Video super-resolution with non-local alignment network

Chao Zhou¹,³ | Can Chen¹,³ | Fei Ding²,³ | Dengyin Zhang²,³

¹ School of Telecommunications & Information Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
² School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210003, China
³ Jiangsu Key Laboratory of Broadband Wireless Communication and Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

Correspondence
Dengyin Zhang, School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing, 210003, China.
Email: zhangdy@njupt.edu.cn

Abstract
Video super-resolution (VSR) aims at recovering high-resolution frames from their low-resolution counterparts. Over the past few years, deep neural networks have dominated the video super-resolution task because of its strong non-linear representational ability. To exploit temporal correlations, most deep neural networks have to face two challenges: (1) how to align consecutive frames containing motions, occlusions and blurring, and establish accurate temporal correspondences, (2) how to effectively fuse aligned frames and balance their contributions. In this work, a novel video super-resolution network, named NLVSR, is proposed to solve above problems in an efficient and effective manner. For alignment, a temporal-spatial non-local operation is employed to align each frame to the reference frame. Compared with existing alignment approaches, the proposed temporal-spatial non-local operation is able to integrate the global information of each frame by a weighted sum, leading to a better performance in alignment. For fusion, an attention-based progressive fusion framework was designed to integrate aligned frames gradually. To penalize the points with low-quality in aligned features, an attention mechanism was employed for a robust reconstruction. Experimental results demonstrate the superiority of the proposed network in terms of quantitative and qualitative evaluation, and surpasses other state-of-the-art methods by 0.33 dB at least.

1 | INTRODUCTION

In this paper, we focus on resolving video super-resolution (VSR) via a deep learning method. VSR [1, 2] refers to the reconstruction of high-resolution (HR) video sequences from their low resolution (LR) counterparts, which is widely used in high-definition television (HDTV), human face hallucination [3–5], remote sensing [6–8] etc. Compared with single image super-resolution (SISR), VSR was provided with huge temporal redundancies existing in consecutive frames, which is crucial for the success of SR reconstruction. With dramatic developments in deep learning (DL), deep neural networks (DNNs) [9, 10] have been one of the most popular approaches to solving the VSR problems over the past few years.

For single image super-resolution (SISR), recent researches [11–19] mainly paid attention to designing an effective network structure to fully exploit the inherent information from LR images. In [11], Dong et al. firstly introduced the convolutional neural network (CNN) into SISR to build a direct mapping from LR images to HR images, which achieved a significant improvement in reconstruction quality over conventional approaches [20–23]. Following this work, a number of specific network structures have been proposed in solving the SISR problem, such as residual network [24], dense network [25], attention mechanism [26] etc.

For VSR, existing networks typically desire effective alignment and fusion modules before the reconstruction to make full use of the temporal information from video signals and address occlusions, fast motions and blurring among neighbouring frames.

1.1 | Alignment

The alignment module in VSR aims at establishing accurate temporal correspondences among consecutive frames. Some researches performed explicit motion estimation and motion compensation (ME&MC) on input LR frames to capture the temporal dependency [27–36]. Specifically, the optical flow was estimated from consecutive frames, and then used to warp...
them to make sure the temporal consistency with the reference frame (i.e. the central frame to be recovered). Some approaches [27, 28] employed off-the-shelf ME&MC algorithms [37–39] before their networks, but these independent motion estimation methods made it difficult to find a global optimal solution for SR. Besides, these off-the-shelf ME&MC algorithms were generally time-consuming as compared to the inference speed of DNNs. Therefore, some end-to-end networks [29–36] were proposed to integrate ME&MC as the part of the network at the cost of introducing extra training difficulty. In general, ME&MC-based approaches have two main disadvantages: (1) It would be highly challenging for them to estimate an accurate motion field, for instance, in the case of large-scale motions, and (2) even with a high-quality estimated optical flow, warped frames also suffer from severe artifacts caused by the motion compensation, which would be propagated into the final super-resolved frames.

Therefore, some recent researches [40–45] tried to perform alignment in a non-ME&MC or implicit ME&MC manner. DUF [40] utilized learned dynamic up-sampling filters and residual images to super-resolved LR input frames. The 3D convolution layer used in this network effectively modelled the temporal correlation among neighbouring frames, but at the expense of a high computation complexity. TDAN [41] introduced deformable convolution to replace explicit ME&MC in their alignment module. Inspired by this work, EDVR [42] integrated pyramidal processing and cascading refinement into deformable convolution operation to tackle complex motions and large parallax problems. Recently, non-local block [46] was also adopted in VSR to capture long-range dependencies among frames [43]. This operation calculates the response at one position as the weighted sum of all positions in the feature maps. Yi et al. [43] claimed that the non-local operation shared a similar purpose as ME&MC in VSR, and reported the state-of-the-art performance with the help of non-local block. This proved non-local block a promising method in modelling temporal dependency. However, the non-local block in [43] follows the original idea proposed in [46], which means that it is not aiming at aligning each frame to the reference frame, but tries to enhance each frame itself using the temporal correlation. Intuitively, neighbouring frames are expected to provide the reference frame with supporting information instead of the information about themselves. Therefore, this may leave some room for improvement in exploiting non-local operation for alignment.

1.2 Fusion

After aligning each neighbouring frame, another crucial problem is how to fuse them in a proper way. A straightforward approach is using one convolution layer to integrate them directly, but this direct fusion process typically results in a limited performance. Recently, some researches [33, 43, 47] proved that fusing multiple frames in a progressive framework could provide a better restoration quality. However, approaches mentioned above all treat each frame equally without considering the visual informativeness of each frame. This means that some locations or frames with low quality make the same contribution as others to the result. To address this, an attention mechanism was applied in [42] to measure how informative one point is, and then assigned different weights to them. Although this approach succeed in introducing attention mechanism to the fusion process, it is essentially realized in a direct fusion manner.

In this work, we propose an end-to-end network for VSR, named NLVSR, to address problems in existing alignment and fusion modules. For alignment, a temporal-spatial non-local (TSNL) block is designed to capture the temporal dependency among neighbouring frames and the spatial dependency in the reference frame itself. Different from the non-local block used in [43], the proposed TSNL block aims at aligning each frame (including the reference frame itself) to provide supporting information for the reconstruction of reference frame. All points on one frame are summarized with carefully designed weights to represent the target point on the reference frame. For fusion, an attention-based progressive fusion module (APF) was proposed to integrate aligned frames gradually. The proposed APF module consists of a series of attention-based progressive fusion blocks (APFBs). Each of them fuses multiple aligned frames with the help of temporal attention, and then return the fused features back to refine each frame. After several A, the LR-fused features are up-scaled to generate the HR output. Ablation study is performed to verify the effectiveness of TSNL and APF module. Comparative experiments demonstrate that the proposed NLVSR network yields the state-of-the-art performance on SR reconstruction quality.

The main contributions of our work are listed as follows:

(1) Different from the previous non-local operation used in VSR whose aim is enhancing each frame itself, we design a novel non-local block, i.e. TSNL block, to align each frame to the reference frame. This non-local operation is capable of capturing the long-range dependency between each neighbouring frame and the reference frame, and extracting supporting information from the neighbouring frames to help the reconstruction of the central frame. We also provide a detailed comparison between the proposed TSNL and other alignment approaches in Section 3 to explain the superiority of TSNL.

(2) We introduce the temporal attention mechanism to a progressive fusion framework. Compared with direct fusion, this progressive framework behave better in making full extraction of temporal information among multiple aligned features. In addition, the attention mechanism in each PFRABs offers a high degree of flexibility in dealing with different quality of information, avoiding the equal treatment on them.

The rest of this paper is organized as follows. In Section 2, we briefly review the related works of SR and non-local operation. Section 3 presents a detailed illustration for NLVSR and a comparative discussion between several state-of-the-art methods. In Section 4, experimental results are provided to verify the superiority of our approach. At last, we draw a conclusion in Section 5.
2 | RELATED WORKS

2.1 | Single image super-resolution with CNNs

Over the past few years, CNN-based methods have dominated the single image/video super-resolution tasks because of its strong non-linear representational ability. It was firstly introduced to SISR in the form of a three layer CNN [11] to learn an end-to-end mapping from LR images to HR ones. However, the shallow architecture limited the performance of the network. To address this, Kim et al. proposed VDSR [12] and DRCN [13] with 20 layers for more contextual information usage of input images. In addition, the residual learning [24] was introduced in these two networks to overcome the training difficulty caused by deep network structure. Tai et al. took a further step toward deeper architecture by introducing recursive blocks in DRRN [14], achieving an improved performance on SR quality. Note that all of these approaches took interpolated HR images as input, which resulted in a huge computation complexity and over-smoothed results. To address this, Shi et al. [16] shifted the up-scaling step to the end of the network with an efficient sub-pixel convolution layer. This specialized convolution layer could up-scale the LR features into the final HR output, allowing the feature extraction before it to be performed in a LR spatial size. This efficient method has been widely adopted not only in the following networks for SISR but also in recent VSR methods. Besides simply stacking convolution layers to reach deeper architecture, some studies achieved significant improvements by making full use of extracted features. In [17], a dense connection network was employed so that each layer was able to receive information from all previous layers. This means that the network could reuse low-level features to provide more information [25]. In [18], Zhang et al. further combined local/global residual learning and dense connection to reach an improved performance. In addition to reusing previous features, Zhang et al. also suggested that channel-wise features with high frequency should be paid more attention because they were more informative than ones with low frequency for HR restoration, and thus they introduced a channel-attention mechanism in [19], which achieved a remarkable SR performance. Refer to [48] for more information about SISR.

2.2 | Video super-resolution with CNNs

Compared with SISR, VSR task is provided with multiple consecutive frames which are rich in temporal redundancy. Thus, the main challenge of VSR methods lies in how to exploit the temporal correlation among video frames. One popular solution in the literature was to align and warp consecutive frames and then feed all of them into an SR network. Following this idea, Liao et al. [27] and Kappeler et al. [28] performed ME&MC directly with off-the-shelf optical flow approaches [37, 38, 39], and then fed the compensated frames to their SR networks. One problem with these two methods is that the motion estimation is independent of the SR network, which means that it is difficult to find a global optimal solution for HR restoration. In addition, compared with the inference speed of DNNs, classical optical flow approaches are usually computationally expensive. Therefore, recent studies [29–36] preferred an end-to-end structure with integrated ME/MC. The first end-to-end VSR network was proposed in [29], which was composed of a spatial transformer network for motion estimation and a spatio-temporal ESPCN [16] for super-resolution. Similarly, Tao et al. [30] exploited sub-pixel information by a sub-pixel motion compensation (SPMC) layer, and then designed an encoder-decoder network with LSTM to recover the centre frame. Liu et al. [31] designed an adaptive architecture to accept different numbers of LR inputs and a robust spatial alignment network to align them. In [32], a task-oriented flow method was proposed to achieve a better recover quality in different contexts. Sajjadi et al. [33] created a frame-recurrent architecture which utilized the last super-resolved frame to help the restoration of the current frame. In [34], a multi-memory network was proposed to extract temporal information from consecutive frames with several ConvLSTM layers [49]. In order to avoid the resolution conflict between LR optical flows and HR outputs, Wang et al. [35] designed a novel VSR network to estimate HR optical flow in a coarse-to-fine manner. Li et al. [36] proposed a motion compensation network with a pyramid structure and adopted channel and spatial attention mechanism in the following reconstruction network. Note that methods above all take explicit ME/MC as part of their network. However, it is quite difficult to train an integrated and often low-capacity ME/MC layer under the current state-of-the-art schemes. Therefore, some studies tried to perform motion estimation in an implicit but more effective manner. Jo et al. [40] utilized 3D convolution layers to model temporal correlations among frames in their network. Tian et al. [41] and Wang et al. [42] introduced deformable convolution layer to replace explicit ME/MC in their alignment module. Wang et al. [42] also proposed a specialized attention module to emphasize important features along spatial and temporal dimension in feature fusion. Yi et al. [43] utilized the non-local block to capture temporal dependency and designed a progressive architecture to fuse aligned frames gradually.

2.3 | Non-local block

Non-local block [46] was designed to capture the long-range dependencies among features, which was inspired by the classical non-local means algorithm [43]. Figure 1(a) is a basic illustration of non-local block, where $T$, $H$, $W$ and $C$ represent the frame number, height, width and channels of video signals, respectively. Mathematically, the response at a position in non-local operation could be defined as:

$$y_i = \frac{1}{C(x)} \sum_{y_j} f(x_i, x_j)g(y_j).$$

(1)

where $x$ and $y$ refer to the input and output data, which have the same size. $i$ is defined as the index of current position to be computed, and $j$ represents the index at any possible locations in $x$. 

FIGURE 1 Examples for different non-local blocks. The feature maps are denoted in the form of their shapes, such as $T \times H \times W \times C$. Proper reshaping is performed when needed. Each blue block represents one convolutional layer whose kernel size is shown in the block. (a) The original non-local block proposed in [46]. (b) The modified non-local block for VSR in [43]. The input and output are the concatenations of all frames. (c) The proposed TSNL block. The input only includes one neighbouring frame and the reference frame, and the output is the aligned neighbouring frame.

where $\theta(\cdot)$ and $\phi(\cdot)$ are two embeddings which could be realized by convolutional layers. In summary, the main idea behind the non-local block is to compute the response at one point as the weighted sum of features at all positions. The weight for each feature is decided by its correlation with the target point.

Yi et al. [43] suggested that the non-local block has a similar function with ME&MC since both of them could capture the temporal dependency across frames. Therefore, they introduced and improved non-local block to VSR task as shown in Figure 1(b). One problem with non-local block in VSR is the huge $THW \times THW$ matrix generated by $\theta(x_i)^T \phi(x_j)$ (denoted as $R$ in Figure 1). For example, when we take five LR frames with size of $480 \times 270$ as input, the dimension of $R$ would be $648,000 \times 648,000$, which might be prohibitive to be calculated or stored in many currently available GPUs. Thus, two measures are introduced in [43] to make non-local block more lightweight: (1) Combine the time dimension with the channel dimension (i.e. reshape the input from $T \times H \times W \times C$ to $H \times W \times CT$), which makes the matrix $R$ not related to the frame number any more, and (2) reduce the space dimension to deepen the channel dimension with a down-scaling factor $r$ (i.e. reshape the input from $H \times W \times CT$ to $H/r \times W/r \times CT/r^2$). As a result, the matrix $R$ finally becomes a matrix of size $(HW/r^2 \times CT) \times (HW/r^2 \times CT)$, which is much smaller than $THW \times THW$.

3 | METHODOLOGY

We first describe the architecture of the proposed NLVSR network. As shown in Figure 2, the input of NLVSR is composed with $2N + 1$ consecutive LR frames, where the central frame (i.e. the reference frame) is the target frame to be super-resolved, and the others are neighbouring frames used for providing supporting information. The whole network consists of four parts: feature extraction module, temporal-spatial non-local (TSNL) module, attention-based progressive fusion (APF) module and up-scaling module. At the beginning of the network, several stacked convolution layers are used to extract features from each input frame, as shown in Figure 3. Then, the TSNL module aligns each frame to the reference frame by a specialized non-local block. After that, the aligned features are fused in APF module gradually with the help of temporal attention mechanism. At the end of the network, the fused features are up-scaled to a HR frame using the sub-pixel magnification layer from [16]. The final SR frame is obtained by adding this HR frame with the bicubically up-sampled frame. The details of TSNL module and APF module are presented in following subsections.

3.1 | Alignment with temporal-spatial non-local block

In this work, we proposed a novel non-local block, named TSNL block, for alignment in VSR. The TSNL block is performed frame by frame, which means the input of it only involves the extracted features from the reference frame $F_t$ and one neighbouring frame $F_{t+n}$ ($|n| \leq N$). The non-local
The architecture of the proposed NLVSR. The output is the super-resolved reference frame. The blue blocks represent the main modules in NLVSR, and the green tensors refer to the output of each module.

$y_i = \frac{1}{C(x)} \sum_j f(r_i, x_j) g(x_j)$,  \hspace{1cm} (3)

where $r_i$ and $x_j$ represent the points on $F_t$ and $F_{t+n}$, respectively, and $y_i$ is the response at position $i$. We choose the embedded Gaussian function for $f(\cdot)$ and a linear function for $g(\cdot)$. The function $C(x)$ is set as $\sum_j f(r_i, x_j)$ so that the normalization can be realized by a softmax layer in practical implementation [46].

As shown in Figure 1(c), we adopt a similar measure as [43] in TSNL block to avoid a huge matrix $R$, which reduces the space dimension with a down-scaling factor $r$ (i.e. the second measure mentioned in Section 2.3). Thus, as shown in Figure 1(c), $F_t$ and $F_{t+n}$ are reshaped to the size of $(H/r) \times (W/r) \times C r^2$ at first. After that, $f(r_i, x_j)$ is used to generate the weight for each point in the neighbouring frame. According to (3), $f(\cdot)$ is a pairwise function computing the correlation between $r_i$ and $x_j$. If $x_j$ is close to $r_i$, $g(x_j)$ will be encouraged with a large weight $f(r_i, x_j)$. In summary, the non-local operation in TSNL integrates the global information of the neighbouring frame $F_{t+n}$ using a weighted sum of $g(x_j)$, and pushes the response $y_i$ to approach to the target point $r_i$ on the reference frame. This is consistent with the purpose of most alignment methods in VSR, which aligns neighbouring frames to establish an accurate correspondence with the reference frame. To better understand the behaviour of the proposed TSNL block, a visual illustration is presented in Figure 4(d). At last, the output of the TSNL block, namely the aligned feature $F_{t+n}$, is obtained by:

$z_i = w(y_i) + x_j$, \hspace{1cm} (4)

where $w(\cdot)$ is a linear mapping realized by a $1 \times 1$ convolution layer, “$+ x_j$” refers to a residual connection in the non-local block which is the same as [46]. Although the TSNL block is applied to each frame, the parameters are shared across them so that the model size is not related to the number of input frames.

In addition to temporal dependency, TSNL block is also used to exploit spatial dependency in the reference frame itself. In other words, we also allow $n = 0$ in $F_{t+n}$ so that the $x_j$ in Equation (3) comes from the reference frame at that case. The result features $F_a$ provide an access to exploiting the self-similarity of the reference frame, which is similar with some non-local matching methods [50–52] applied in image processing. The effectiveness of this self-similarity is discussed in Section 4. Due to the participation of $F_a$, the number of aligned features fed into the following fusion module is $2N + 2$ in total, including $2N + 1$ aligned frames $F_{t+n}$ and the original reference frame $F_t$.

Here, we also offer detailed comparisons between the proposed TSNL module and several alignment methods.

(1) TSNL vs explicit ME&MC: ME&MC methods [27–36] assume that the optical flow between two frames is pixel-wise dense, which means each pixel could be displaced at a new position on the reference frame [43]. After arranging each pixel, interpolation is conducted to make sure each pixel back onto a regular grid. Figure 4(a) gives a visual illustration of ME&MC. In this figure, the new position of each pixel on the reference frame is assumed to be a regular grid for a convenient explanation.
the response at one position is derived by summarizing all points with carefully designed weights, which would weaken the influence of these ‘bad’ points.

(2) TSNL vs deformable convolution: Deformable convolution is employed in [41, 42] as an implicit ME&MC method for alignment. In this layer, each point \( p_0 \) on the aligned frame \( F_{t+n}^a \) could be derived by:

\[
F_{t+n}^a(p_0) = \sum_{k=1}^{K} w_k \cdot F_{t+n}^a(p_0 + p_k + \Delta p_k) ,
\]

where \( p_k \in \{(-1,-1), (-1,0), \cdots, (1,1)\} \) is the pre-specified offsets for \( K \) sampling positions, and \( \Delta p_k \) is the learnable offsets predicted from the concatenated frames \( [F_{t+n}, F_t] \). Figure 4(b) presents a visual illustration for deformable convolution. Actually, deformable convolution could be viewed as a compromise between explicit ME&MC and the TSNL block. To represent one point on the reference frame, ME&MC methods only involve one point from the neighbouring frame, the deformable convolution involves several points around the target position as shown in Figure 4(b), and the TSNL block involves all points on the neighbouring frame. Therefore, compared with deformable convolution, TSNL block has following advantages: (1) Deformable convolution obtains information from several points in a local receptive field, whereas TSNL block could aggregate the global information from the whole neighbouring frame, (2) deformable convolution learns to select several suitable points to represent the target point on the reference frame, which is similar to motion estimation (only one point selected). Both of them are challenging. However, TSNL do not have to solve this problem.

(3) TSNL vs NLRB: In [43], a modified non-local block (hereinafter referred to as “NLRB”) was proposed to replace ME&MC in VSR. It is reported that this non-local operation do not aims at aligning neighbouring frames but capturing the long-range dependencies across frames. Figure 4(c) presents a visual illustration of NLRB. Although the proposed TSNL block and NLRB [22] both belongs to the non-local operation, the core ideas of them are fundamentally different. As shown in Figure 4(d), the main target of TSNL block is to align each frame to the reference frame. Thus, only the points on the current neighbouring frame (e.g. \( F_{t+n} \) in Figure 4(d)) are considered. Besides, the weight for each of them is decided by its correlation with the target point on the reference frame (e.g. the point \( p_t \) on the reference frame \( F_t \) shown in Figure 4(d)). However, as shown in Figure 4(c), NLRB involves the points from all frames when calculating the response for one frame (e.g. the frame \( F_{t+n}^a \) in Figure 4(c)), and the weight for each point is decided by its correlation with the target point on the current frame (e.g. the point \( x_j \) on the frame \( F_{t+n} \) shown in Figure 4(c)), not the reference frame. This means NLRB just utilizes the long-range dependencies across frames to refine each frame. Intuitively, we hope neighbouring frames provide supporting information for the reference frame instead of the information about themselves. Therefore, we believe that the TSNL block is more promising in the alignment.
Progressive fusion with temporal attention

In this section, we introduce an attention-based progressive fusion (APF) module to integrate multiple aligned features. This module is composed of a series of APFBs. Each of them fuses the temporal information from multiple frame features and then, in turn, utilizes the fused information to refine each aligned frame. At last, the features of all frames are concatenated as the output of this module.

The structure of one APFB is shown in Figure 5, which is designed in a multi-channel fashion. The input of this block includes $2N + 2$ features ($2N + 1$ from the aligned frames and one from the reference frame). After the first convolution layer, all of them are integrated through a temporal attention block, which can be described as:

$$I_{t+n}^1 = C_1(I_{0}^t + n), \quad n \in [-N, N]$$  \hspace{1cm} (6)

$$I^r = C_1(I^0)$$  \hspace{1cm} (7)

$$I_a = F_A(I_{t-n}, \ldots, I_{t+n}, I^r)$$  \hspace{1cm} (8)

where $I_{t+n}^1$ and $I^0$ refer to the features from $2N + 1$ aligned frames and the reference frame, respectively. $C_1(\cdot)$ represents the first convolution layer in APFB. The temporal attention block is denoted as $F_A(\cdot)$, and $I_a$ is the integrated features obtained from it. Then, this integrated information is concatenated with each frame feature and output after the second convolution. A skip connection is applied in each channel for residual learning. The final output of APFB could be defined as:

$$O_{t+n} = I_{t+n} + C_2([I_{t+n}^1, I_a])$$  \hspace{1cm} (9)

where $C_2(\cdot)$ represents the second convolution layer and $[\cdot, \cdot]$ refers to the concatenation operation. Note that $C_1(\cdot)$ and $C_2(\cdot)$ in different channels share the same convolution kernel in order to reduce the parameters. The output of one APFB is also the input of the next APFB. At the last APFB, the output features of each frame $[O_{t-N}, \ldots, O_{t+N}, O_r]$ are concatenated as the final output of the module.

When integrating the information from multiple frames, we employ a temporal attention mechanism, i.e. $F_A(\cdot)$ in Equation (8), based on the observation that pixels from different frames are not informative equally. Some of them contain less information for reconstruction or are affected by the alignment error, which may lead to a bad influence on the following restoration. Thus, the main target of this attention mechanism is evaluating each pixel from different frames and penalizing the pixels with low quality. Figure 6 illustrates how this temporal attention works. For each frame, we use the reference frame as a criterion, and evaluate the quality of each pixel based on its similarity with the point at the same location on the reference frame. As shown in Figure 6, the similarity is defined as:

$$S(I^t_{t+n}) = \text{sigmoid}(p(I^t_{t+n})^T q(I^t_{t+n}))$$  \hspace{1cm} (10)
where \( p(\cdot) \) and \( q(\cdot) \) are two embeddings, and the \textit{sigmoid}(\cdot) function is used to limit the outputs in \([0, 1]\). Then, this similarity would be multiplied to the feature \( I_{t+\sigma} \) in a pixel-wise manner:

\[
I'_{t+\sigma} = S(I_t, I'_{t+\sigma}) \odot I_{t+\sigma},
\]

where \( \odot \) represents the pixel-wise product. This operation enforces a penalizing weight onto each pixel according to its similarity with the corresponding pixel on the reference frame. At last, a convolutional layer is used to integrate all frames into \( I_{t+\sigma} \), as shown in the top of Figure 6. Note that the temporal attention has a similar architecture with non-local block. However, the targets of them are completely different: for non-local block, the main goal is computing a weighted sum of all points to capture the dependency between the reference frame and one neighbouring frame; for temporal attention, the main goal is penalizing the point with low quality by a weight smaller than 1. There is no normalization in temporal attention.

In summary, we design a progressive fusion framework by stacking several APFBs. Each block is able to further fuse the information from multiple aligned frames. In addition, the temporal attention mechanism helps them to penalize the pixel with low quality, which further improves the reconstruction performance. The effectiveness of this fusion strategy is verified in Section 4.

**4 | EXPERIMENTS**

**4.1 | Implementation details**

There are five stacked convolutional layers in feature extraction module, whose kernel size is set as \( 3 \times 3 \). In APF module, there are 20 APFBs in total. The following up-scaling module is the same as \([16]\). We set the channel size as 64 in the whole network except the up-scaling module. The leaky ReLU \([53]\) is applied as the activation function following each convolution layer. The number of input frames is set as 7 as default (i.e. \( N = 3 \)), and the output is only the central frame. During training, we employ Charbonnier penalty function \([54]\) as the loss function, which could be formulated as:

\[
L_{SR} = \sqrt{\|H - SR(I)\|^2 + \varepsilon^2},
\]

where \( I \) refers to the LR input frames, \( H \) represents the target reference frame, \( SR(\cdot) \) denotes the SR network, and \( \varepsilon \) is a constant whose value is \( 1 \times 10^{-3} \). The spatial size of training frames is set as \( 32 \times 32 \), and the batch size is set as 16. We adopt Adam optimizer \([55]\) with \( \beta_1 = 0.9 \) and \( \beta_2 = 0.999 \) to train the proposed network. The initial learning rate is set as \( 5 \times 10^{-4} \) and decays to \( 1 \times 10^{-5} \) finally. We pre-train our network without temporal attention in APF module at first, and then use the pre-trained parameters as the initialization to train the whole network. Refer to \([9, 10]\) for more details about networks training and testing. The whole experiment is conducted in TensorFlow with one NVIDIA Titan Xp GPU.

**4.2 | Datasets**

For training, we use the video sequences collected in \([34]\), which include 522 HD video clips for training and 20 for validation. Most of them are chosen from documentaries with little post-processing. For testing, we choose two representative datasets: Vid4 \([29]\) and udm10 \([33, 56]\). The Vid4 dataset is widely used for VSR evaluation. It includes four video sequences whose spatial sizes are smaller than \( 720 \times 576 \). The udm10 dataset has a larger spatial size \( (1272 \times 720) \) and is composed of 10 video sequences.

Following \([40, 43]\), we employ a \( 13 \times 13 \) Gaussian blur kernel on each HR frame, and set the standard deviation as 1.6. Then, the LR input frames are obtained by down-sampling these blurry HR frames. To enhance the performance of the network, the data augmentation from \([18, 57]\) is applied during the training process, which randomly flips and rotates input frames before they are fed into the network.

**4.3 | Comparisons with state-of-the-art methods**

Here, we compare our network with nine state-of-the-art methods in VSR: VESPCN \([29]\), RVSR-LTD \([31]\), MCResNet \([58]\), DRVSR \([30]\), FRVSR \([33]\), DUF_52L \([40]\), RBPN \([59]\), EDVR \([42]\), PFNL \([43]\). Note that most of them \([29, 31, 42, 58, 59]\) obtain LR frames by bicubically down-sampling HR frames, whereas some recent methods \([30, 33, 40, 43]\) involved a Gaussian blur to the HR images before down-sampling. Clearly, the latter strategy is closer to the reality. However, different downsampling approaches make the fair comparison difficult since the relationships between LR and HR frames are changed. In order to tackle this problem, Yi et al. \([43]\) retrained several popular VSR methods \([29, 30, 31, 33, 40, 58]\) under a unified condition. We present part of experimental results from \([43]\) in Tables 1 and 2. To make a fair comparison with these methods, we also implement the proposed network under the same condition as \([43]\). For other two methods \([42, 59]\), we only present the results reported in their publications. All comparisons are conducted with a scaling factor of 4X. The quantitative evaluation metrics includes peak-signal-to-noise-ratio (PSNR) and structural similarity (SSIM), which are computed on the Y channel of YCbCr colour space.

First, we present the results on Vid4 testing dataset, which is composed of four video sequences: Calendar, city, foliage and walk. The quantitative results are presented in Table 1. It can be seen that the proposed NLVSR network achieves the best performance. It outperforms the state-of-the-art method, i.e. PFNL, by 0.18 dB in PSNR, and 0.0055 in SSIM on average. Specifically, the proposed NLVSR surpass PFNL by 0.18 dB/0.0066, 0.19 dB/0.0065, 0.10 dB/0.0053 and 0.22 dB/0.0037 on Calendar, city, foliage and walk, respectively. Although EDVR provides a higher PSNR values on walk, but the SSIM of NLVSR is still the highest. Several qualitative results of these methods on calendar sequence are presented in Figure 7. It can be observed that DUF_52L, PFNL, FRVSR and
NLVSR significantly outperform other networks in the zoom-in region. The characters from their SR results could be easily identified. Among these four methods, NLVSR recovers finer details with fewest distortion especially in character ‘A’ and ‘R’. Although Vid4 dataset is popular in VSR evaluation, it only includes four video sequences with a low resolution (smaller than 720 × 576). Thus, we also conduct a comparison on the Udm10 dataset, whose spatial size is 1272 × 720. Quantitative results are presented in Table 2. It can be seen that NLVSR achieves the best performance on both PSNR and SSIM for all video sequences. The average gains compared with other methods are 0.33 dB on PSNR and 0.0019 on SSIM at least. We also provide some super-resolved frames in Figure 8 to evaluate the visual quality of these VSR methods. As shown in Figure 8(a), the number ‘536’ can be hardly identified from the super-resolved frames of VESPCN, MCResNet, RVSR-LTD, DRVSR, and FRVSR. PFNL and DUF yield a better performance but suffer from some distortions, e.g. the number 6 becomes incomplete and is attached to the neighbouring number 3. Compared with these methods, NLVSR retains more details and present a more accurate result. From the visual results of the sequence photography, we find that many VSR methods fail in recovering some texture like dense parallel lines. For example, as shown in Figure 8(b), the reconstructed frames from all networks except NLVSR have a wrong direction in the texture of sweater. This mistake happens in all frames from VESPCN, RVSR-LTD, MCResNet, DRVSR and FRVSR. For DUF, only one recovered frame has the correct texture direction (31 frames in total from the sequence photography). PFNL is much better than above methods but still fails in the reconstruction of ten frames. However, NLVSR recovers all frames successfully, which means that it has a stronger ability than other approaches in handling this kind of texture.

In addition to the quality evaluation, we further show the tradeoff between accuracy, model size and test time of each network in the Figure 9. It can be observed that DRVSR, MCResNet, RVSR-LTD and DRVSR have a relatively small model size and fast inference speed, but their performance on PSNR is not ideal. FRVSR and DUF have a similar recovery accuracy and both require more parameters than other approaches. NLVSR achieves the best accuracy and has a similar model size with...
PFNL. However, the inference speed of NLVSR is relatively slow as compared to other methods. The reason is that the TSNL and APF module are both implemented in a frame-by-frame manner in our code. It will perform better in a parallelized implementation.

4.4 Effectiveness of TSNL

In order to evaluate the effectiveness of TSNL module, we design four networks with different alignment strategies: (1) the SISR method without any alignment, which only take the
LR reference frame as input, (2) STMC from [29], which is an explicit ME&MC method, (3) NLRB from [43], which is an implicit ME&MC method, and (4) the proposed TSNL block. The deformable convolution from [41, 42] is intended to be involved, but it is too complex (e.g., 8 GPUs used in [42] during the training) and it seems that there is no efficient implementation in TensorFlow which could run in 11GB GPU memory. To make a fair comparison, we use the same framework after alignment, which includes one convolution layer with $5 \times 5$ kernel, 20 residual blocks without batch normalization, and the same sub-pixel up-scaling module from [16]. The number of input frames is set as 7. The learning rate is set as $1 \times 10^{-4}$ during the whole training process. We name these four networks as: SISR, STMC-VSR, NLRB-VSR and TSNL-VSR. Note that the loss function of SISR, NLVSR-VSR and TSNL-VSR is the same as Equation (12), whereas STMC-VSR involves an extra limitation term for its motion estimation sub-network (refer to [29] for details).

As shown in Figure 10(a), three alignment methods all achieve a significant improvement compared with the SISR method. STMC-VSR converges early, but NLRB-VSR and TSNL-VSR exceed it at last. We can observe a performance gap about 0.15 dB between NLRB-VSR and TSNL-VSR, which implies the superiority of TSNL in alignment.

As discussed in Section 3, TSNL module not only exploits the temporal correlation across frames but also the self-similarity of the reference frame. To evaluate the contribution of this self-similarity to the SR performance, we present the training processes of NLVSR with it and without it in Figure 10(b). The learning rate is set as $5 \times 10^{-4}$. It can be seen that the PSNR gain of the self-similarity in TSNL module is about 0.07 dB on average after $1 \times 10^5$ steps. Actually, we do not expect too much improvement from the self-similarity of reference frame since there are other 6 aligned neighbouring frames and the original reference frame contributing to the reconstruction at the same time. However, as shown in the Figure 10(b), the participation of self-similarity speeds up the convergence of the network, which is beneficial to shortening the training time.

4.5 Effectiveness of APF

In this sub-section, we evaluate the effectiveness of progressive fusion strategy and temporal attention mechanism in APF module. First, we design a baseline model with a direct fusion strategy, which includes 30 residual blocks without batch normalization...
normalization. The second and third model replace the 30 residual blocks with 20 APFBs shown in Figure 5, and there is no temporal attention mechanism in the second model. Note that the parameter number of 30 residual blocks is almost the same as that of 20 APFBs. The TSNL module is applied in all models before fusion to make a fair comparison.

Table 3 presents the PSNR values of these three models on the validation dataset. It shows that the PSNR gain from the progressive fusion strategy is 0.58 dB, which indicates that the progressive fusion is much better than direct fusion in extracting temporal information from aligned frames. Comparing model 2 and model 3 in Table 3, we can observe an improvement of 0.15 dB from the proposed temporal attention mechanism. This implies that the attention mechanism is helpful for APF module in balancing the contribution of pixels with different quality.

4.6 Influence of input frames

NLVSR can accept any number of input frames in the form of \(2N + 1\). In above experiments, we set the frame number as 7 (i.e. \(N = 3\)) in order to be consistent with recent studies. Here, we further explore the influence of different frame numbers on the network performance. We trained 3 networks with 3, 5, and 7 frames as input, which are named as NLVSR-3f, NLVSR-5f, and NLVSR-7f, respectively. As shown in Figure 10(c), the more input frames are involved, the higher performance are achieved. This is consistent with the common sense that more input frames means more temporal information beneficial to the reconstruction. However, we find that the performance gap between NLVSR-3f and NLVSR-5f is much larger than that between NLVSR-5f and NLVSR-7f. The reason is that the frame with a long distance from the reference frame has a relatively low temporal correlation with it, and thus typically makes small contributions to the reconstruction. Besides, the inference time is linearly related to the frame number in the TSNL and APF module because of their multi-channel structures. Therefore, a moderate frame number like 5 or 7 would be a reasonable choice.

5 CONCLUSION

In this paper, we propose a novel network for VSR, termed NLVSR. The proposed network performs a specialized non-local operation on each frame to capture temporal and spatial long-range dependencies across frames, and align each of them to the reference frame. Then, the aligned information from each frame is fused in an attention-based progressive framework, which could balance the contribution of each aligned frame according to their quality. Extensive experiments demonstrate that the proposed network could recover accurate and temporally consistent SR frames, and achieve the state-of-the-art performance on public benchmark datasets.

ACKNOWLEDGMENTS

This work was partially supported by National Natural Science Foundation of China (No. 61872423), Industry Prospective Primary Research & Development Plan of Jiangsu Province (No.BE2017111), the Scientific Research Foundation of the Higher Education Institutions of Jiangsu Province (No. 19KJA180006), and Postgraduate Research & Practice Innovation Program of Jiangsu Province (No. KYCX19_0891), China.

ORCID

Chao Zhou https://orcid.org/0000-0001-6117-1805

REFERENCES

1. Nasrollahi, K., Moeslund, T.B.: Super-resolution: A comprehensive survey. Mach. Vis. Appl. 25(6), 1423–1468 2014
2. Yue, L., et al.: Image super-resolution: The techniques, applications, and future. Signal Process.128, 389–408 2016
3. Wang, N., et al.: A comprehensive survey to face hallucination. Int. J. Comput. Vis. 106(1), 9–30 2014
4. Jiang, J., et al.: SRLSP: A face image super-resolution algorithm using smooth regression with local structure prior. IEEE Trans. Multimedia 19(1), 27–40 2017.
5. Jiang, J., et al.: Facial image hallucination through coupled-layer neighbor embedding. IEEE Trans. Circuits Syst. V. Technol. 26(9), 1674–1684 2016
6. Jiang, K., et al.: A progressively enhanced network for video satellite imagery super-resolution. IEEE Signal Process. Lett. 25(11), 1630–1634 2018
7. Luo, Y., et al.: Video satellite imagery super resolution via convolutional neural networks. IEEE Geosci. Remote Sens. Lett. 14(12), 2398–2402 2017
8. Dhason, H., et al.: Super-resolution mapping of hyperspectral satellite images using hybrid genetic algorithm. IET Image Process. 14(7), 1281–1290 2019
9. Schmidhuber, J.: Deep learning in neural networks: An overview. Neural Networks 61, 85–117 2015
10. LeCun, Y., Bengio, Y., Hinton G.: Deep learning. Nature 521(7553), 436–444 2015
11. Dong, C., et al.: Image super-resolution using deep convolutional networks. IEEE Trans. Pattern Anal. Mach. Intell. 38(2), 295–307 2015
12. Kim, J., Lee, J. K., Lee, K. M.: Accurate image super-resolution using very deep convolutional networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, pp. 1646–1654 (2016)
13. Kim, J., Lee, J. K., Lee, K. M.: Deeply-recursive convolutional network for image super-resolution. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, pp. 1637–1645 (2016)
14. Tai, Y., Yang, J., Liu, X.: Image super-resolution via deep recursive residual network. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI, pp. 2790–2798 (2017)
15. Gao, H., et al.: Image super-resolution based on conditional generative adversarial network. IET Image Process. 14(3), 3006–3013 2020
16. Shi W.: Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, pp. 1874–1883 (2016)
17. Tong, T., et al.: Image super-resolution using dense skip connections. In: Proceedings of IEEE International Conference on Computer Vision (ICCV). Venice, pp. 4809–4817 (2017)
18. Zhang, Y., et al.: Residual dense network for image super-resolution. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Salt Lake City, UT, pp. 2742–2481 (2018)
19. Zhang, Y., et al.: Image super-resolution using very deep residual channel attention networks. In: Proceedings of European Convention on Computer Vision. Munich, Germany, pp. 294–310 (2018)
20. Zhang, L., Wu, X.: An edge-guided image interpolation algorithm via directional filtering and data fusion. IEEE Trans. Image Process. 15(8), 2226–2238 (2006)
21. Liu, X., et al.: Image interpolation via regularized local linear regression. IEEE Trans. Image Process. 20(12), 3455–3469 (2011)
22. Liu, X., et al.: Image interpolation via graph-based Bayesian label propagation. IEEE Trans. Image Process. 23(3), 1084–1106 (2014)
23. Freeman, W. T., Jones, T. R., Pasztor, E. C.: Example-based super-resolution. IEEE Comput. Graph. Appl. 22(2), 56–65 (2002)
24. He, K., et al.: Deep residual learning for image recognition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Las Vegas, NV, pp. 770–778 (2016)
25. Huang, G., et al.: Densely connected convolutional networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI, pp. 2261–2269 (2017)
26. Sanghyun, W., et al.: CBAM: Convolutional block attention module. In: Proceedings of European Conference on Computer Vision. Munich, Germany, pp. 3–19 (2018)
27. Liao, R., et al.: Video super-resolution via deep draft-ensemble learning. In: Proceedings of IEEE International Conference on Computer Vision (ICCV). Santiago, Chile, pp. 531–539 (2015)
28. Kappeler, A., et al.: Video super-resolution with convolutional neural networks. IEEE Trans. Comput. Imaging 2(2), 109–122 (2016)
29. Caballero, J., et al.: Real-time video super-resolution with spatiotemporal networks and motion compensation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI, pp. 2848–2857 (2017)
30. Tao, X., et al.: Detail-revealing deep video super-resolution. In: Proceedings of IEEE International Conference on Computer Vision (ICCV). Venice, pp. 4482–4490 (2017)
31. Liu, D., et al.: Robust video super-resolution with learned temporal dynamics. In: Proceedings of IEEE International Conference on Computer Vision (ICCV). Venice, pp. 2526–2534 (2017)
32. Xue, T., et al.: Video enhancement with task-oriented flow. Int. J. Comput. Vision 127(8), 1106–1125 (2019)
33. Sajjadi, M. S. M., Vernulapalli, R., Brown, M.: Frame-recurrent video super-resolution. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Salt Lake City, UT, pp. 6626–6634 (2018)
34. Wang, Z., et al.: Multi-memory convolutional neural network for video super-resolution. IEEE Trans. Image Process. 28(5), 2530–2544 (2019)
35. Wang, L., et al.: Deep video super-resolution using HR optical flow estimation. IEEE Trans. Image Process. 29, 4323–4336 (2020)
36. Li, F., Bai, H., Zhao, Y.: Learning a deep dual attention network for video super-resolution. IEEE Trans. Image Process. 29, 4474–4488 (2020)
37. Drudea, M., Nedevschi, S.: Total variation regularization of localglobal optical flow. In: Proceedings of IEEE Conference on Intelligent Transportation Systems (ITSC). Washington DC, pp. 318–323 (2011)
38. Brox, T., et al.: High accuracy optical flow estimation based on a theory for warping. In: Proceedings of European Conference on Computer Vision. Prague, Czech Republic, pp. 25–36 (2004)
39. Xu, L., Jia, J., Matsushita, Y.: Motion detail preserving optical flow estimation. IEEE Trans. Pattern Anal. Mach. Intell. 34(9), 1744–1757 (2012)
40. Jo, Y., et al.: Deep video super-resolution network using dynamic upsampling filters without explicit motion compensation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Salt Lake City, UT, pp. 3224–3232 (2018)
41. Tian, Y., Zhang, Y., Fu, Y.: TDAN: Temporally deformable alignment network for video super-resolution. arXiv preprint. arXiv:1812.02898 (2018)
42. Wang, X., et al.: EDVR: Video restoration with enhanced deformable convolutional networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA, pp. 1954–1963 (2019)
43. Yi, P., Wang, Z., Jiang, K.: Progressive fusion video super-resolution network via exploiting non-local spatio-temporal correlations. In: Proceedings of IEEE International Conference on Computer Vision (ICCV). Seoul, Korea, pp. 3106–3115 (2019)
44. G. Niu, Z. Ma: Local non-linear alignment for non-linear dimensionality reduction. IET Comput. Vision 11(5), 331–341 (2017)
45. H. Wang, et al.: Deformable non-local network for video super-resolution. IEEE Access 7, 177734–177744 (2019)
46. Sanghyun, W., et al.: Non-local neural networks. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Salt Lake City, UT, pp. 7794–7803 (2018)
47. Hans, M., Shakhnarovich, G., Ukit, N.: Recurrent back-projection network for video super-resolution. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA, pp. 3897–3906 (2019)
48. Li, K., et al.: Survey of single image super-resolution reconstruction. IET Image Process. 14(11), 2273–2290 (2020)
49. Shi, X., et al.: Convolutional LSTM network: A machine learning approach for precipitation nowcasting. In: Proceedings of Advances in Neural Information Processing System. Montreal, pp. 802–810 (2015)
50. Buades, A., Coll, B., Morel, J.: A non-local algorithm for image denoising. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). San Diego, CA, pp. 60–65 (2005)
51. Barnes, C., et al.: Patchmatch: A randomized correspondence algorithm for structural image editing. ACM Trans. Graphics 28(3), 24–24 (2009)
52. Glasner, D., Bagon, S., Irani, M.: Super-resolution from a single image. In: IEEE 12th International Conference on Computer Vision. Kyoto, pp. 394–399 (2009)
53. He, K., et al.: Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification. In: Proceedings of IEEE International Conference on Computer Vision. Santiago, Chile, pp. 1026–1034 (2015)
54. Lai, W. S., et al.: Deep Laplacian pyramid networks for fast and accurate super-resolution. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, HI, pp. 5835–5843 (2017)
55. Kingma, D. P., Ba, J.: Adam: A method for stochastic optimization. In: Proceedings of the International Conference on Learning Representations (ICLR). San Diego, CA, pp. 1–15 (2015)
56. Yi, P., et al.: Multi-temporal ultra dense memory network for video super-resolution. IEEE Trans. Circuits Syst. Video Technol. 30(8), 2503–2516 (2020)
57. Lim, B., et al.: Enhanced deep residual networks for single image super-resolution. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshops. Las Vegas, NV, pp. 1–9 (2016)
58. Li, D., Wang, Z.: Video super-resolution via motion compensation and deep residual learning. IEEE Trans. Comput. Imaging 3(4), 749–762 (2017)
59. Haris, M., Shakhnarovich, G., Ukit, N.: Recurrent back-projection network for video super-resolution. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Long Beach, CA, pp. 3897–3906 (2016)

How to cite this article: Zhou C, Chen C, Ding F, Zhang D. Video super-resolution with non-local alignment network. IET Image Process. 2021;15:1655–1667.
https://doi.org/10.1049/ipr2.12134