Revisiting Temporal Alignment for Video Restoration

Kun Zhou1,2* Wenbo Li3* Liying Lu3 Xiaoguang Han1 Jiangbo Lu2†
1The Chinese University of Hong Kong (Shenzhen), 2SmartMore Corporation 3The Chinese University of Hong Kong

kunzhou@link.cuhk.edu.cn, {wenboli,llylu}@cse.cuhk.edu.hk
hanxiaoguang@cuhk.edu.cn, jiangbo.lu@gmail.com

1. Residual Block

As illustrated in Table 1, our residual block is comprised of two convolutions, where the first convolution is followed by ReLU activation. In previous works [1, 2, 11], the channel numbers typically keep identical within the residual block. In contrast, we set the channel dimension of hidden representations ($M_1$) no more than 64 to reduce the parameters of our network and speed up the training and inference phases:

$$M_1 = \max(M/2, 64),$$

(1)

<table>
<thead>
<tr>
<th>Input</th>
<th>Layer1</th>
<th>Layer2</th>
<th>Output</th>
</tr>
</thead>
<tbody>
<tr>
<td>x</td>
<td>Conv($M_1$, 3, 1) + ReLU</td>
<td>Conv($M_1$, 3, 1) =&gt; y</td>
<td>x+y</td>
</tr>
</tbody>
</table>

Table 1. The structure of our residual block. $M$ refers to the channel number of input.

2. Hyper-parameters in ARW

We examine the impacts of the patch size and $\alpha$ in our ARW module. By default, we use a patch size of 3 to calculate the accuracy-based re-weighting and set the $\alpha$ to $-1.0$ in consistency-based re-weighting. Also, we explore different settings of the two hyper-parameters. The results are shown in the Table 2. It is observed that a larger patch size leads to a performance drop and different values of $\alpha$ have limited influence on the final quality.

<table>
<thead>
<tr>
<th>P. Size</th>
<th>PSNR (dB)</th>
<th>$\alpha$</th>
<th>PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>3 x 3</td>
<td>37.84</td>
<td>-0.5</td>
<td>37.83</td>
</tr>
<tr>
<td>5 x 5</td>
<td>37.80</td>
<td>-1.0</td>
<td>37.84</td>
</tr>
<tr>
<td>7 x 7</td>
<td>37.72</td>
<td>-2.0</td>
<td>37.81</td>
</tr>
</tbody>
</table>

Table 2. Ablation study of the hyper-parameters in ARW.

3. More RBs in Reconstruction Module

To better understand the effect of the residual block (RB) numbers, apart from the default setting with 40 RBs, we train three additional models with 10, 20 and 60 RBs. As shown in the Table 3, the final performance is positively correlated with the number of RBs.

<table>
<thead>
<tr>
<th>Num. of RBs</th>
<th>PSNR (dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>37.51</td>
</tr>
<tr>
<td>20</td>
<td>37.69</td>
</tr>
<tr>
<td>40</td>
<td>37.84</td>
</tr>
<tr>
<td>60</td>
<td>37.89</td>
</tr>
</tbody>
</table>

Table 3. The influence of different RB numbers.

4. More Results

4.1. Temporal Consistency

We compare the temporal consistency of our method with several state-of-art video SR approaches [2, 4, 6, 11]. The visual results are illustrated in Figure 1. It is observed that other methods fail to restore the consistent textures clearly. While our method empowered with iterative alignment and two adaptively reweighting strategies is able to generate realistic image contents that are closest to the ground-truth.

4.2. Comparison with State-of-the-art

In Table 4 and 5, we give the detailed comparison with several state-of-the-art video SR approaches [1, 1, 2, 11] on REDS4 [8] and Vid4 [7]. The PSNR and SSIM of each video sequence are reported. For most video clips of both two validation sets, our model consistently achieves the best performance. Moreover, we provide extensive qualitative comparison on UDM10 [13], Vid4 [7], Vimeo-90K-T [12], and REDS4 [8] for video SR (in Figure 2, 3), VDB-T [9] for video deblurring (in Figure 4) and Set8 [10], DAVIS [5] for video denoising (in Figure 5). All the qualitative results demonstrate that our method has the capacity to handle various challenging cases in these three video restoration tasks.

---

*Equal contribution
†Corresponding author
Table 4. Quantitative comparison on REDS4 [8] benchmark under ×4 setting for video super-resolution. Numbers in red and blue refer to the best and second-best results. All the results are evaluated in the RGB channel. ‘∗’ indicates the results are from [1].

<table>
<thead>
<tr>
<th></th>
<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 (Y)</td>
<td>20.39/0.5720</td>
<td>23.34/0.8110</td>
<td>30.76/0.8147</td>
<td>-</td>
<td>30.60/0.9141</td>
<td>31.00/0.9152</td>
<td>-</td>
<td>-</td>
<td>31.10/0.9163</td>
<td>31.30/0.9167</td>
</tr>
<tr>
<td>Average (Y)</td>
<td>23.78/0.6547</td>
<td>27.33/0.8318</td>
<td>30.76/0.8264</td>
<td>27.26/0.8215</td>
<td>27.39/0.8279</td>
<td>27.36/0.8258</td>
<td>27.90/0.8380</td>
<td>-</td>
<td>-</td>
<td>27.05/0.8355</td>
</tr>
</tbody>
</table>

Table 5. Quantitative comparison on Vid4 [7] under ×4 setting for video super-resolution. We report the PSNR (dB)/SSIM results on both the RGB and the Y channel. Numbers in red and blue refer to the best and second-best results.

4.3. Comparison with BasicVSR++

In Table 5 of our main text, the PSNR/SSIM values of BasicVSR++ on Vimeo-90K-T are obtained by pre-training on REDS. Though our model is only trained on Vimeo-90K without pre-training (as a typical setup), our method still performs better than it (37.79dB → 37.84dB). As for REDS4 and Vid4, BasicVSR++ aggregates the information from the full sequence (i.e., 100 frames for REDS4 and 34-49 frames for Vid4) for super-resolving a video frame. In contrast, we adopt the commonly used 5/7-frame settings, like other methods evaluated in Table 4 of our manuscript. In summary, BasicVSR++ actually used extra information and different test setups.

4.4. Video Results

We also provide three videos for visual inspection. “city.mp4”. This video illustrates the visual comparison between bicubic and our method on a Vid4 clip for video super-resolution. It can be observed that our method restores much clear image details (e.g., the finer structure of buildings).

“IMG0030.mp4”. This video demonstrates the visual results of our method on a testing sequence of VDB-T [9] for the video deblurring task. The blurry input and the generated frames are shown in it.

“motorbike.mp4”. This video shows the restoration results on a sequence of Set8 [10] for video denoising.

References

ponents in video super-resolution and beyond. In Proceedings of the IEEE conference on computer vision and pattern recognition, 2021. 1, 2


Figure 3. Qualitative comparison on Vimeo-90K-T [12] and REDS4 [8] for video SR.


Figure 4. Qualitative comparison on VDB-T [9] for video deblurring.
Figure 5. Qualitative comparison on Set8 [10], DAVIS [5] for video denoising. The values beneath images represent the PSNR (dB).