an upper bound for the hypothesis space induced by FSTRN as follows.

Theorem 1 (Covering bound for FSTRN)**.** *For the i-th FRB ($i = 1, \ldots, D$), suppose the Lipschitz constant of the PReLU is ρ^i, and the spectral norm of the weight matrices are bounded: $\|A_1^i\|_\sigma \le s_1^i$ and $\|A_2^i\|_\sigma \le s_2^i$. Also, suppose there are two reference matrices M_1^i and M_2^i respectively for A_1^i and A_2^i, which are satisfied that $\|A_i^i - M_i^i\|_\sigma \le b_i^i$, $i = 1, 2$. Similarly, suppose the spectral norm of weight matrices A_1, A_2, A_3, and A_{HR} are respectively upper bounded by s_1, s_2, s_3, and s_{HR}. Also, there are 4 corresponding reference matrices M_i, $i \in \{1, 2, 3, HR\}$ such that $\|A_i - M_i\| \le b_i$. Meanwhile, suppose the Lipschitz constant of nonlinearity σ_1 is ρ_1. Then, the ε-covering number satisfies that*

$$\begin{aligned}\mathcal{N}(\mathcal{H}) \le & \frac{b_1^2\|X\|_2^2\bar{\alpha}}{\varepsilon^2}\log\left(2W^2\right) + \sum_{d=1}^{D}\mathcal{N}_{FRB}(d) \\ & + (*)\frac{b_2^2}{\varepsilon_2^2}\log\left(2W^2\right)\left[\left(\frac{b_2}{\varepsilon_2}\right)^2 + \left(\frac{s_2 b_3}{\varepsilon_3}\right)^2\right] \\ & + \frac{b_{HR}^2\|X\|_2^2}{\varepsilon^2}\log\left(2W^2\right), \end{aligned} \quad (4.1)$$

where

$$\begin{aligned}\mathcal{N}_{FRB}(d) = & \left(\frac{\|X\|_2 s_1 \rho^d}{\varepsilon^d}\right)^2 \prod_{i=1}^{d}\left[\left(\rho^i s_1^i s_2^i\right)^2 + 1\right] \\ & \left[\left(b_1^d\right)^2\left(1 + s_2^d\right)^2 + \left(b_2^d s_1^d\right)^2\right], \end{aligned} \quad (4.2)$$

$$(*) = \left(\|X\|_2 s_1 \rho_1\right)^2 \prod_{d=1}^{D} \left[\left(\rho^d s_1^d s_2^d\right)^2 + 1\right], \quad (4.3)$$

$$\varepsilon^d = \frac{\varepsilon - s_{HR} - 1}{\bar{\alpha}} \prod_{i=1}^{d} \left[\rho^i (1 + s_1^i)(1 + s_2^i) + 1\right], \quad (4.4)$$

$$\varepsilon_2 = \frac{\varepsilon - s_{HR} - 1}{\bar{\alpha}} \left\{\prod_{i=1}^{D} \left[\rho^i (1 + s_1^i)(1 + s_2^i) + 1\right] + 1\right\} \rho_1(1 + s_2) + s_{HR} + 1, \quad (4.5)$$

and

$$\bar{\alpha} = \left\{\prod_{j=1}^{D} \left[\rho^j (1 + s_1^j)(1 + s_2^j) + 1\right]\right\} \rho_1(1 + s_2), \quad (4.6)$$

A detailed proof is omitted here and given in the appendix based on [2, 11]. Finally, we can obtain the following theorem. For the brevity, we denote the right-hand side (RHS) of eq. (4.1) as $\frac{R}{\varepsilon}$.

Theorem 2 (Generalization Bound for FSTRN)**.** *For any real $\delta \in (0, 1)$, with probability at least $1 - \delta$, the following inequality holds for any hypothesis F_θ:*

$$\mathcal{R}(F_\theta) \leq \hat{\mathcal{R}}(F_\theta) + \frac{8}{N^{\frac{3}{2}}} + \frac{36}{N}\sqrt{R}\log N + 3\sqrt{\frac{\log(2/\delta)}{2N}}. \quad (4.7)$$

Theorem 2 can be obtained from Theorem 1. A detailed proof is given in the appendix. Eq. (4.7) gives an $O\left(1/\sqrt{N}\right)$ generalization bound for our proposed algorithm FSTRN. Another strength of our result is that all factors involved do not explicitly rely on the size of our neural network, which could be extremely large. This strength can prevent the proposed result from meaninglessness. Overall, this result theoretically guarantees the feasibility and generalization ability of our method.

5. Experiments

In this section, we first analyze the contributions of the network and then present the experimental results obtained to demonstrate the effectiveness of the proposed model on benchmark datasets quantitatively and qualitatively.

5.1. Settings

Datasets and metrics. For a fair comparison with existing works, we used 25 YUV format benchmark video sequences as our training sets, which have been previously used in [14, 15, 28, 31, 38]. We tested the proposed model on the benchmark challenging videos same as [14] with the same settings, including the Dancing, Flag, Fan, Treadmill and Turbine videos, which contain complex motions with severe motion blur and aliasing. Following [5, 41], SR was only applied on the luminance channel (the Y channel in YCbCr color space), and performances were evaluated with the peak signal-to-noise ratio (PSNR) and structural similarity (SSIM) on the luminance channel.

Training settings. Data augmentation was performed on the 25 YUV video sequences dataset. Following [14, 15], to enlarge the training set, we trained the model in a volume-based way by cropping multiple overlapping volumes from the training videos. During the cropping, we took a large spatial size as 144×144 and the temporal step as 5, and the spatial and temporal strides were set as 32 and 10, respectively. Furthermore, inspired by [40], the flipped and transposed versions of the training volumes were considered. Specifically, we rotated the original images by 90° and flipped them horizontally and vertically. As a result, we could generate 13020 volumes from the original video dataset. After this, both of the training and testing LR inputs generating processes are divided into two stages: smoothing each original frame by a Gaussian filter with a standard deviation of 2, and downsampling the preceding frames using the bicubic method. In addition, to maintain the number of output frames equal to original video in the test stage, frame padding was applied at the test videos head and tail.

In these experiments, we focused on video SR of upscale factor 4, which is usually considered the most challenging and universal case in video SR. The number of FRBs and the dropout rate were empirically set to be 5 and 0.3. The Adam optimizer [22] was used to minimize the loss function with standard back-propagation. We started with a step size of $1e-4$ and then reduced it by a factor of 10 when the training loss stopped going down. The batch size was set depending on the GPU memory size.

Blocks	#Params	#FLOPs
C3DRB	$\sim$ 111K	$\sim$ 566M
FRB	$\sim$ 49K	$\sim$ 252M
Reduce ratio	55.86%	55.48%

Table 1: #Params and #FLOPs comparisons of one residual block using single C3D (Fig. 1b) and one FRB (Fig. 1c).

5.2. Study of FRB

In this section, we investigate the effect of the proposed FRB on efficiency. We analyze the computational efficiency of the FRB compared to the residual block built directly using C3D (C3DRB). Supposing we have all input and output feature-map size of 64, each input consists 5 frames with the size 32×32, then a detail params and floating-point operations (FLOPs) comparison of the proposed FRB and

Methods	Dancing PSNR / SSIM	Treadmill PSNR / SSIM	Flag PSNR / SSIM	Fan PSNR / SSIM	Turbine PSNR / SSIM	Average PSNR / SSIM
Bicubic	26.78 / 0.83	21.58 / 0.65	26.97 / 0.78	33.42 / 0.93	26.06 / 0.76	27.80 / 0.80
SRCNN[5]	27.91 / 0.87	22.61 / 0.73	28.71 / 0.83	34.25 / 0.94	27.84 / 0.81	29.20 / 0.84
SRGAN[25]	27.11 / 0.84	22.40 / 0.72	28.19 / 0.83	33.48 / 0.93	27.38 / 0.81	28.65 / 0.84
RDN[47]	27.51 / 0.82	22.69 / 0.72	28.62 / 0.82	34.46 / 0.93	28.10 / 0.82	29.30 / 0.84
BRCN[14]	28.08 / 0.88	22.67 / 0.74	28.86 / 0.84	34.15 / 0.94	27.63 / 0.82	29.16 / 0.85
VESPCN[3]	27.89 / 0.86	22.46 / 0.74	29.01 / 0.85	34.40 / 0.94	28.19 / 0.83	29.40 / 0.85
FSTRN(ours)	**28.66 / 0.89**	**23.06 / 0.76**	**29.81 / 0.88**	**34.79 / 0.95**	**28.57 / 0.84**	**29.95 / 0.87**

Table 2: Comparison of the PSNR and SSIM results for the test video sequences by Bicubic, SRCNN[5], SRGAN[25], RDN[47], BRCN[14], VESPCN[3], and our FSTRN with scale factor 4.

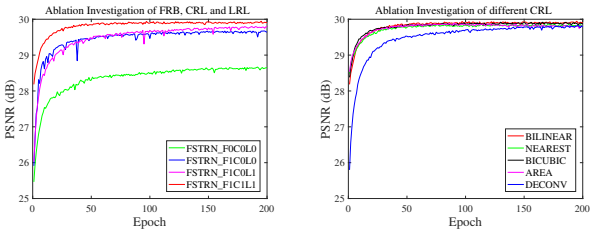
the C3DRB are summarized in Table 1. It's obvious to see that the FRB can greatly reduce parameters and calculations by more than half amount. In this way, the computational cost can be greatly reduced, so we can build a larger, C3D-based model to directly video SR under limited computing resources with better performance.

5.3. Ablation investigations

We conducted ablation investigation to analyze the contributions of FRBs and GRL with different degradation models in this section. Fig. 4a shows the convergence curves of the degradation models, including: 1) the baseline obtained without FRB, CRL and LRL (FSTRN_F0C0L0); 2) baseline integrated with FRBs (FSTRN_F1C0L0); 3) baseline with FRBs and LRL (FSTRN_F1C0L1); 4) baseline with all components of FRBs, CRL and LRL (FSTRN_F1C1L1), which is our FSTRN. The number D of FRBs was set to 5, and CRL uses bilinear interpolation.

The baseline converges slowly and performs relatively poor (green curve), and the additional FRBs greatly improve the performance (blue curve), which can be due to the efficient inter-frame features capture capabilities. As expected, LRL further improved network performance (magenta curve). Finally, the addition of CRL was applied (red curve), constituted GRL on both LR and HR space. It can be clearly seen that the network performed faster convergence speed and better performance, which demonstrated the effectiveness and superior ability of FRB and GRL.

Furthermore, to show how different interpolation methods in CRL affect the network performance, we investigated different interpolation method for CRL. Specifically, we explored bilinear, nearest, bicubic, area and deconvolution based interpolations. As shown in Fig. 4b, different interpolation method except deconvolution behaved almost the same, reason for this is because the deconvolution needs a process to learn the upsampling filers, while other methods do not need. All the different interpolation method converged to almost the same performance, indicated that the performance improvement of FSTRN is attributed to the introduction of GRL, and has little to do with specific interpolation method in CRL.



(a) (b)

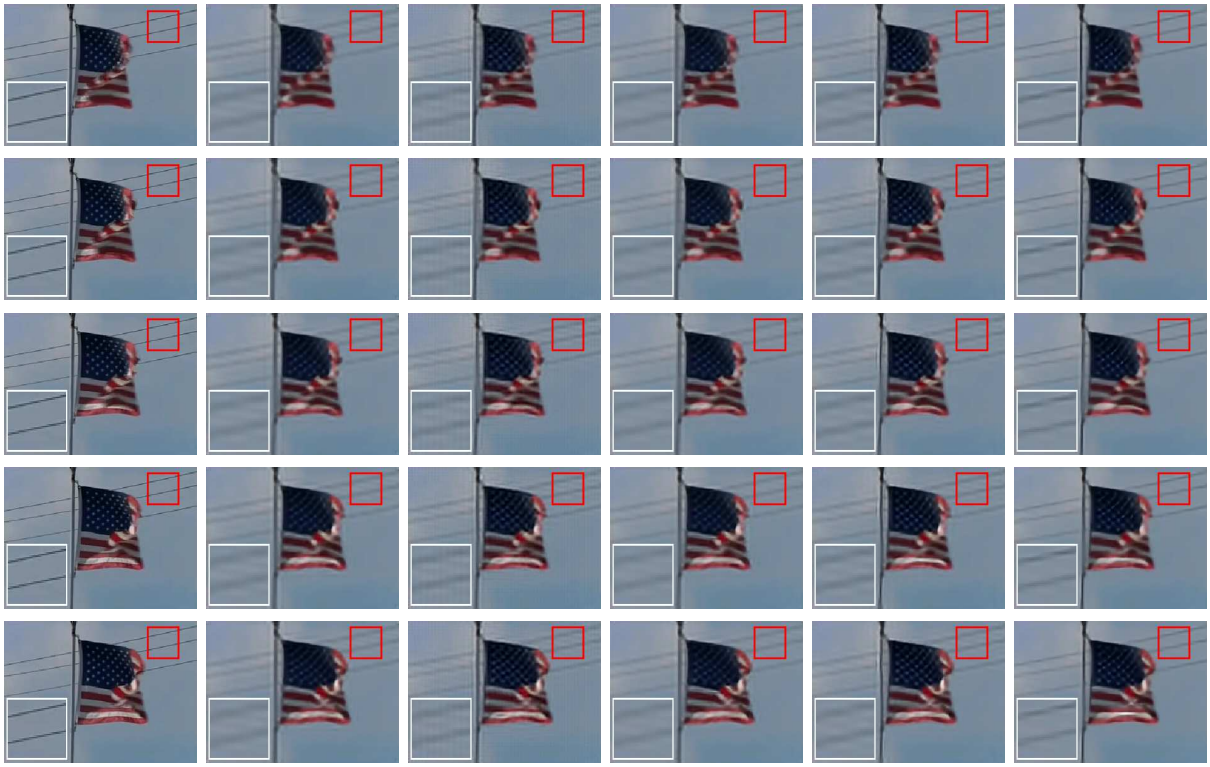
Figure 4: Convergence analysis on different degradation models (a) and different interpolation method for CRL (b). The curves for each combination are based on the PSNR on test video with scaling factor $\times 4$ in 200 epochs.

5.4. Comparisons with state-of-the-art

We compared the proposed method with different single-image SR methods and state-of-the-art multi-frame SR methods, both quantitatively and qualitatively, including Bicubic interpolation, SRCNN [5, 6], SRGAN [25], RDN [47], BRCN [14, 15] and VESPCN [3]. The number D of FRBs was set to 5 in following comparisons and the upscale method of CRL was set to bilinear interpolation.

The quantitative results of all the methods are summarized in Table 2, where the evaluation measures are the PSNR and SSIM indices. Specifically, compared with the state-of-the-art SR methods, the proposed FSTRN shows significant improvement, surpassing them 0.55 dB and 0.2 on average PSNR and SSIM respectively.

In addition to the quantitative evaluation, we present some qualitative results in terms of single-frame (in Figure 2) and multi-frame (in Figure 5) SR comparisons, showing visual comparisons between the original frames and the $\times 4$ SR results. It is easy to see that the proposed FSTRN recovers the finest details and produces most pleasing results, both visually and with regard to the PSNR/SSIM indices.



(a) Original (b) SRCNN (c) RDN (d) BRCN (e) VESPCN (f) FSTRN

Figure 5: Comparison between original frames ($1st \sim 5th$ frames, from the top row to bottom) of the Flag video and the SR results obtained by SRCNN, RDN, BRCN, VESPCN and FSTRN, respectively. Our results show sharper outputs with smoother inter-frame transitions compared to other works.

Our results show sharper outputs and even in grid processing, which is recognized as the most difficult to deal in SR, the FSTRN can handle it very well, showing promising performance.

6. Conclusion

In this paper, we present a novel fast spatio-temporal residual network (FSTRN) for video SR problem. We also design a new fast spatio-temporal residual block (FRB) to extract spatio-temporal features simultaneously while assuring high computational efficiency. Besides the residuals used on the LR space to enhance the feature extraction performance, we further propose a cross-space residual learning to exploit the similarities between the low-resolution (LR) input and the high-resolution (HR) output. Theoretical analysis provides guarantee on the generalization ability, and empirical results validate the strengths of the proposed approach and demonstrate that the proposed network significantly outperforms the current state-of-the-art SR methods.

7. Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grants 61822113, 41871243, 41431175, 61771349, the National Key R & D Program of China under Grant 2018YFA0605501, Australian Research Council Projects FL-170100117, DP-180103424, IH-180100002 and the Natural Science Foundation of Hubei Province under 2018CFA050.

References

[1] Peter L Bartlett, Dylan J Foster, and Matus J Telgarsky. Spectrally-normalized margin bounds for neural networks. In *NIPS*, pages 6240–6249, 2017.

[2] Peter L Bartlett and Shahar Mendelson. Rademacher and gaussian complexities: Risk bounds and structural results. *JMLR*, 3(Nov):463–482, 2002.

[3] Jose Caballero, Christian Ledig, Andrew P. Aitken, Alejandro Acosta, Johannes Totz, Zehan Wang, and Wenzhe Shi. Real-time video super-resolution with spatio-temporal net-

works and motion compensation. In *CVPR*, pages 2848–2857, 2017.

[4] Liujuan Cao, Rongrong Ji, Cheng Wang, and Jonathan Li. Towards domain adaptive vehicle detection in satellite image by supervised super-resolution transfer. In *AAAI*, pages 1138–1144, 2016.

[5] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In David J. Fleet, Tomás Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *ECCV*, volume 8692 of *Lecture Notes in Computer Science*, pages 184–199. Springer, 2014.

[6] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(2):295–307, Feb 2016.

[7] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *ECCV*, pages 391–407, 2016.

[8] Vincent Dumoulin and Francesco Visin. A guide to convolution arithmetic for deep learning. *CoRR*, abs/1603.07285, 2016.

[9] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Deep sparse rectifier neural networks. In Geoffrey J. Gordon, David B. Dunson, and Miroslav Dudík, editors, *AISTATS*, volume 15 of *JMLR Proceedings*, pages 315–323. JMLR.org, 2011.

[10] Jun Guo and Hongyang Chao. Building an end-to-end spatial-temporal convolutional network for video super-resolution. In Satinder P. Singh and Shaul Markovitch, editors, *AAAI*, pages 4053–4060. AAAI Press, 2017.

[11] Fengxiang He, Tongliang Liu, and Dacheng Tao. Why resnet works? residuals generalize. *CoRR*, abs/1904.01367, 2019.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *ICCV*, pages 1026–1034. IEEE Computer Society, 2015.

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016.

[14] Yan Huang, Wei Wang, and Liang Wang. Bidirectional recurrent convolutional networks for multi-frame super-resolution. In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, editors, *NIPS*, pages 235–243, 2015.

[15] Yan Huang, Wei Wang, and Liang Wang. Video super-resolution via bidirectional recurrent convolutional networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4):1015–1028, April 2018.

[16] Shuiwang Ji, Wei Xu, Ming Yang, and Kai Yu. 3d convolutional neural networks for human action recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(1):221–231, Jan 2013.

[17] Jiening Jiao, Wei-Shi Zheng, Ancong Wu, Xiatian Zhu, and Shaogang Gong. Deep low-resolution person re-identification. In *AAAI*, 2018.

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

[19] Armin Kappeler, Seunghwan Yoo, Qiqin Dai, and Aggelos K. Katsaggelos. Video super-resolution with convolutional neural networks. *IEEE Transactions on Computational Imaging*, 2(2):109–122, June 2016.

[20] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *CVPR*, pages 1646–1654, 2016.

[21] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Deeply-recursive convolutional network for image super-resolution. In *CVPR*, pages 1637–1645, 2016.

[22] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *CoRR*, abs/1412.6980, 2014.

[23] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. In Peter L. Bartlett, Fernando C. N. Pereira, Christopher J. C. Burges, Léon Bottou, and Kilian Q. Weinberger, editors, *NIPS*, pages 1106–1114, 2012.

[24] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, Nov 1998.

[25] Christian Ledig, Lucas Theis, Ferenc Huszar, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew P. Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, and Wenzhe Shi. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, pages 105–114, 2017.

[26] Renjie Liao, Xin Tao, Ruiyu Li, Ziyang Ma, and Jiaya Jia. Video super-resolution via deep draft-ensemble learning. In *ICCV*, pages 531–539, 2015.

[27] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *CVPRW*, pages 1132–1140. IEEE Computer Society, 2017.

[28] Ce Liu and Deqing Sun. On bayesian adaptive video super resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(2):346–360, Feb 2014.

[29] Ding Liu, Zhaowen Wang, Yuchen Fan, Xianming Liu, Zhangyang Wang, Shiyu Chang, Xinchao Wang, and Thomas S. Huang. Learning temporal dynamics for video super-resolution: A deep learning approach. *IEEE Transactions on Image Processing*, 27(7):3432–3445, July 2018.

[30] Ding Liu, Zhaowen Wang, Bihan Wen, Jianchao Yang, Wei Han, and Thomas S. Huang. Robust single image super-resolution via deep networks with sparse prior. *IEEE Transactions on Image Processing*, 25(7):3194–3207, July 2016.

[31] Matan Protter, Michael Elad, Hiroyuki Takeda, and Peyman Milanfar. Generalizing the nonlocal-means to super-resolution reconstruction. *IEEE Transactions on Image Processing*, 18(1):36–51, Jan 2009.

[32] Mehdi SM Sajjadi, Raviteja Vemulapalli, and Matthew Brown. Frame-recurrent video super-resolution. In *CVPR*, pages 6626–6634, 2018.

[33] Wenzhe Shi, Jose Caballero, Ferenc Huszar, Johannes Totz, Andrew P. Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *CVPR*, pages 1874–1883, 2016.

[34] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[35] Nitish Srivastava, Geoffrey E. Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

[36] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott E. Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, pages 1–9, 2015.

[37] Ying Tai, Jian Yang, Xiaoming Liu, and Chunyan Xu. Memnet: A persistent memory network for image restoration. In *ICCV*, pages 4549–4557, 2017.

[38] Hiroyuki Takeda, Peyman Milanfar, Matan Protter, and Michael Elad. Super-resolution without explicit subpixel motion estimation. *IEEE Transactions on Image Processing*, 18(9):1958–1975, Sept 2009.

[39] Xin Tao, Hongyun Gao, Renjie Liao, Jue Wang, and Jiaya Jia. Detail-revealing deep video super-resolution. In *ICCV*, pages 4482–4490, 2017.

[40] Radu Timofte, Rasmus Rothe, and Luc Van Gool. Seven ways to improve example-based single image super resolution. In *CVPR*, pages 1865–1873, 2016.

[41] Radu Timofte, Vincent De Smet, and Luc J. Van Gool. Anchored neighborhood regression for fast example-based super-resolution. In *ICCV*, pages 1920–1927. IEEE Computer Society, 2013.

[42] Tong Tong, Gen Li, Xiejie Liu, and Qinquan Gao. Image super-resolution using dense skip connections. In *ICCV*, pages 4809–4817, 2017.

[43] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. Learning spatiotemporal features with 3d convolutional networks. In *ICCV*, pages 4489–4497, 2015.

[44] Du Tran, Heng Wang, Lorenzo Torresani, Jamie Ray, Yann LeCun, and Manohar Paluri. A closer look at spatiotemporal convolutions for action recognition. *CoRR*, abs/1711.11248, 2017.

[45] Zhaowen Wang, Ding Liu, Jianchao Yang, Wei Han, and Thomas S. Huang. Deep networks for image super-resolution with sparse prior. In *ICCV*, pages 370–378, 2015.

[46] Saining Xie, Chen Sun, Jonathan Huang, Zhuowen Tu, and Kevin Murphy. Rethinking spatiotemporal feature learning for video understanding. *CoRR*, abs/1712.04851, 2017.

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