Investigating Tradeoffs in Real-World Video Super-Resolution – Supplementary Material –

Kelvin C.K. Chan Shangchen Zhou Xiangyu Xu Chen Change Loy ✉
S-Lab, Nanyang Technological University
{chan0899, s200094, xiangyu.xu, ccloy}@ntu.edu.sg

1. Architecture and Experimental Settings

Architecture. We use a simple architecture in this work for explorational purpose. First, a convolution is used to extract shallow features from the input image. A stack of 20 residual blocks are then used to extract deep features. A final convolutional layer is then used to produce the clean image. We adopt BasicVSR [1] as the VSR network. We reduce the number of residual blocks from 60 to 40 to maintain comparable complexity to the original BasicVSR.

Loss Function. For the output fidelity loss $\mathcal{L}_{pix}$ and image cleaining loss $\mathcal{L}_{clean}$, we use Charbonnier loss [3] since it better handles outliers and improves the performance over the conventional ℓ_2 loss [7]. In addition, we use perceptual loss [6] $\mathcal{L}_{per}$ and adversarial loss [4] $\mathcal{L}_{adv}$ to achieve better visual quality.

In the first stage, we pretrain the generator (*i.e.*, RealBasicVSR) with the fidelity loss and image cleaning loss:

$$\mathcal{L}_{1st} = \mathcal{L}_{pix} + \mathcal{L}_{clean}. \tag{1}$$

We then finetune the network with also perceptual loss and adversarial loss:

$$\mathcal{L}_{2nd} = \mathcal{L}_{pix} + \mathcal{L}_{clean} + \lambda_{per}\mathcal{L}_{per} + \lambda_{adv}\mathcal{L}_{adv}. \tag{2}$$

In our experiments, λ_{per}=1 and λ_{adv}=5×10^{-2}. Note that in the second stage, the weights of the cleaning module are kept fixed.

Training Degradations. Following Real-ESRGAN [11], we adopt the second-order order degradation model, and we apply random blur, resize, noise, and JPEG compression as image-based degradations. In addition, we incorporate video compression, which is a common technique to reduce video size. Unlike the aforementioned degradations, video compression implicitly considers the interdependencies between video frames, providing us with temporally and spatially varying degradations. The settings of image-based degradations follow Real-ESRGAN [11]. For the video compression, in each iteration, we randomly select one of the following codecs: "libx264", "h264", and "mpeg4". The bitrate is uniformly selected from the range $[10^4, 10^5]$. Video compression is added right after JPEG compression.

Qauntitative Metrics. Our quantitative metrics are computed on the Y-channel. To save computational cost, we compute the metrics on the *first*, *middle*, *last* frames of each seqeuence. The details is shown in Table 1.

Table 1. **Frames used in our quantitative comparison.** To save computational cost, we compute the metrics only on the frames specified below.

Video ID	Frame Numbers
030	000, 020, 040
031	000, 017, 033
032	000, 024, 048
033	000, 023, 046
others	000, 050, 099

Implementation. We implement our models with PyTorch and train the models using eight NVIDIA Tesla V100 GPUs. Code will be made publicly available at MMEditing [9] and https://github.com/ckkelvinchan/RealBasicVSR.

2. Discussion of Baselines

In this work, we compare our RealBasicVSR with seven state of the arts, including four image models: RealSR [5], DAN [8], Real-ESRGAN [11], BSRGAN [13] and three video models: BasicVSR++[1] [2], RealVSR [12], DBVSR [10]. They are representative methods in image and video super-resolution that achieve promising performance.

With specific designs in training, these methods demonstrate significant improvements when compared to non-blind methods. However, while these methods succeed in removing degradations in the input images, they are inferior in recovering details beyond the image itself or its local

[1] Trained with bicubic downsampling, as a reference.

neighbors, due to the fact that they do not exploit long-term information available in videos.

Despite being extensively discussed in non-blind VSR, the use of long-term information has not been explored in real-world VSR. In this work, we find that such long-term information, if used with designated designs, is also useful in real-world VSR. With the benefits of our findings and designs, RealBasicVSR is able to restore more details than the methods in comparison, as shown in Fig. 1 and Fig. 2.

3. Dynamic Refinement

In this section, we show additional examples demonstrating the effects of our dynamic refinement. As shown in Fig. 3, unpleasant artifacts remain in the outputs when applying cleaning once, and unnatually flat outputs due to over-cleaning are observed when our cleaning module is applied five times. In contrast, our refinement scheme automatically stops the refinement to avoid over-smoothing while cleaning excessive artifacts, leading to improved performance. More sophisticated decision processes are left as our future work.

4. Limitation of RealBasicVSR

While RealBasicVSR shows much better capability when compared to existing works, it does not work well when the degradations are too extreme or too different from the training degradations. This is a common problem in real-world restoration, and a more thorough understanding of the real-world degradations is needed. Further improvements on the generalizability is left as our future work.

References

[1] Kelvin C.K. Chan, Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. BasicVSR: The search for essential components in video super-resolution and beyond. In *CVPR*, 2021. 1

[2] Kelvin C.K. Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. BasicVSR++: Improving video super-resolution with enhanced propagation and alignment. *arXiv preprint arXiv:2104.13371*, 2021. 1

[3] Pierre Charbonnier, Laure Blanc-Feraud, Gilles Aubert, and Michel Barlaud. Two deterministic half-quadratic regularization algorithms for computed imaging. In *ICIP*, 1994. 1

[4] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014. 1

[5] Xiaozhong Ji, Yun Cao, Ying Tai, Chengjie Wang, Jilin Li, and Feiyue Huang. Real-world super-resolution via kernel estimation and noise injection. In *CVPRW*, 2020. 1

[6] Justin Johnson, Alexandre Alahi, and Fei-Fei Li. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, 2017. 1

[7] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *CVPR*, 2017. 1

[8] Zhengxiong Luo, Yan Huang, Shang Li, Liang Wang, and Tieniu Tan. Unfolding the alternating optimization for blind super resolution. In *NeurIPS*, 2020. 1

[9] Contributors MMEditing. MMEditing: OpenMMLab Image and Video Editing Toolbox, 3 2022. 1

[10] Jinshan Pan, Songsheng Cheng, Jiawei Zhang, and Jinhui Tang. Deep blind video super-resolution. In *ICCV*, 2021. 1

[11] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-ESRGAN: Training real-world blind super-resolution with pure synthetic data. In *ICCVW*, 2021. 1

[12] Xi Yang, Wangmeng Xiang, Hui Zeng, and Lei Zhang. Real-world video super-resolution: A benchmark dataset and a decomposition based learning scheme. In *ICCV*, 2021. 1

[13] Kai Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. In *ICCV*, 2021. 1

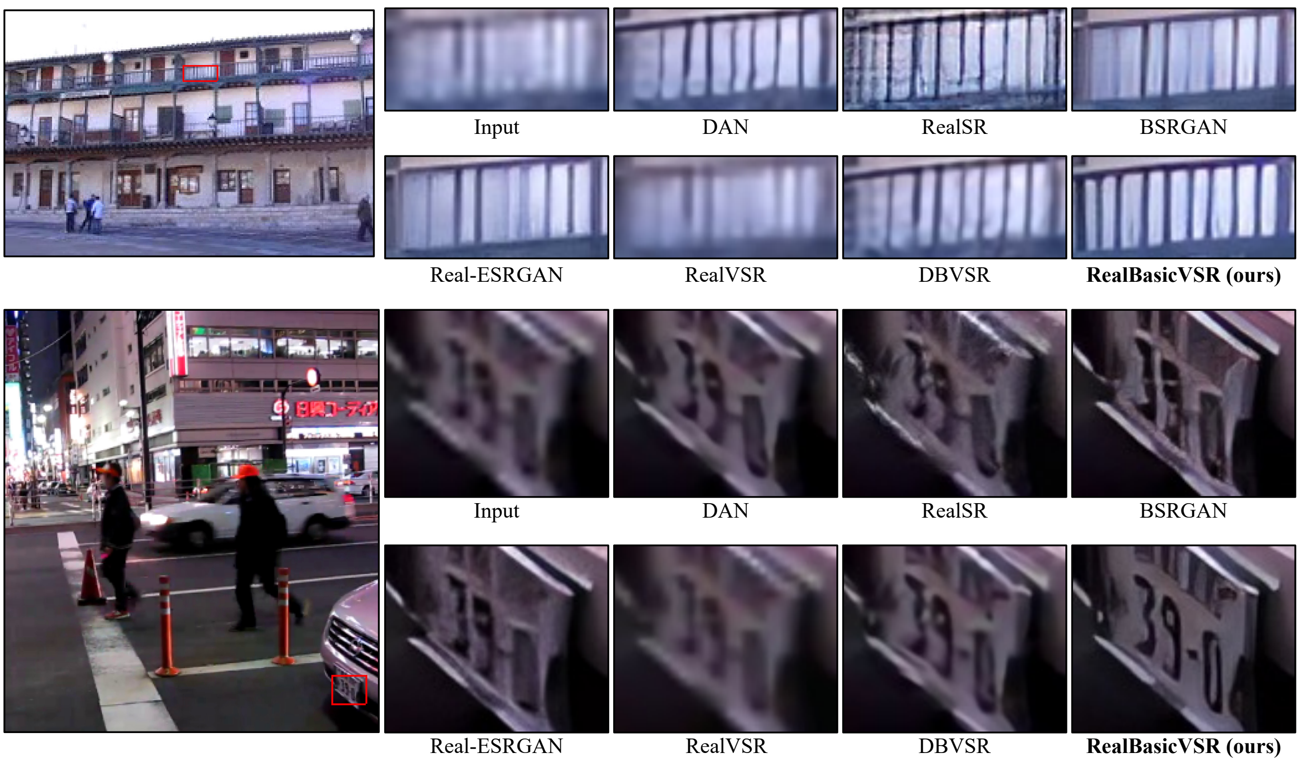

Figure 1. **Qualitative Comparison.** By employing the long-term information effectively, RealBasicVSR restores more details when compared to existing state of the arts. **(Zoom-in for best view)**

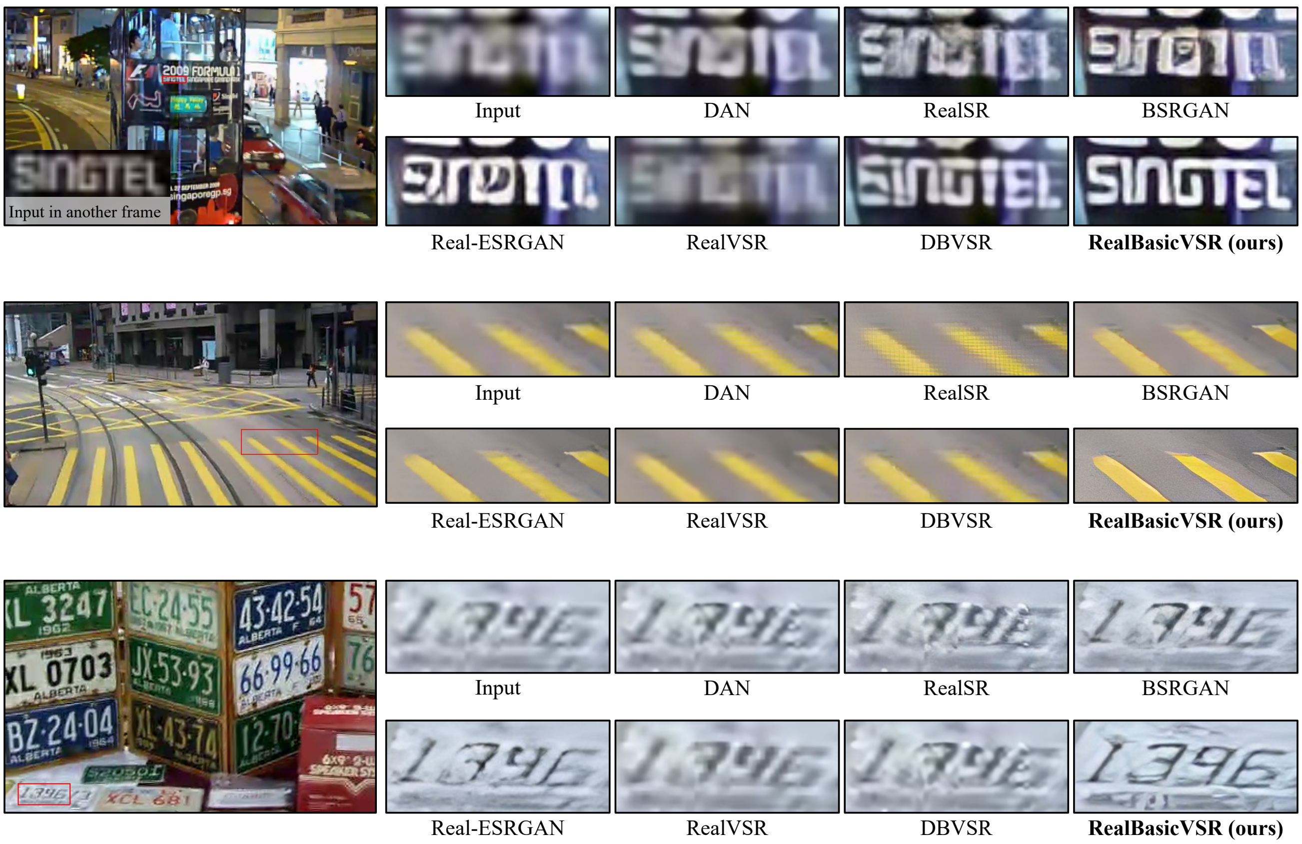

Figure 2. **Qualitative Comparison.** By employing the long-term information effectively, RealBasicVSR restores more details when compared to existing state of the arts. **(Zoom-in for best view)**

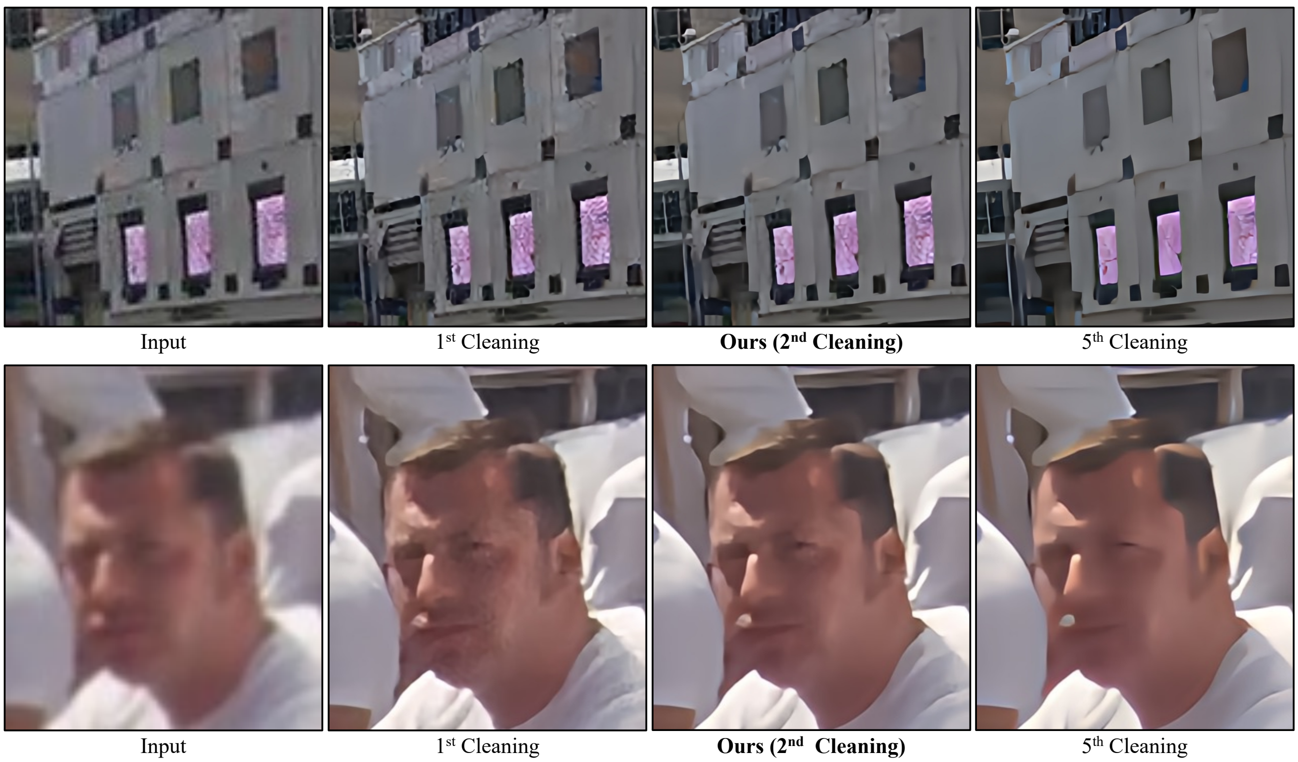

Figure 3. **Dynamic Refinement.** Our dynamic refinement scheme removes remaining noises and artifacts in the first cleaning while avoiding over-smoothing. **(Zoom-in for best view)**