Online Research @ Cardiff

This is an Open Access document downloaded from ORCA, Cardiff University's institutional repository: https://orca.cardiff.ac.uk/id/eprint/127488/

This is the author's version of a work that was submitted to / accepted for publication.

Citation for final published version:


Publishers page: http://dx.doi.org/10.1109/ACCESS.2019.2958030
<http://dx.doi.org/10.1109/ACCESS.2019.2958030>

Please note:
Changes made as a result of publishing processes such as copy-editing, formatting and page numbers may not be reflected in this version. For the definitive version of this publication, please refer to the published source. You are advised to consult the publisher’s version if you wish to cite this paper.

This version is being made available in accordance with publisher policies. See http://orca.cf.ac.uk/policies.html for usage policies. Copyright and moral rights for publications made available in ORCA are retained by the copyright holders.
Deformable Non-local Network for Video Super-Resolution

HUA WANG1, DEWEI SU1, CHUANGCHUANG LIU1, LONGCUN JIN1, (Member, IEEE), XIANFANG SUN2 AND XINYI PENG1

1School of Software Engineering, South China University of Technology, Guangzhou 510006, China
2School of Computer Science and Informatics, Cardiff University, Cardiff CF24 3AA, U.K.

Corresponding author: Longcun Jin (lcjin@scut.edu.cn)

This work was supported in part by the Pearl River Technology Nova Project under Grant 201710010020, the Fundamental Research Funds for the Central Universities under Grant 2019MS086 and in part by the National Science Foundation of China under Grant 61300135.

ABSTRACT The video super-resolution (VSR) task aims to restore a high-resolution (HR) video frame by using its corresponding low-resolution (LR) frame and multiple neighboring frames. At present, many deep learning-based VSR methods rely on optical flow to perform frame alignment. The final recovery results will be greatly affected by the accuracy of optical flow. However, optical flow estimation cannot be completely accurate, and there are always some errors. In this paper, we propose a novel deformable non-local network (DNLN) which is a non-optical-flow-based method. Specifically, we apply the deformable convolution and improve its ability of adaptive alignment at the feature level. Furthermore, we utilize a non-local structure to capture the global correlation between the reference frame and the aligned neighboring frames, and simultaneously enhance desired fine details in the aligned frames. To reconstruct the final high-quality HR video frames, we use residual in residual dense blocks to take full advantage of the hierarchical features. Experimental results on benchmark datasets demonstrate that the proposed DNLN can achieve state-of-the-art performance on VSR task. The code is available at https://github.com/wh1h/DNLN.

INDEX TERMS Convolutional neural networks, deep learning, deformable convolution, non-local operation, video super-resolution.

I. INTRODUCTION

The target of super-resolution (SR) is to generate a corresponding high-resolution (HR) image or video from its low-resolution (LR) version. As an extension of single image super-resolution (SISR), video super-resolution (VSR) provides a solution to restore the correct content from the degraded video, so that the reconstructed video frames will contain more details with higher clarity. Such kind of technology with important practical significance can be widely used in many fields such as video surveillance [1], ultra-high definition television [2] and so on.

Different from SISR which only considers one single low-resolution image as input at a time, VSR devotes to effectively making use of intrinsic temporal information among multiple low-resolution video frames. Although vanilla SISR approaches can be directly applied to video frames by treating them as single images, abundant detail information available from neighboring frames will be wasted. Such practice is hard to reconstruct promising video frames, so they are not well adapted to VSR task.

To overcome the limitation of the SISR, existing VSR methods [3]–[7] usually take a LR reference frame and its multiple neighboring frames as inputs to reconstruct a corresponding HR reference frame. Due to the motion of the camera or objects, the neighboring frames should be spatially aligned first for utilizing the temporal information. To this end, most traditional VSR methods [8]–[11] generally calculate the optical flow and estimate the sub-pixel motion between LR frames to achieve the alignment operation. However, fast and reliable flow estimation still remains a challenging problem. The brightness constancy assumption, which most motion estimation algorithms rely on, may be invalid due to the existence of motion blur and occlusion. Incorrect motion compensation will introduce artifacts in aligned neighboring frames and affect the quality of final reconstructed video frames. Hence, explicit flow estimation and motion compensation methods could be sub-optimal for VSR task.

In this paper, we propose a novel deformable non-local network (DNLN), which is non-optical-flow-based, to per-
form both implicit motion estimation and video super-resolution. Our network mainly consists of four modules: feature extraction module, alignment module, non-local attention module and SR reconstruction module. Inspired by TDAN [12], we apply the deformable convolution [13] in our alignment module and enhance its ability of adaptively warping frames. Specifically, we introduce a hierarchical feature fusion block (HFFB) [14] to effectively handle the videos with large and complex motions. Through the stacks of deformable convolutions, we align the neighboring frame to the reference frame at the feature level and gradually improve the alignment accuracy. Then in the non-local attention module, we exploit a non-local structure to capture the global correlation between the reference feature and each aligned neighboring feature, which assesses the importance of different regions in neighboring feature. Such operation is expected to highlight the features complementary to the reference frame and exclude regions with improper alignment. The features with attention guidance are fused and then fed into the final SR reconstruction module. Here, we use residual in residual dense blocks (RRDB) [15] to generate the SR reference frame. RRDBs help to make full use of the information from different hierarchical levels and retain more details of the input LR frame.

In summary, the main contributions of this paper can be concluded as follows:

- We propose a novel deformable non-local network (DNLN) to accomplish high quality video super-resolution. Our method achieves the most advanced VSR performance on several benchmark datasets.
- We design an alignment module based on deformable convolution, which can realize the feature level alignment in a coarse to fine manner without explicitly motion compensation.
- We propose a non-local attention module to select significant features from neighboring frames which are conducive to the recovery of the reference frame.

The rest of the paper is organized as follows: Section 2 introduces the related work. Section 3 elaborates the structure of the proposed network. Section 4 shows our experimental results on benchmark datasets, including visual comparisons with other methods. The effectiveness of each components in our network is analyzed in Section 5. Finally, Section 6 draws conclusions.

II. RELATED WORK
A. SINGLE IMAGE SUPER-RESOLUTION
Dong et al. [16] first proposed SRCNN for single image super-resolution to learn the nonlinear mapping between LR and HR images in an end-to-end manner, which achieves better performance than previous work. Kim et al. [17] further improved SRCNN by stacking more convolution layers and using residual learning to increase network depth. Tai et al. introduced recursive blocks in DRNN [18], which employs parameters sharing strategy to make the training stable. All of these methods first upscale the LR input to the desired size and the reconstruction process is based on the upscaled products. Such pre-processing step inevitably results in loss of details and additional computation cost. To avoid these problems, extracting features from the original LR input and upscaling spatial resolution at the end of the network become the main direction of SR network. Dong et al. [19] directly took the original LR image as input and brought in the transpose convolution layer (also known as the deconvolution layer) for upsampling features to high resolution outcomes. Shi et al. [20] proposed an effective sub-pixel convolution layer for amplifying the final LR feature map to SR output and accelerating the network.

Afterwards, Timofte et al. [21] provided a new large dataset (DIV2K) in the NTIRE 2017 challenge that consists of 1000 2K resolution images. This dataset enables researchers to train deeper and wider networks which leads to various development of SR methods. The most advanced SISR networks, such as EDSR [22], DBPN [23], RDN [24] and RCAN [25], have far better training performance on this dataset than previous networks.

B. VIDEO SUPER-RESOLUTION
Liao et al. [26] proposed DECN and made use of two classical optical flow methods: TV-L1 and MDP flow to generate SR drafts with different parameters, and then produced the final result through a deep network. Kappler et al. [27] proposed VSRNet, which uses a hand-designed optical flow algorithm to perform motion compensation on the input LR frame, and takes the warped frame as the CNN input to predict the HR video frame. Caballero et al. [11] introduced the first end-to-end VSR network: ESPCN, which studies early fusion, slow fusion, and 3D convolution to learn temporal relationships. They applied a multi-scale spatial transformer to warp the LR frame and eventually generated a HR frame through another deep network. Tao et al. [9] proposed a sub-pixel motion compensation layer for frame alignment and used a convolutional LSTM architecture in following SR reconstruction network. Recently, Haris et al. [28] proposed RBPN which learns from the idea of back-projection to iteratively extract temporal features between frames. They treated frames independently rather than concatenated them together.

Most previous VSR methods exploit optical flow to estimate motion between frames and perform motion compensation to integrate effective features. While various approaches [29]–[33] are proposed to calculate the optical flow, it is still intractable to obtain precise flow estimation in the case of occlusion and large movement. Xue et al. [34] proposed task-oriented TOFlow with learnable task-oriented motion prompts. It achieved better VSR results than fixed flow algorithm, which reveals that standard optical flow is not the best motion representation for video recovery. To circumvent this problem, DUF [35] uses an adaptive upsampling with dynamic filters instead of the explicit estimation process. TDAN [12] uses deformable convolutions to adaptively
align the video frame at the feature level without computing optical flow. These kind of methods transcend the flow-based approaches through implicit motion compensation.

**C. DEFORMABLE CONVOLUTION**

To enhance the CNNs’ capability of modeling geometric transformations, Dai et al. [36] proposed deformable convolutions. It adds additional offsets to the regular grid sampling locations in the standard convolution and enables arbitrary deformation of the sampling grid. To further enhance the modeling capability, they proposed modulated deformable convolutions [13] which can further learn modulation scalar for sampling kernels. The modulation scalar lies in the range [0, 1], which can adjust the weight for each sampling location. The deformable convolution is effective for high-level vision tasks such as object detection and semantic segmentation. TDAN [12] is the first to utilize deformable convolutions in the VSR task. It is an end-to-end network which adaptively aligns the input frames at the feature level without explicit motion estimation. EDVR [37] further exploits the deformable convolutions with a pyramid and cascading structure. It shows superior performance to previous optical-flow-based VSR networks.

**D. NON-LOCAL BLOCK**

Inspired by the classic non-local method in computer vision, Wang et al. [38] proposed a building block for video classification by virtue of non-local operations. For image data, long-range dependencies are commonly modeled via large receptive fields formed by deep stacks of convolutional layers. While the non-local operations capture long-range dependencies directly by computing interactions between any two positions, regardless of their positional distance. It computes the response at a position as a weighted sum of all positions in the input feature maps. The set of positions can be in space, time, or spacetime, so the non-local operations are applicable for image or video problems.

**III. DEFORMABLE NON-LOCAL NETWORKS**

**A. NETWORK ARCHITECTURE**

Given a sequence of 2N+1 consecutive low-resolution frames \( \{ I_{t-N}^{LR}, \ldots, I_t^{LR}, I_t^{HR}, I_{t+1}^{LR}, \ldots, I_{t+N}^{LR} \} \), where \( I_t^{LR} \) is the reference frame and the others are the neighboring frames, our goal is to recover the corresponding high quality video frame through the reference frame and its 2N neighboring frames. Therefore, our network takes \( I_{t-N}^{LR}, I_t^{LR}, I_{t+1}^{LR}, \ldots, I_{t+N}^{LR} \) as inputs, and finally reconstructs \( I_t^{HR} \). The overall network structure is shown in Fig.1, which can be divided into four parts, including feature extraction module, alignment module, non-local attention module and the final SR reconstruction module.

For all the input LR frames, we first extract their features via a shared feature extraction module. It consists of one convolutional layer and several residual blocks. The feature extraction can be represented as:

\[
F_T = H_{fea}(I_t^{LR}),
\]

where the output \( F_T \) denotes the extracted LR feature maps. Then each LR neighboring feature \( F_i \) will enter the alignment module along with the LR reference feature \( F_t \). Our alignment module which consists of stacked deformable convolutions is responsible for performing adaptive feature level alignment:

\[
F_i^a = H_{align}(F_i, F_t), i \in [t-N, t+N]\ and\ i \neq t,
\]

where \( F_i^a \) denotes the neighboring feature after alignment. Subsequently, each aligned neighboring feature and the reference feature are fed into a non-local attention module. By calculating the global correlation between them, connections of pixels are established and informative regions in \( F_i^a \) will be further enhanced. The output \( F_i^a' \) of the non-local attention module can be expressed as:

\[
F_i^a' = H_{nl}(F_i^a, F_t).
\]

The last part is the SR reconstruction module, here we use the residual in residual dense blocks (RRDB). We concatenate 2N+1 features and fuse them through a 3 x 3 convolution layer, then the fused feature maps are fed into RRDBs for further reconstruction. Besides, we use a skip connection to propagate LR reference feature to the end of the network and do an element-wise addition with the outcome of RRDBs. Finally, a high quality HR reference frame is recovered from the output feature. The reconstruction module is defined as...
The proposed alignment module and the detailed illustration of deformable convolution operation.

FIGURE 2. The proposed alignment module and the detailed illustration of deformable convolution operation.

follows:

\[ F_{\text{fusion}} = \text{Conv} \left( \left[ F_{t-N}^a, \ldots, F_t^a, F_{t+1}^a, \ldots, F_{t+N}^a \right] \right) , \]  
\[ t_t^{SR} = H_{\text{rec}} (H_{\text{RRDBs}} (F_{\text{fusion}}) + F_t) , \]  
where \( \cdot \) denotes concatenation of the features. \( H_{\text{rec}} \) contains an upscaling layer and a reconstruction layer.

B. ALIGNMENT MODULE

In order to make use of temporal information from consecutive frames, traditional VSR methods are based on optical flow to perform frame alignment. However, explicit motion compensation method could be sub-optimal for video super-resolution task. We use modulated deformable convolutions [13] in the alignment module to get rid of such limitation.

For each location \( p \) on the output feature map \( Y \), a normal convolution process can be expressed as:

\[ Y(p) = \sum_{k=1}^{K} \omega_k \cdot X (p + p_k) , \]  
where \( p_k \) represents the sampling grid with \( K \) sampling locations and \( \omega_k \) denotes the weights for each location. For example, \( K = 9 \) and \( p_k \in \{-1, -1\}, \{0, 0\}, \ldots, \{1, 1\} \) defines a \( 3 \times 3 \) convolutional kernel. In the modulated deformable convolution, predicted offsets and modulation scalar are added to the sampling grid making deformable kernels spatially-variant. Here, we utilize the deformable convolution for temporal alignment. Let \( F_{i,b-1} \) and \( F_{i,b} \) denote the input and output of the deformable convolution in our module, respectively. The operation of modulated deformable convolution is as follows:

\[ F_{i,b}(p) = \sum_{k=1}^{K} \omega_k \cdot F_{i,b-1} (p + p_k + \Delta p_{i,k}) \cdot \Delta m_{i,k} , \]  
where \( \Delta p_{i,k} \) and \( \Delta m_{i,k} \) are the learnable offset and modulation scalar for the \( k \)-th location, respectively. The convolution will be operated on the irregular positions with dynamic weights to achieve adaptive sampling on input features. Since the offsets and modulation scalar are both learned, each input neighboring feature will be concatenated with the reference one to generate the corresponding deformable sampling parameters:

\[ \Delta P_i, \Delta M_i = f ([F_i, F_t]) , \]  
where \( \cdot \) denotes the concatenation operation. And \( \Delta P = \{ \Delta p_k \} \), \( \Delta M = \{ \Delta m_k \} \). As the \( \Delta p_k \) may be fractional,
we use the bilinear interpolation, which is the same as that proposed in [36].

The alignment module proposed in DNLN is composed of several deformable convolutions as shown in Fig.2. In each deformable convolution, a reference feature $F_i$ and a neighboring feature $F_t$ are concatenated as an input. Then they pass through a $3 \times 3$ convolution layer to reduce channels and a hierarchical feature fusion block (HFFB) [14] to increase the size of receptive field. The following $3 \times 3$ convolution layer is used to obtain the offset $\Delta P_i$ and modulation scalar $\Delta M_i$ for the deformable kernel. The structure of HFFB is depicted in Fig.3. It introduces a spatial pyramid of dilated convolutions to effectively enlarge receptive field with relatively low computational cost, which contributes to deal with complicated and large motions between frames. In HFFB, the feature maps obtained using kernels of different dilation rates are hierarchically added before being concatenated. With the same size of receptive field, the multiple dilated convolutions addition is dense more than just one dilated convolution. The use of HFFB is beneficial to acquire an effective receptive field, so we can more efficiently exploit the temporal dependency of pixels to generate the sampling parameters.

According to Eq.(7), the deformable kernel can adaptively select sampling positions on neighboring features, learn implicit motion compensation between two frames, and complete the alignment of features. With a cascade of deformable convolutions, we can gradually align the neighboring features and improve the alignment accuracy of sub-pixels. It is noticed that when passing through a deformable convolution layer, the reference feature always keeps unchanged, only to provide a reference for the alignment of neighboring features. Through such a coarse to fine process, the neighboring frames can be well warped at the feature level.

C. NON-LOCAL ATTENTION MODULE

Due to the factors such as occlusion, blurring and parallax problems, even after the alignment module, the neighboring frames still have some areas that are not well aligned or don’t contain the missing details needed for the reference frame. Therefore, it is essential to dynamically select valid inter-frame information before merging features. The proposed DNLN introduces a non-local attention module to achieve this goal. By capturing the global correlation between the aligned neighboring feature $F^a_i$ and the reference one $F_i$, the non-local module can effectively enhance desirable fine details in $F^a_i$ which can be complementary to the reference frame, and suppress the misaligned areas.

Let $x$ and $y$ denote the input feature $F^a_i$ and $F_i$ in Fig.4, respectively. The non-local operation in our module can be defined as:

$$z_p = x_p + W_z \sum_{n=1}^{N} \frac{f(x_p, y_n)}{C(y)} (W_g \cdot y_n), \quad (9)$$

where $z$ denotes the module output $F^a_i'$. Here $p$ is the index of an output position, and $n$ is the index that enumerates all positions on $y$. $W_g y_n$ computes the expression of input $y$ at position $n$. The function $f(x_p, y_n)$ calculates the relationship between $x_p$ and $y_n$. We use embedded Gaussian function to represent this pairwise relationship and it is normalized by a factor $C(y)$:

$$f(x_p, y_n) = \frac{\exp((W_a x_p, W_e y_n))}{\sum_n \exp((W_a x_p, W_e y_n))}, \quad (10)$$

$W_a x_p$, $W_e y_n$ are used to linearly embed the input and pairwise relationship is obtained from such a softmax computation. Then we calculate a value of position $p$ by using these relationships and the corresponding expression of all positions on $y$. The value is added to the input $x_p$ to get the final output $z_p$. Through non-local operation, the neighboring features can make full use of the correlation with the reference feature at the pixel level and enhance the desired missing details.

D. SR RECONSTRUCTION MODULE

The output of the non-local attention module $F^a_i'$ is aggregated with the reference feature $F_i$ through a feature fusion layer, and then fed into the following SR reconstruction module. The SR reconstruction module mainly consists of stacked residual in residual dense blocks (RRDB) and a
The structure of RRDB can be seen in Fig. 5. It combines multi-level residual network and dense connections. Benefiting from them, RRDBs can make full use of hierarchical features from input frames and obtain better restoration quality. More details about RRDB can be found in [15]. The global skip connection transfers the shallow features of the reference frame to the end of the network, making the reconstruction module focus on learning residual features from the neighboring frames. It can well keep spatial information of the input LR reference frame and ensure the input frame and the corresponding super-resolved one have more structural similarity. Finally, a high-resolution reference frame is produced by a sub-pixel upsampling layer and a reconstruction layer.

IV. EXPERIMENTS

A. TRAINING DATASETS AND DETAILS

Datasets To train high-performance VSR networks, a large video dataset is required. Xue [34] et al. collected videos from Vimeo and released a VSR dataset vimeo-90k after processing. The dataset contains 64612 training samples with various and complex real-world motions. Each sample contains seven consecutive frames with a fixed resolution of 448 × 256. We use the vimeo-90k dataset as our training dataset. To generate LR images, we downscale the HR images 4 × with MATLAB imresize function, which first blurs the input frames using cubic filters and then downsamples them using bicubic interpolation.

Training Details In our network, the convolutional layers have 64 filters and their kernel sizes are set to 3 × 3, if not specified otherwise. In the feature extraction module, we...
TABLE 1. Quantitative comparison of state-of-the-art SR algorithms on Vid4 for 4x. Red indicates the best and blue indicates the second best performance (PSNR/SSIM). In the evaluation, the first and last two frames are not included and we do not drop any border pixels except DUF. Eight pixels near image boundary are cropped for DUF.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Calendar</td>
<td>1.74</td>
<td>20.37/0.7512</td>
<td>22.31/0.7348</td>
<td>-</td>
<td>-</td>
<td>22.24/0.7290</td>
<td>-</td>
<td>23.95/0.8016</td>
<td>24.12/0.8741</td>
</tr>
<tr>
<td>City</td>
<td>1.63</td>
<td>25.17/0.6024</td>
<td>26.07/0.6938</td>
<td>-</td>
<td>-</td>
<td>26.75/0.7686</td>
<td>-</td>
<td>27.74/0.8051</td>
<td>27.90/0.8311</td>
</tr>
<tr>
<td>Foliage</td>
<td>1.48</td>
<td>23.47/0.5866</td>
<td>24.69/0.6632</td>
<td>-</td>
<td>-</td>
<td>25.24/0.7065</td>
<td>-</td>
<td>26.21/0.7078</td>
<td>26.28/0.7067</td>
</tr>
<tr>
<td>Walk</td>
<td>1.44</td>
<td>26.11/0.7977</td>
<td>28.64/0.8718</td>
<td>-</td>
<td>-</td>
<td>29.03/0.8777</td>
<td>-</td>
<td>30.63/0.9144</td>
<td>30.85/0.9129</td>
</tr>
<tr>
<td>Average</td>
<td>1.42</td>
<td>23.79/0.6347</td>
<td>25.43/0.7383</td>
<td>25.35/0.7557</td>
<td>25.86/0.7625</td>
<td>26.69/0.8229</td>
<td>27.36/0.8528</td>
<td>27.15/0.8204</td>
<td>27.29/0.8247</td>
</tr>
</tbody>
</table>

TABLE 2. Quantitative comparison of state-of-the-art SR algorithms on SPMCs-11 for 4x.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>car05_001</td>
<td>6.21</td>
<td>27.73/0.7825</td>
<td>29.86/0.8484</td>
<td>-</td>
<td>-</td>
<td>30.10/0.8626</td>
<td>30.79/0.8707</td>
<td>31.95/0.9021</td>
<td>31.96/0.9011</td>
</tr>
<tr>
<td>hdtech_003_001</td>
<td>0.70</td>
<td>19.42/0.4863</td>
<td>20.41/0.6096</td>
<td>-</td>
<td>-</td>
<td>20.86/0.6523</td>
<td>22.05/0.7438</td>
<td>21.91/0.7257</td>
<td>22.15/0.7366</td>
</tr>
<tr>
<td>hitachi_nee3_001</td>
<td>3.01</td>
<td>19.61/0.5938</td>
<td>23.71/0.8369</td>
<td>-</td>
<td>-</td>
<td>22.88/0.8044</td>
<td>25.77/0.8992</td>
<td>26.30/0.9049</td>
<td>26.60/0.9080</td>
</tr>
<tr>
<td>hko04_001</td>
<td>0.49</td>
<td>28.54/0.8003</td>
<td>31.68/0.8631</td>
<td>-</td>
<td>-</td>
<td>30.89/0.8654</td>
<td>32.98/0.9898</td>
<td>33.38/0.9016</td>
<td>33.46/0.9041</td>
</tr>
<tr>
<td>HKVTG_004</td>
<td>0.11</td>
<td>27.46/0.6831</td>
<td>28.81/0.7649</td>
<td>-</td>
<td>-</td>
<td>28.49/0.7487</td>
<td>29.16/0.7860</td>
<td>29.51/0.7979</td>
<td>29.53/0.7976</td>
</tr>
<tr>
<td>jvc_009_001</td>
<td>1.24</td>
<td>25.40/0.7585</td>
<td>28.31/0.8171</td>
<td>-</td>
<td>-</td>
<td>27.85/0.8542</td>
<td>29.18/0.8961</td>
<td>30.06/0.9105</td>
<td>30.65/0.9205</td>
</tr>
<tr>
<td>NYVTG_006</td>
<td>0.10</td>
<td>24.85/0.8014</td>
<td>31.01/0.8859</td>
<td>-</td>
<td>-</td>
<td>30.12/0.8603</td>
<td>32.30/0.9090</td>
<td>33.22/0.9231</td>
<td>33.35/0.9254</td>
</tr>
<tr>
<td>PRVTG_012</td>
<td>0.12</td>
<td>25.63/0.7136</td>
<td>26.56/0.7806</td>
<td>-</td>
<td>-</td>
<td>26.62/0.7788</td>
<td>27.39/0.8166</td>
<td>27.60/0.8242</td>
<td>27.68/0.8260</td>
</tr>
<tr>
<td>RMVTG_011</td>
<td>0.18</td>
<td>23.96/0.6573</td>
<td>26.02/0.7569</td>
<td>-</td>
<td>-</td>
<td>25.89/0.7500</td>
<td>27.56/0.8113</td>
<td>27.63/0.8170</td>
<td>27.75/0.8199</td>
</tr>
<tr>
<td>ven3_011</td>
<td>0.36</td>
<td>29.47/0.8979</td>
<td>34.59/0.9629</td>
<td>-</td>
<td>-</td>
<td>32.85/0.9336</td>
<td>34.63/0.9677</td>
<td>36.61/0.9735</td>
<td>36.33/0.9739</td>
</tr>
<tr>
<td>ven5_015</td>
<td>0.36</td>
<td>27.44/0.8483</td>
<td>31.04/0.9262</td>
<td>-</td>
<td>-</td>
<td>30.03/0.9118</td>
<td>31.88/0.9371</td>
<td>32.37/0.9409</td>
<td>33.04/0.9466</td>
</tr>
<tr>
<td>Average</td>
<td>1.17</td>
<td>25.73/0.7291</td>
<td>26.38/0.8279</td>
<td>27.87/0.8220</td>
<td>29.43/0.8664</td>
<td>30.05/0.8749</td>
<td>30.23/0.8782</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

B. COMPARISON WITH THE STATE-OF-THE-ART METHODS

We compare our DNLN with several state-of-the-art SISR and VSR methods: RCAN [25], VESPCN [11], TOFlow [34], FRVSR [10], DUF [35] and RBPN [28]. Note that most previous methods are trained with different datasets and we just compare with the results they provided. The SR results are evaluated with PSNR and SSIM [41] quantitatively on Y channel (i.e., luminance) of transformed YCbCr space. In the evaluation, the first and last two frames are not included and we do not crop any border pixels except DUF [35]. Eight pixels near image boundary are cropped for DUF due to its severe boundary effects.

We evaluated our models on three datasets: Vid4 [5], SPMCS [9], and Vimeo-90K-T [34] with average flow magnitude (pixel/frame) provided in [28]. Vid4 is a commonly used dataset which contains four video sequences: calendar, city, foliage and walk. However, we can observe that Vid4 has limited inter-frame motion and there exists artifacts on its ground-truth frames. SPMCS consists of higher quality video clips with various motions and diverse scenes. Vimeo-90K-T is a much larger dataset. It contains a wide range of flow magnitude between frames which can well judge the performance of the VSR methods.

Table 1 shows the quantitative results on Vid4. Our model outperforms the others in most cases. In the training process, we firstly perform data augmentation by doing horizontal or vertical flipping, 90° rotation and random cropping of the images. The batch size is set to 8. The network takes seven consecutive frames as inputs, and LR patches with the size of 50 × 50 are extracted for training. Our model is trained by Adam optimizer [39] with β1 = 0.9, β2 = 0.999, and ε = 10−8. The initial learning rate is 10−4 before 70 epochs and later decreases to half every 20 epochs. All experiments were conducted on two NVIDIA RTX 2080 GPUs using PyTorch 1.0 [40]. We train the network end-to-end by minimizing L1 loss between the predicted frame and the ground truth HR frame. We also employ the L2 loss to finetune the model, which could result in better performance.
FIGURE 7. Visual results on SPMCS for $4 \times$ scaling factor. Zoom in to see better visualization.

FIGURE 8. Visual results on Vimeo-90K-T for $4 \times$ scaling factor. Zoom in to see better visualization.
ground truth frame.

In comparison to Vid4, SPMCS contains more high-frequency information with higher resolution, which requires the superb recovery abilities of algorithms. Results on SPMCS are shown in Table 2. DNLN achieves the best results and outperforms other methods by a large margin on PSNR, which proves the superiority of our model. Visual comparisons are depicted in Fig.7. Due to the abundance of textures, most methods cannot fully recover the frames and obviously produce blurring artifacts. Although DUF and RBPN could reproduce part of the HR patterns, it is obvious that our DNLN is the unique approach to restore the abundant details and clean edges. Such visual comparisons demonstrate that our network can extract more sophisticated features from LR space with the proposed modules.

Table 3 presents the quantitative outcomes of Vimeo-90K-T. As suggested in [28], we classified the video clips into three groups (e.g. slow, medium and fast) according to the motion velocity. While the motion velocity increases, video frames with larger motion amplitude will contain more useful temporal information but also make the recovery more challenging. Our DNLN ensures optimal performance on all three groups, surpassing RBPN by 0.21 dB, 0.20 dB and 0.19 dB on PSNR, respectively. Since the flow magnitude of Vimeo-90K-T in fast group is higher than Vid4 and SPMCS, the content between video frames varies greatly, which reflects that DNLN could take full advantage of temporal information among multiple frames. The qualitative evaluations are shown in Fig.8. For the railing texture in third row, only our method restores the correct and clear pattern while others suffer from varying degrees of blurring. In some cases, even the SR frames recovered by different methods have the same sharp edges, our DNLN is more accurate and faithful to the ground truth. Such as the images in fourth row, the results restored by RBPN and DNLN are equally clear, while the former produces the stripes with wrong directions.

C. MODEL SIZE AND RUNNING TIME ANALYSES

Table 4 shows comparisons about model size and running time of the methods. The running time is test with input size $112 \times 64$. Our DNLN has the largest model size but also achieves the best performance. Here, we adopt a smaller model S-DNLN with only 3 deformable convolutions in alignment module and 14 RRDBs in reconstruction module. We can see that S-DNLN has a comparable number of parameters and running time with RBPN. Nevertheless, it can still get a better result which further validates the effectiveness of our proposed modules.

![Figure 9](image9.png)

**FIGURE 9.** Qualitative results of ablation on alignment module.

![Figure 10](image10.png)

**FIGURE 10.** Convergence analysis on the three modules of proposed network. The curves for each combination are based on the PSNR with scaling factor $4 \times$.

### Table 4. Number of parameters and time cost on Vimeo-90K-T for $4 \times$.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Parameter (M)</th>
<th>RCAN</th>
<th>TOFlow</th>
<th>DUF</th>
<th>RBPN</th>
<th>S-DNLN</th>
<th>DNLN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time (s)</td>
<td>0.051</td>
<td>0.126</td>
<td>0.175</td>
<td>0.086</td>
<td>0.095</td>
<td>0.119</td>
<td></td>
</tr>
<tr>
<td>PSNR</td>
<td>33.34</td>
<td>34.84</td>
<td>36.37</td>
<td>37.18</td>
<td>37.23</td>
<td>37.38</td>
<td></td>
</tr>
</tbody>
</table>

V. ABLATION STUDY

To further investigate the proposed method, we conducted ablation experiments by removing the main components of our network. The results are shown in Table 5. First, we remove the alignment module, thus the shallow features would be directly fed into the following network without warping. The PSNR of the results on Vimeo-90K-T is relatively low, which indicates that the alignment operation is crucial for utilizing the inter-frame information. Second, we remove the non-local attention module and the performance decreases a lot. Third, we replace the RRDBs by simply stacking common residual blocks and it also harms the performance. We visualize the convergence process of these combinations in Fig.10. The results demonstrate the effectiveness and benefits of our proposed three modules.

From the ablation experiments above, we can observe that the network performance would be significantly affected by the alignment preprocessing. So we further validated the impact of deformable convolutions on the reconstruction capability. As shown in Table 6, with only one deformable convolution, the PSNR value can improve greatly. It demonstrates the importance of alignment operations for making efficient use of the neighboring frames. As the number of...
deformable convolutions increases, the network gains a better performance. The visual comparisons are shown in Fig.9. From left to right, the network alleviates the blurring artifacts of the office building and recovers more accurate details. In addition, we replaced the HFFB used in deformable convolutions with a 3 × 3 convolution layer. The performance of network decreases by roughly 0.29 dB. It proves that by enlarging the receptive field, the deformable convolution can more effectively cope with complex and large motions.

In order to study the influence of inter-frame information on the recovery results, we leveraged different number of frames to train our network. From Table 7, we can observe that there is a significant improvement in DNLN when switching from 3 frames to 5 frames, and the performance of DNLN/5 is even better than RBPN which uses 7 frames. When further switching to 7 frames, we can still get a better result but the improvement becomes minor.

VI. CONCLUSION

In this paper, we propose a novel deformable non-local network (DNLN), which is a non-flow-based method for effective video super-resolution. To deal with complicated and large motion compensation, we introduce the deformable convolution with HFFB in our alignment module, which can well align the frames at the feature level. In addition, we adopt a non-local attention module to further extract complementary features from neighboring frames. By making full use of the temporal information, we finally restore a high quality video frame through a reconstruction module. Extensive experiments on benchmark datasets illustrate the effectiveness of our DNLN in video super-resolution.

REFERENCES


<table>
<thead>
<tr>
<th>Alignment module</th>
<th>Non-local module</th>
<th>SR module</th>
<th>PSNR</th>
<th>SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>✔</td>
<td>✔</td>
<td>36.82</td>
<td>0.9418</td>
</tr>
<tr>
<td></td>
<td>✔</td>
<td>✔</td>
<td>37.34</td>
<td>0.9471</td>
</tr>
<tr>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>37.25</td>
<td>0.9462</td>
</tr>
<tr>
<td>✔</td>
<td>✔</td>
<td>✔</td>
<td>37.38</td>
<td>0.9473</td>
</tr>
</tbody>
</table>

TABLE 6. Ablation study on alignment module.

<table>
<thead>
<tr>
<th>Model</th>
<th>PSNR</th>
<th>SSIM</th>
</tr>
</thead>
<tbody>
<tr>
<td>w/o deform</td>
<td>36.81</td>
<td>0.9417</td>
</tr>
<tr>
<td>1dconv</td>
<td>37.09</td>
<td>0.9446</td>
</tr>
<tr>
<td>2dconv</td>
<td>37.19</td>
<td>0.9455</td>
</tr>
<tr>
<td>3dconv</td>
<td>37.33</td>
<td>0.9469</td>
</tr>
<tr>
<td>4dconv</td>
<td>37.36</td>
<td>0.9471</td>
</tr>
<tr>
<td>5dconv</td>
<td>37.38</td>
<td>0.9473</td>
</tr>
<tr>
<td>5dconv, w/o HFFB</td>
<td>37.09</td>
<td>0.9447</td>
</tr>
</tbody>
</table>

TABLE 7. Experimental results with a different number of input frames.

<table>
<thead>
<tr>
<th>Input</th>
<th>3 frames</th>
<th>5 frames</th>
<th>7 frames</th>
</tr>
</thead>
<tbody>
<tr>
<td>PSNR / SSIM</td>
<td>37.06 / 0.9435</td>
<td>37.29 / 0.9463</td>
<td>37.38 / 0.9473</td>
</tr>
</tbody>
</table>


