Learning Spatiotemporal Frequency-Transformer for Compressed Video Super-Resolution Supplementary Material

Zhongwei Qiu[1,2], Huan Yang[3], Jianlong Fu[3], and Dongmei Fu[1,2]

[1] University of Science and Technology Beijing
[2] Shunde Graduate School of University of Science and Technology Beijing
[3] Microsoft Research
qiuzhongwei@xs.ustb.edu.cn, huayan@microsoft.com, jianf@microsoft.com, fdm_ustb@ustb.edu.cn

In this supplementary material, we introduce the algorithm details in Section 1. More implemental details are shown in Section 2. The influence of compression is discussed in Section 3. More visualization results and failure cases are shown in Section 4. The limitations of FTVSR are discussed in Section 4.

1 Algorithm Details

The algorithm details of FTVSR with divided time-space-frequency attention Λ_{ST} are shown in Algorithm 1. In Algorithm 1, the upsampling network $\varphi(\cdot)$ and the flow network $\phi(\cdot)$ is same as TTVSR [4]. We follow COMISR [3], BasicVSR [1] and TTVSR [4] to use bidirectional propagation scheme, which includes forward and backward propagation. For clarity, only forward propagation is shown in Algorithm 1.

2 Implemental Details

In this section, we introduce more implemental details that can not be involved in the main paper since the limited pages. All the ablation results are based on the backbone of BasicVSR [1] and final model is based on the backbone of TTVSR [4]. For DCT, we transform image patches of size $B \times B, B = 8$ to pixel domain and align them at frequency dimension. Thus, the frequency number is 64. For frequency tokenization, the block size $K \times K$ is 8×8 for a trade-off of computational costs and the performances for different block sizes are shown in Table 1. For COMISR [3], we retrain it as the settings in [3] to generate visualization results.

3 Comparison of Compression

We follow the same setting as COMISR [3] that adopts ffmpeg to perform the video compression. The compression command is "ffmpeg -i LR.mp4 -vcodec libx264 -crf CRFvalue save.mp4", where CRF value can be 15, 25, and 35. The

Algorithm 1 FTVSR with divided time-space-frequency attention Λ_{ST}

Input: I_{LR}: $\{I^t_{LR}, t \in [1,T]\}$; T: the length of sequence; N: the block numbers of each frame; F: the frequency numbers. H_{init} initialization by zero. $U(\cdot)$: Bicubic upsampling. $\varphi(\cdot)$: upsampling network. $\phi(\cdot)$: flow estimation. DCT$(\cdot)$: Discrete Cosine Transform. rDCT$(\cdot)$: inverse Discrete Cosine Transform. $W(\cdot)$: flow warp. $\Lambda_S(\cdot)$: space-frequency attention. $\Lambda_T(\cdot)$: time-frequency attention. Γ: fusion layer.
Output: I_{SR}: $\{I^t_{SR}, t \in [1,T]\}$;

$H = \{H_{init}\}$;
for $t = 1$; $t <= T$; $t++$ **do**
  $O^t = \phi(I^t_{LR}, I^{t-1}_{LR})$;
  $\hat{H}^t = W(H^{t-1}, O^t)$;
  $\mathcal{Q} = \text{DCT}(U(I^t_{LR})) = \{\tau^q_{(t,i,f)}, i \in [1,N], f \in [1,F]\}$;
  $\mathcal{K} = \text{DCT}(\varphi(\mathsf{I}_{LR})) = \{\tau^k_{(t',i,f)}, t' \in [1,t-1], i \in [1,N], f \in [1,F]\}$;
  $\mathcal{V} = \text{DCT}(\varphi(\mathsf{I}_{LR})) = \{\tau^v_{(t',i,f)}, t' \in [1,t-1], i \in [1,N], f \in [1,F]\}$;
  $R^t = \Lambda_S(\tau^q_{(t,i,f)}, \tau^k_{(t,i,f)}, \tau^k_{(t,i,f)})$;
  $P^t = \Lambda_T(R^t, \hat{H}^t, \hat{H}^t)$;
  $D^t_{LR} = \text{DCT}(\varphi(I^t_{LR}))$;
  H add $\Gamma(D^t_{LR}, P^t)$;
  $I^t_{SR} = \text{rDCT}(\Gamma(P^t, D^t_{LR}) + D^t_{LR})$
end for

Table 1. The ablation study of block size for space-frequency attention Λ_S

Method	Block Size	CRF15	CRF25	CRF35
FTVSR (Λ_S)	4×4	29.62/0.840	27.20/0.761	24.12/0.646
	6×6	29.64/0.840	27.22/0.761	24.12/0.645
	8×8	**29.63/0.840**	**27.23/0.761**	**24.12/0.646**
	12×12	29.63/0.840	27.22/0.761	24.12/0.646
	16×16	29.61/0.839	27.20/0.760	24.10/0.644

differences of no compression and compression are shown in Figure 1. As shown in Figure 1, compared with no compression, compression brings more artifacts, which broke the texture structure in the image.

4 Visualization and Failure Cases

4.1 More visualization results

We compare FTVSR and SOTA methods (EDVR [5], MUCAN [2], BasicVSR [1], IconVSR [1] and COMISR [3]) on the compressed videos with different compression rates. The visualization results are shown in Figure 2.

4.2 Limitations and Failure Cases

We discuss the limitations of the proposed FTVSR in this subsection and show some failure cases in Figure 3.

Fig. 1. The visualization of compressed images with different compression rates

Small Parts As shown in the first row of Figure 3, some small parts in image, which its textures are relatively small, are not easy to recover since the limited input information.

Motion Parts As shown in the second row of Figure 3, the textures with complex motion patterns (e.g. rotation) are also not easy to recover. FTVSR can capture the contour texture of the rotating wheel, but fail on the detail texture of the wheel hub.

References

1. Chan, K.C., Wang, X., Yu, K., Dong, C., Loy, C.C.: BasicVSR: The search for essential components in video super-resolution and beyond. In: CVPR. pp. 4947–4956 (2021)
2. Li, W., Tao, X., Guo, T., Qi, L., Lu, J., Jia, J.: MuCAN: Multi-correspondence aggregation network for video super-resolution. In: ECCV. pp. 335–351. Springer (2020)
3. Li, Y., Jin, P., Yang, F., Liu, C., Yang, M.H., Milanfar, P.: COMISR: Compression-informed video super-resolution. In: ICCV (2021)
4. Liu, C., Yang, H., Fu, J., Qian, X.: Learning trajectory-aware transformer for video super-resolution. In: CVPR. pp. 5687–5696 (2022)
5. Wang, X., Chan, K.C., Yu, K., Dong, C., Change Loy, C.: EDVR: Video restoration with enhanced deformable convolutional networks. In: CVPRW (2019)

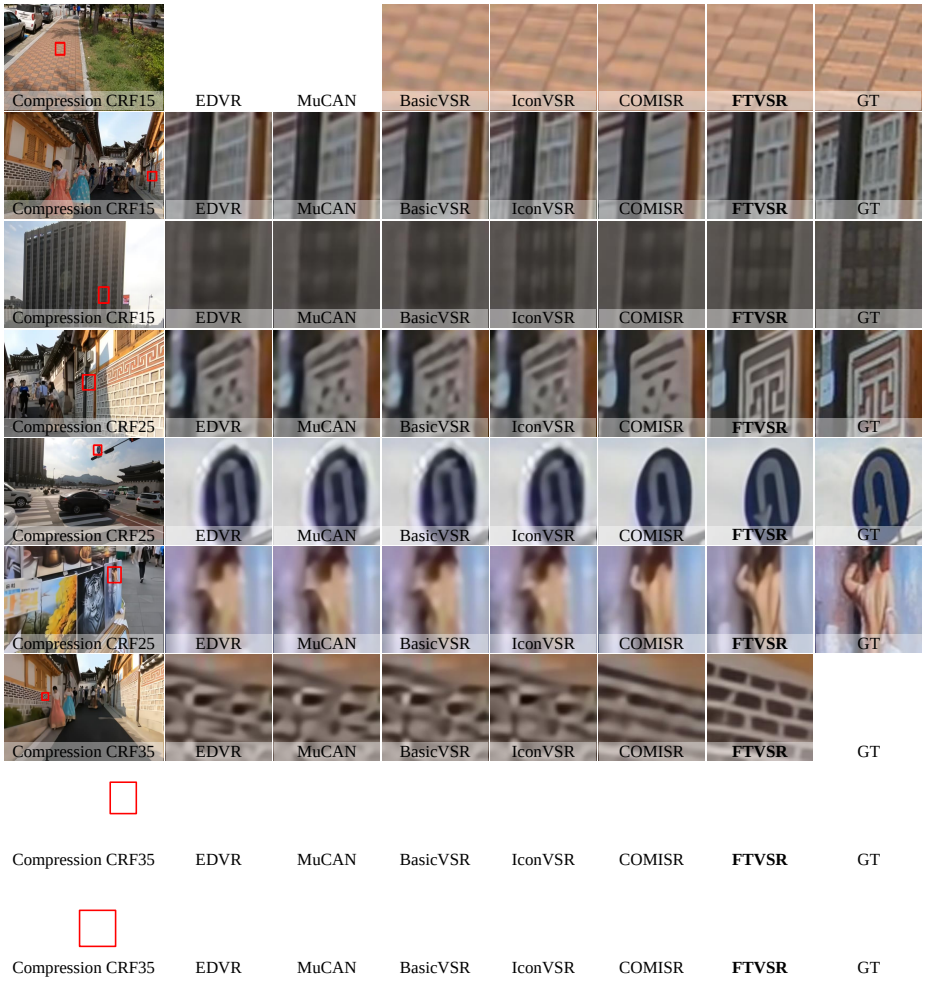

Fig. 2. The visualization of more results on compressed videos with different compression rates

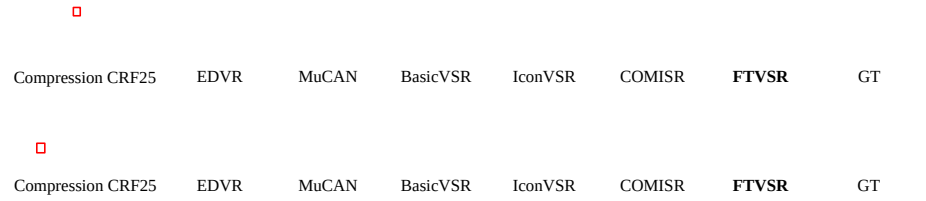

Fig. 3. The comparison of failure cases with other methods (EDVR [5], MUCAN [2], BasicVSR [1], IconVSR [1], COMISR [3]) on compressed videos with compression rate of CRF25