Learning Data-Driven Vector-Quantized Degradation Model for Animation Video Super-Resolution Supplementary Material

Zixi Tuo*[1], Huan Yang†[2], Jianlong Fu[2], Yujie Dun[1], Xueming Qian[1,3]
[1]Xi'an Jiaotong University [2]Microsoft Research Asia
[3]Shaanxi Yulan Jiuzhou Intelligent Optoelectronic Technology Co., Ltd
zixit99@gmail.com, {huayan, jianf}@microsoft.com, {dunyj, qianxm}@mail.xjtu.edu.cn

In this supplementary material, Sec. 1 first illustrates the implementation details of VQD-SR, which includes the implementation of VQ degradation model in Sec. 1.1, the details of VSR model in Sec. 1.2, and the experiment details in Sec. 1.3. Then Sec. 2 introduces our RAL dataset with statistics and representative samples. Sec. 3 shows more comparison results.

1. Implementation Details

1.1. VQ Degradation Model

Network Architecture. The architecture of our multi-scale VQGAN for degradation modeling is described in Tab. 4. The inputs of (b) Middle Branch and (c) Bottom Branch are the intermediate outputs of (a) Top Branch, represented by x_m and x_b respectively. The corresponding output features $\hat{x}_m$ and $\hat{x}_b$ are further added back to the top branch when decoding. The vector-quantization is conducted pixel by pixel on the encoded outputs z_t, z_m and z_b in latent space with 256 channels VQ codebook sized 1024. The compression factor f, which denotes the patch size when projecting each entry in the VQ codebook from latent space to the original image space, is set to $\{8,4,2\}$ controlled by the number of downsample steps in these three branches. During training, the middle branch and the bottom branch only take effect in the second-stage. All the training procedures are performed on eight NVIDIA 32G V100 GPUs.

Degradation Pipeline. The whole degradation pipeline with multi-scale VQGAN to transfer the real-world degradation priors can be formulated as: $x = D^n(y) = (\text{FFmpeg} \circ \text{VQD} \circ \text{Down} \circ \text{Noise} \circ \text{Blur})(y)$. Where x denotes the degraded LR clips and y denotes the HR clips. For basic operators (blur, noise, and FFmpeg), we follow the settings and hyperparameters in AnimeSR [8]. For VQ degradation, we keep the nearest neighbor search on the top and the middle branch while adopting the stochastic top-k VQ strategy on the bottom branch.

Stochastic Top-k VQ strategy. Specifically, for each training clip in every iteration, we uniformly sample an integer

*This work was done while Zixi Tuo was a research intern at Microsoft Research Asia.
†Corresponding author.

Table 1. Ablation Study of the value K in stochastic Top-k VQ strategy.

	Top-1	Top-30	Top-50	Top-100
MANIQA↑	0.3770	0.3846	**0.3857**	0.3756

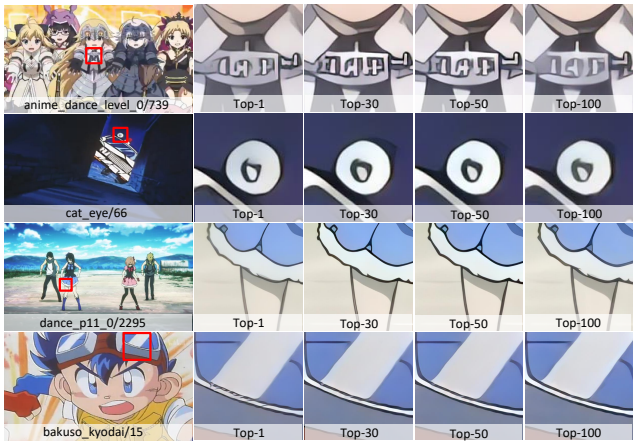

Figure 1. Ablation Study of the value K in stochastic Top-k VQ strategy. We choose K = 50 in our VQD-SR.

Table 2. Results of different animation VSR methods in NIQE. '$*$' denotes fine-tune on animation dataset AVC-Train [8].

	RealBasicVSR*	AnimeSR	VQD-SR
NIQE↓	8.5358	8.7088	**8.4737**

Table 3. Evaluations with PSNR/SSIM on ATD test2k.

	AnimeSR	VQD-SR
PSNR↑	35.49	**35.54**
SSIM↑	0.9701	**0.9702**

Table 4. Architecture of Multi-scale VQGAN. The proposed multi-scale VQGAN is composed of three parallel branches: (a) Top Branch, (b) Middle Branch, and (c) Bottom Branch. The downsample block is realized by a 3×3 convolution layer with stride 2. The upsample block is composed of a 3×3 convolution layer with stride 1 and a pixelshuffle layer. $C = 128$ is the number of base channel, $n_z = 256$ is the embedding dimension of VQ codebook.

(a) Top Branch

Encoder	Decoder
$x \in \mathbb{R}^{H \times W \times 3}$	$z_{q(t)} \in \mathbb{R}^{H/8 \times W/8 \times n_z}$
Conv2D $\rightarrow \mathbb{R}^{H \times W \times C}$	Conv2D $\rightarrow \mathbb{R}^{H/8 \times W/8 \times 4C}$
Downsample Block, $2 \times$ Residual Block$\rightarrow x_b \in \mathbb{R}^{H/2 \times W/2 \times C}$	Non-Local Block $\rightarrow \mathbb{R}^{H/8 \times W/8 \times 4C}$
Downsample Block, $2 \times$ Residual Block $\rightarrow x_m \in \mathbb{R}^{H/4 \times W/4 \times 2C}$	$2 \times$ Residual Block, Conv2D $\rightarrow \mathbb{R}^{H/8 \times W/8 \times 2C}$
Downsample Block, $2 \times$ Residual Block$\rightarrow \mathbb{R}^{H/8 \times W/8 \times 2C}$	(+ $\hat{x}_m$), $2 \times$ Residual Block, Upsample Block$\rightarrow \mathbb{R}^{H/4 \times W/4 \times 2C}$
Conv2D, $2 \times$ Residual Block$\rightarrow \mathbb{R}^{H/8 \times W/8 \times 4C}$	(+ $\hat{x}_b$), $2 \times$ Residual Block, Upsample Block$\rightarrow \mathbb{R}^{H/2 \times W/2 \times C}$
Non-Local Block$\rightarrow \mathbb{R}^{H/8 \times W/8 \times 4C}$	$2 \times$ Residual Block, Upsample Block$\rightarrow \mathbb{R}^{H \times W \times C}$
GroupNorm, Conv2D $\rightarrow z_t \in \mathbb{R}^{H/8 \times W/8 \times n_z}$	$2 \times$ Residual Block $\rightarrow \hat{x} \in \mathbb{R}^{H \times W \times 3}$

(b) Middle Branch

Encoder	Decoder
$x_m \in \mathbb{R}^{H/4 \times W/4 \times 2C}$	$z_{q(m)} \in \mathbb{R}^{H/4 \times W/4 \times n_z}$
Conv2D, $2 \times$ Residual Block$\rightarrow \mathbb{R}^{H/4 \times W/4 \times 2C}$	Conv2D$\rightarrow \mathbb{R}^{H/4 \times W/4 \times 4C}$
Conv2D, $2 \times$ Residual Block$\rightarrow \mathbb{R}^{H/4 \times W/4 \times 4C}$	$2 \times$ Residual Block, Conv2D $\rightarrow \mathbb{R}^{H/4 \times W/4 \times 2C}$
GroupNorm, Conv2D $\rightarrow z_m \in \mathbb{R}^{H/4 \times W/4 \times n_z}$	$2 \times$ Residual Block, Conv2D $\rightarrow \hat{x}_m \in \mathbb{R}^{H/4 \times W/4 \times 2C}$

(c) Bottom Branch

Encoder	Decoder
$x_b \in \mathbb{R}^{H/2 \times W/2 \times C}$	$z_{q(b)} \in \mathbb{R}^{H/2 \times W/2 \times n_z}$
Conv2D, $2 \times$ Residual Block$\rightarrow \mathbb{R}^{H/2 \times W/2 \times 2C}$	Conv2D$\rightarrow \mathbb{R}^{H/2 \times W/2 \times 4C}$
Conv2D, $2 \times$ Residual Block$\rightarrow \mathbb{R}^{H/2 \times W/2 \times 2C}$	$2 \times$ Residual Block, Conv2D $\rightarrow \mathbb{R}^{H/2 \times W/2 \times 2C}$
Conv2D, $2 \times$ Residual Block$\rightarrow \mathbb{R}^{H/2 \times W/2 \times 4C}$	$2 \times$ Residual Block, Conv2D $\rightarrow \mathbb{R}^{H/2 \times W/2 \times 2C}$
GroupNorm, Conv2D $\rightarrow z_b \in \mathbb{R}^{H/2 \times W/2 \times n_z}$	$2 \times$ Residual Block, Conv2D $\rightarrow \hat{x}_b \in \mathbb{R}^{H/2 \times W/2 \times C}$

k from [1, K] as the degradation level, and utilize the k_{th} nearest codebook entry to conduct the element-wise quantization on the encoded output in the bottom branch. A larger k denotes a more severe degradation, and K denotes the max degradation level in training. As shown in Tab. 1 and Fig. 1, we compare the results of animation VSR when $K = \{1, 30, 50, 100\}$ on VQD-SR (base). When trained with a small K (*e.g.*, $K = 1$, also the nearest neighbor search), the animation VSR model has limited generalization ability due to the rigid degradation level in training. However, when K is too large (*e.g.*, $K = 100$), the degradations are so severe that contaminate the original image structures and disturb the training of VSR model. Based on the results, we finally choose $K = 50$ for our VQ degradation model, which leads to better results with sharper lines in smooth shapes. Referring to Fig. 7-8, we also show some visual examples of LR video frames degraded by multi-scale VQGAN in multi-levels with different k.

1.2. Video Super-Resolution Model

We follow the VSR model in AnimeSR [8] because of its efficiency but remove the SR feedback in the recurrent block as shown in Fig. 2. Different from natural domain videos, the continuities between animation video frames are relatively poor which causes difficulties for explicit alignment modules and further impacts the final VSR results. Thus, the explicit alignment module is left out in the architecture of animation VSR model, where the misaligned recurrent features are directly adopted, which also greatly shrinks the computation costs. However, as is studied by Chan *et al*. [2] that although long-term information is beneficial for VSR, it may suffer from error accumulation during propagation. For VSR models without explicit alignment modules, this problem could be more severe. As shown in Fig. 3, we find that too many misaligned recurrent features make the animation VSR model [8] susceptible to error accumulation, and sometimes cause a collapse of GAN loss in training, leading to corrupt VSR models. Based on this observation, we remove the SR feedback in the recurrent block to ensure more stable training. We train the VSR model on eight NVIDIA 32G V100 GPUs.

1.3. Experiment Details

Evaluations with MANIQA. We test the super-resolution methods on AVC-RealLQ [8], which is a real-world animation video dataset containing 46 low-quality clips with 100 frames per clip. As the lack of ground truths for testing, we follow AnimeSR [8] and adopt the no-reference image quality assessment (NR-IQA) metric MANIQA [9] to evaluate the final SR results in the main paper. MANIQA is

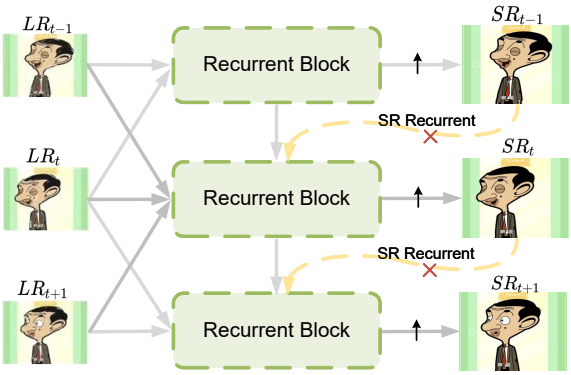

Figure 2. Architecture of VSR model. We follow the VSR model in AnimeSR [8] but remove the SR feedback in the recurrent block for stable training.

Figure 3. Too many recurrent features cause unstable training leading to corrupt animation VSR models.

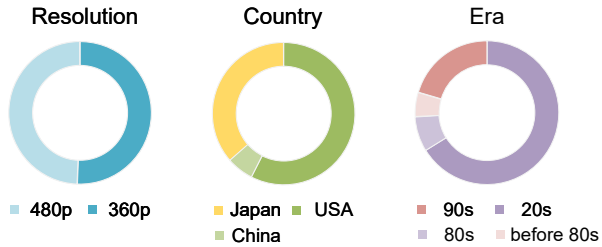

Figure 4. Statistics of RAL.

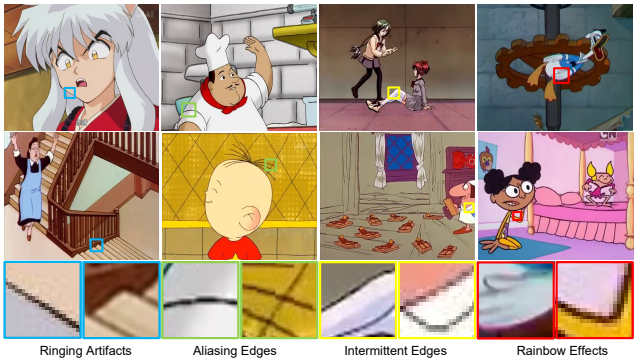

Figure 5. Samples of RAL and typical degradation phenomena in real-world LR animation videos.

designed for rating GAN-based distorted images, which is suitable for evaluating GAN-based image restoration algorithms. Following [9] and [8], we measure MANIQA every 10 frames in each video. For each testing frame, we randomly crop 224×224 sized images 20 times and calculate the average score. We run the evaluation process 3 times and report the mean metrics for all of the reported results.

Evaluations with NIQE. We show the evaluations of different animation VSR methods with NIQE [4] on AVC-RealLQ in Tab. 2. NIQE is a hand-crafted feature-based method that has less ability to measure the diverse distortions in the real world and especially some GAN-based distortions caused by image restoration algorithms. [8] also has mentioned that NIQE lacks consistent with the perceptual visual quality. However, our VQD-SR still outperforms the other two SOTA animation VSR methods in NIQE.

Evaluations on ATD test2k. ATD-12K [6] is a large-scale HR animation triplet dataset, which comprises 12,000 triplets. As only HR images are provided in ATD, we follow the conventional SR setting that reports PSNR/SSIM of AnimeSR and ours on $4\times$ bicubic (BI) downsampled images in Tab. 3. The results demonstrate the advantage of our method under the ideal BI setting. However, the setting of AVC-RealLQ in the main paper is a more challenging but practical scenario for real applications of animation VSR.

User Study on Visual Quality. We conduct A-B tests to further compare the visual quality of VQD-SR with other six SOTA methods (BasicVSR [1], PDM [3], Real-ESRGAN [7], BSRGAN [10], RealBasicVSR [2], AnimeSR [8]) successively. For each comparison with one of the six methods, there are 20 subjects involved in the tests on the SR results of AVC-RealLQ [8] with 46 animation video clips. Considering the subtle qualitative differences between frames, we uniformly sample 4 frames for testing in each clip. The user interface for the A-B test, as shown in Fig. 6, provides the users with two images in random order which include one VQD-SR frame and one frame from the other method. Users are asked to select one with higher visual quality. As the resolutions of SR results are too large (*e.g.*, 5760×4320) to fit the screen and display the details at the same time if only providing the complete images, we further show two side-by-side zoom-in windows controlled by mouse with adjustable positions and sizes. The final results are the percentages of votes which prefer VQD-SR to other methods.

2. Statistics and Samples of RAL

Our **R**eal **A**nimation **L**ow-quality (RAL) video dataset contains over 10K LR frames extracted from 441 real-world low-quality animation videos and contains rich real-world degradations in animation domain. The statistics of RAL is shown in Fig. 4. In Fig. 5, we also show some representative samples and typical real-world degradations in RAL.

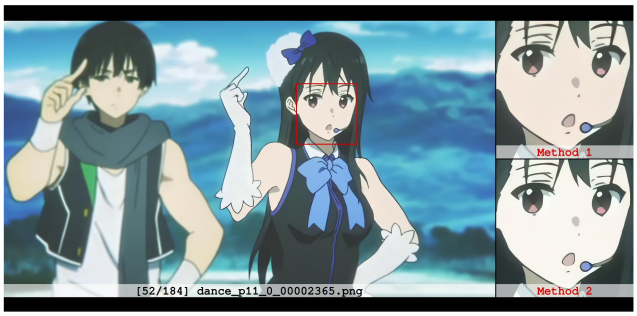

Figure 6. UI of the A-B test. The main window shows the complete image with the current testing progress and frame name beneath. Two sub-windows on the right display the zoom-in details of two methods where the mouse hovers, and the size of the viewport can also be adjusted by scrolling the mouse wheel. Users are asked to select the one with higher visual quality by pressing '1' or '2' on the keyboard.

3. More Qualitative Comparisons

In this section, we show more qualitative results to verify the effectiveness of our proposed methods.

Animation VSR Results. In Fig. 9 - 12, we compare our VQD-SR with six SOTA SR methods. The first crop shown in each case is the bicubic $4\times$ upscaled original input for reference, as the absence of ground truths in real scenarios. Our VQD-SR is capable to recover visually natural and sharper lines (Fig. 9, Fig.11) with fewer artifacts, restore clear details (Fig. 10), handle some intended scenarios (*e.g.*, the out-focus background blur) with fewer over-sharp artifacts (Fig. 12).

HR-SR Enhancement. In Fig. 14, we show the enhanced HR animation video frames with different SR models. Our HR-SR enhancement strategy could alleviate the compression artifacts and sharpen the edges without contaminating the original details in animation HR frames. As shown in Fig. 13, the proposed HR-SR strategy is valid to improve the results of animation VSR, regardless of the specific VSR model, with the help of more effective ground truths for training.

We further extend the HR-SR enhancement strategy from animation videos to natural videos (REDS [5]) in Fig. 15. Because of the complex textures and irregular illumination conditions, directly adopting HR-SR enhancement strategy to natural videos would cause amplified illumination artifacts (row 1), contaminated details (row 2), and over-sharp textures (row 3), leading to unappealing results.

References

[1] Kelvin CK 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*, pages 4947–4956, 2021. 3

[2] Kelvin CK Chan, Shangchen Zhou, Xiangyu Xu, and Chen Change Loy. Investigating tradeoffs in real-world video super-resolution. In *CVPR*, pages 5962–5971, 2022. 2, 3

[3] Zhengxiong Luo, Yan Huang, Shang Li, Liang Wang, and Tieniu Tan. Learning the degradation distribution for blind image super-resolution. In *CVPR*, pages 6063–6072, 2022. 3

[4] Anish Mittal, Rajiv Soundararajan, and Alan C Bovik. Making a "completely blind" image quality analyzer. *SPL*, 20(3):209–212, 2012. 3

[5] Seungjun Nah, Sungyong Baik, Seokil Hong, Gyeongsik Moon, Sanghyun Son, Radu Timofte, and Kyoung Mu Lee. NTIRE 2019 challenge on video deblurring and super-resolution: Dataset and study. In *CVPRW*, pages 0–0, 2019. 4

[6] Li Siyao, Shiyu Zhao, Weijiang Yu, Wenxiu Sun, Dimitris Metaxas, Chen Change Loy, and Ziwei Liu. Deep animation video interpolation in the wild. In *CVPR*, pages 6587–6595, 2021. 3

[7] Xintao Wang, Liangbin Xie, Chao Dong, and Ying Shan. Real-ESRGAN: Training real-world blind super-resolution with pure synthetic data. In *ICCV*, pages 1905–1914, 2021. 3

[8] Yanze Wu, Xintao Wang, Gen Li, and Ying Shan. AnimeSR: Learning real-world super-resolution models for animation videos. *arXiv preprint arXiv:2206.07038*, 2022. 1, 2, 3, 7, 8, 9, 10

[9] Sidi Yang, Tianhe Wu, Shuwei Shi, Shanshan Lao, Yuan Gong, Mingdeng Cao, Jiahao Wang, and Yujiu Yang. MANIQA: Multi-dimension attention network for no-reference image quality assessment. In *CVPR*, pages 1191–1200, 2022. 2, 3

[10] Kai Zhang, Jingyun Liang, Luc Van Gool, and Radu Timofte. Designing a practical degradation model for deep blind image super-resolution. In *ICCV*, pages 4791–4800, 2021. 3

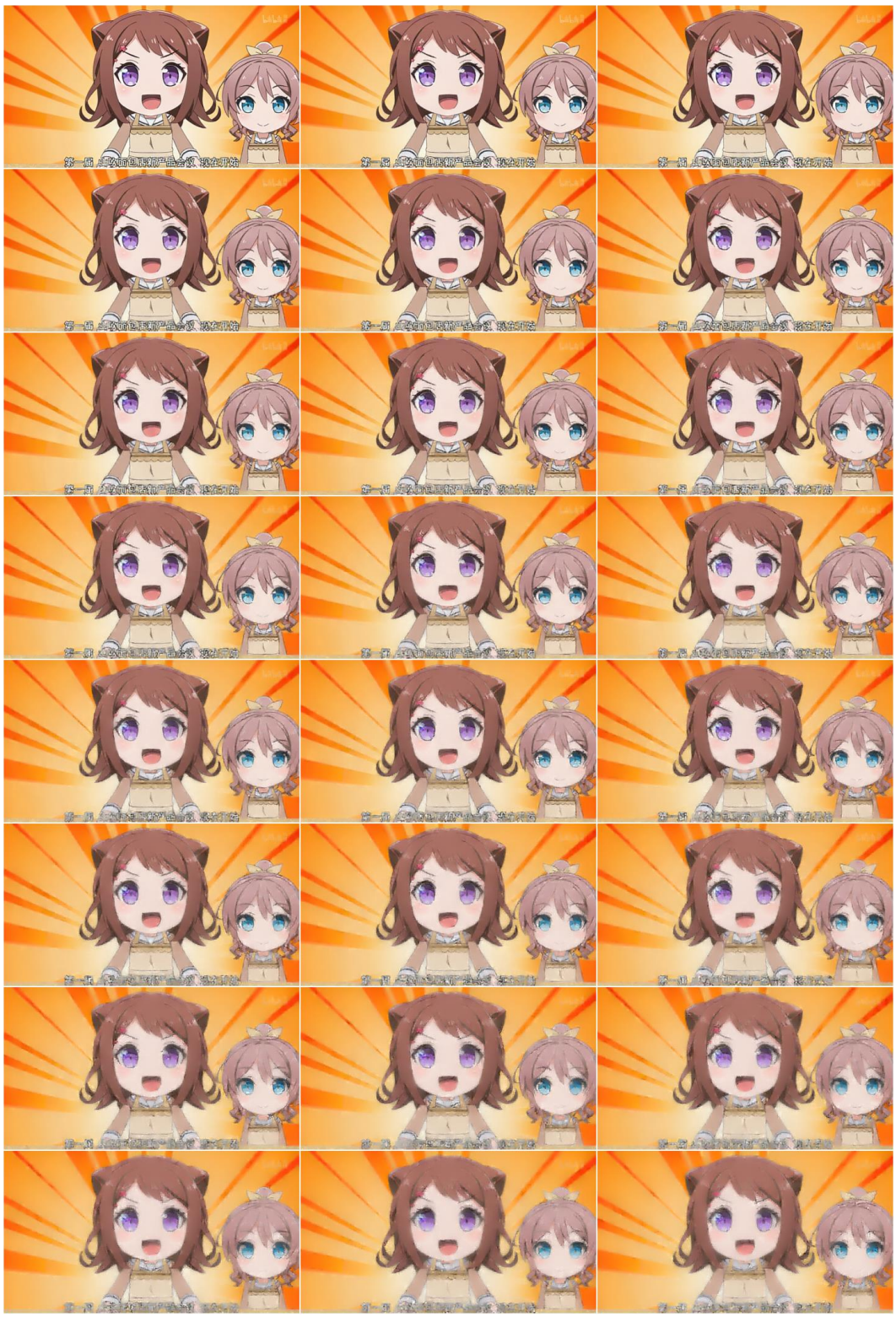

Figure 7. Examples of LR frames degraded by multi-scale VQGAN in multi-levels. From top to bottom, left to right, we show the degradation levels in ascending order. For the sake of clear comparisons, we show the results every 3 levels in the first 70 levels ($k = 1, 4, 7, ..., 70$) here. Input HR: AVC-Train/$bang\_dream\_p1\_0/00000049.png$

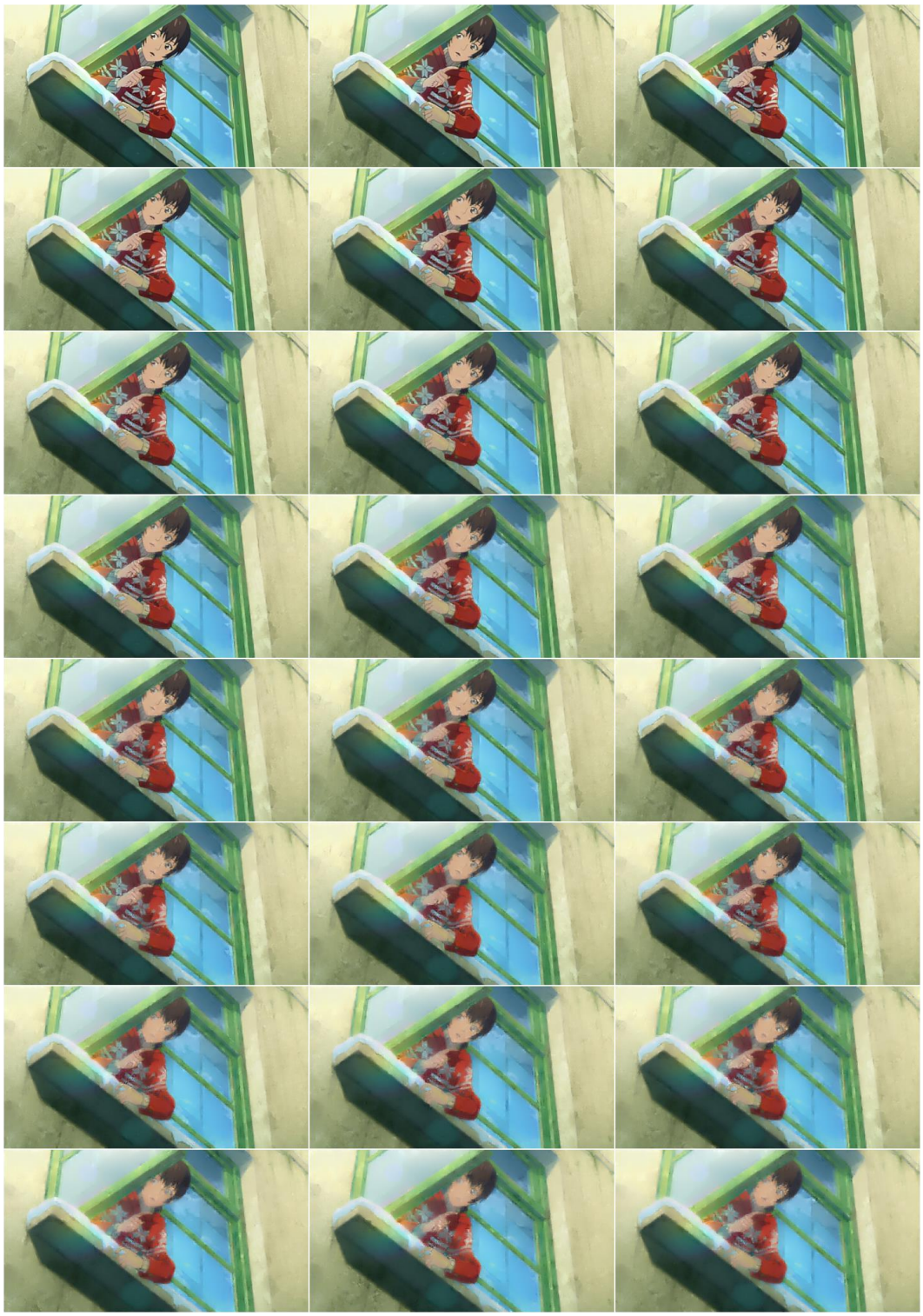

Figure 8. Examples of LR frames degraded by multi-scale VQGAN in multi-levels. From top to bottom, left to right, we show the degradation levels in ascending order. For the sake of clear comparisons, we show the results every 3 levels in the first 70 levels ($k = 1, 4, 7, ..., 70$) here. Input HR: AVC-Train/$b0034m9wleq.10005\_movie001\_0/00000049.png$

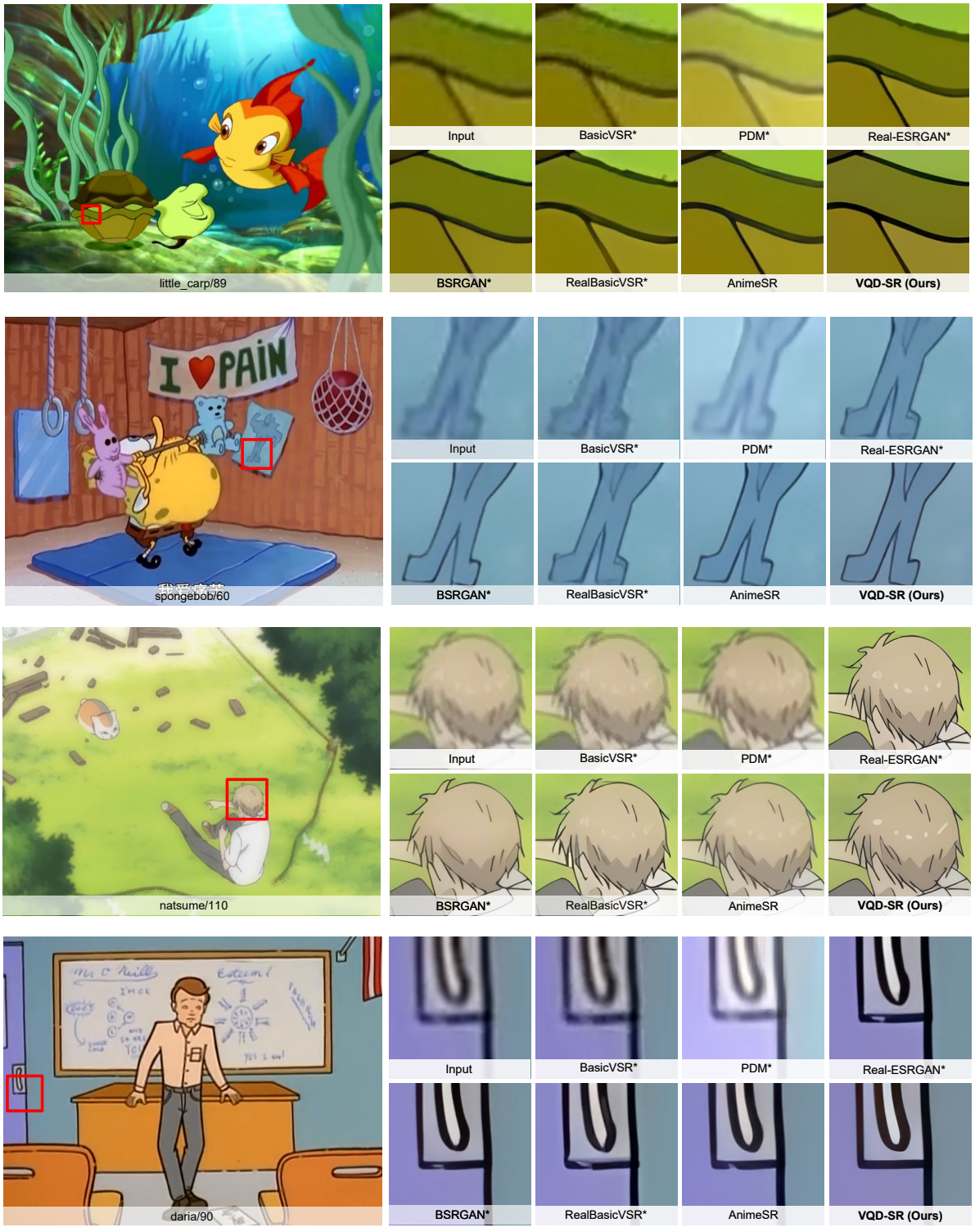

Figure 9. Qualitative comparisons with SOTA methods. '$*$' denotes fine-tune on animation dataset AVC-Train [8]. Our VQD-SR is capable to recover visually natural and sharper lines with fewer artifacts.

Figure 10. Qualitative comparisons with SOTA methods. '$*$' denotes fine-tune on animation dataset AVC-Train [8]. Our VQD-SR is capable to restore clear details.

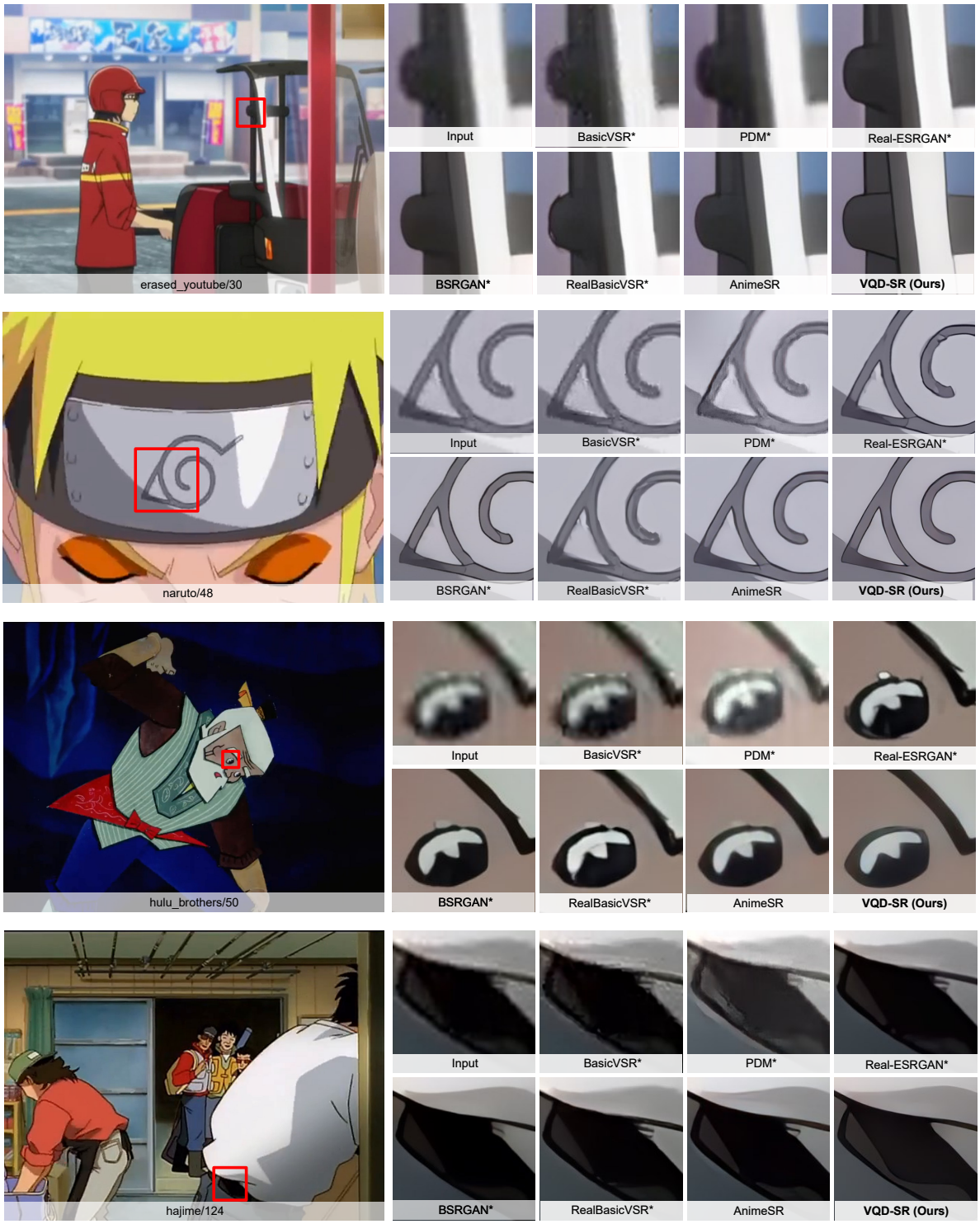

Figure 11. Qualitative comparisons with SOTA methods. '$*$' denotes fine-tune on animation dataset AVC-Train [8]. Our VQD-SR is capable to recover visually natural and sharper lines with fewer artifacts.

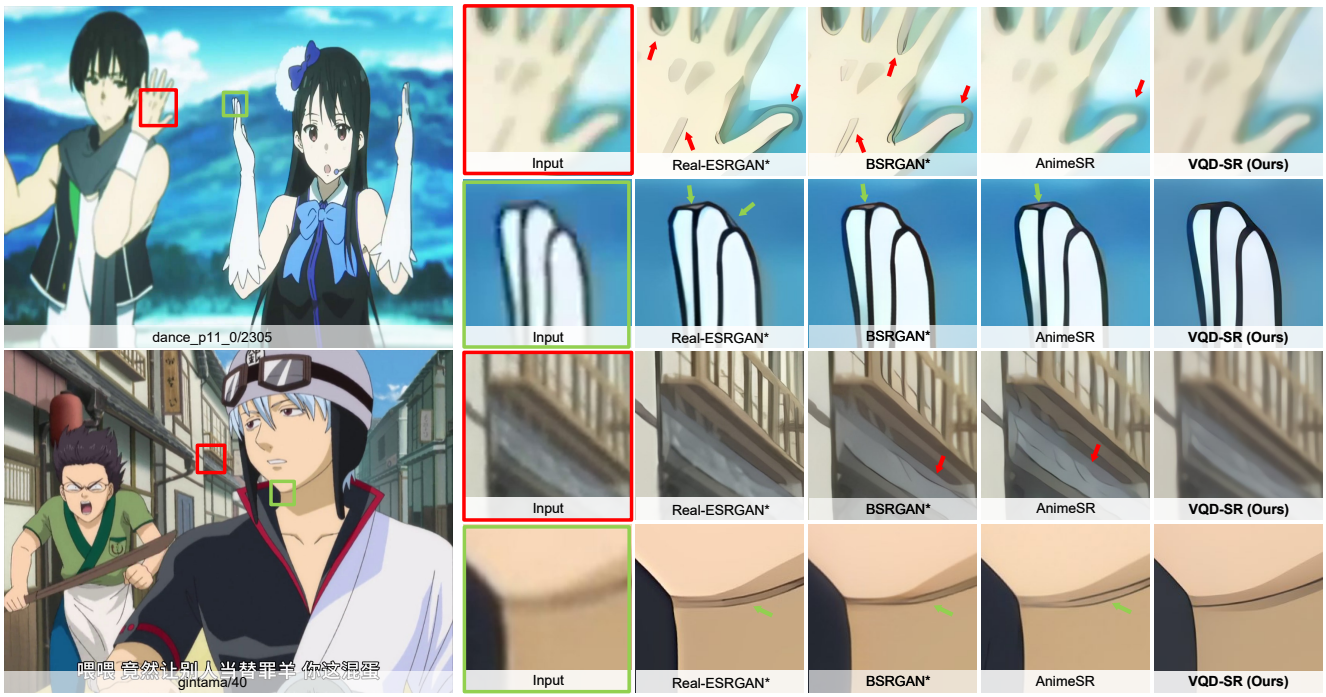

Figure 12. Qualitative comparisons with SOTA methods. '$*$' denotes fine-tune on animation dataset AVC-Train [8]. Our VQD-SR is capable to handle some intended scenarios (*e.g.*, the out-focus background blur) with fewer over-sharp artifacts. The red crops indicate objects in the out-focus background which should be naturally smooth and the green crops indicate the foreground objects which should be clear and sharp.

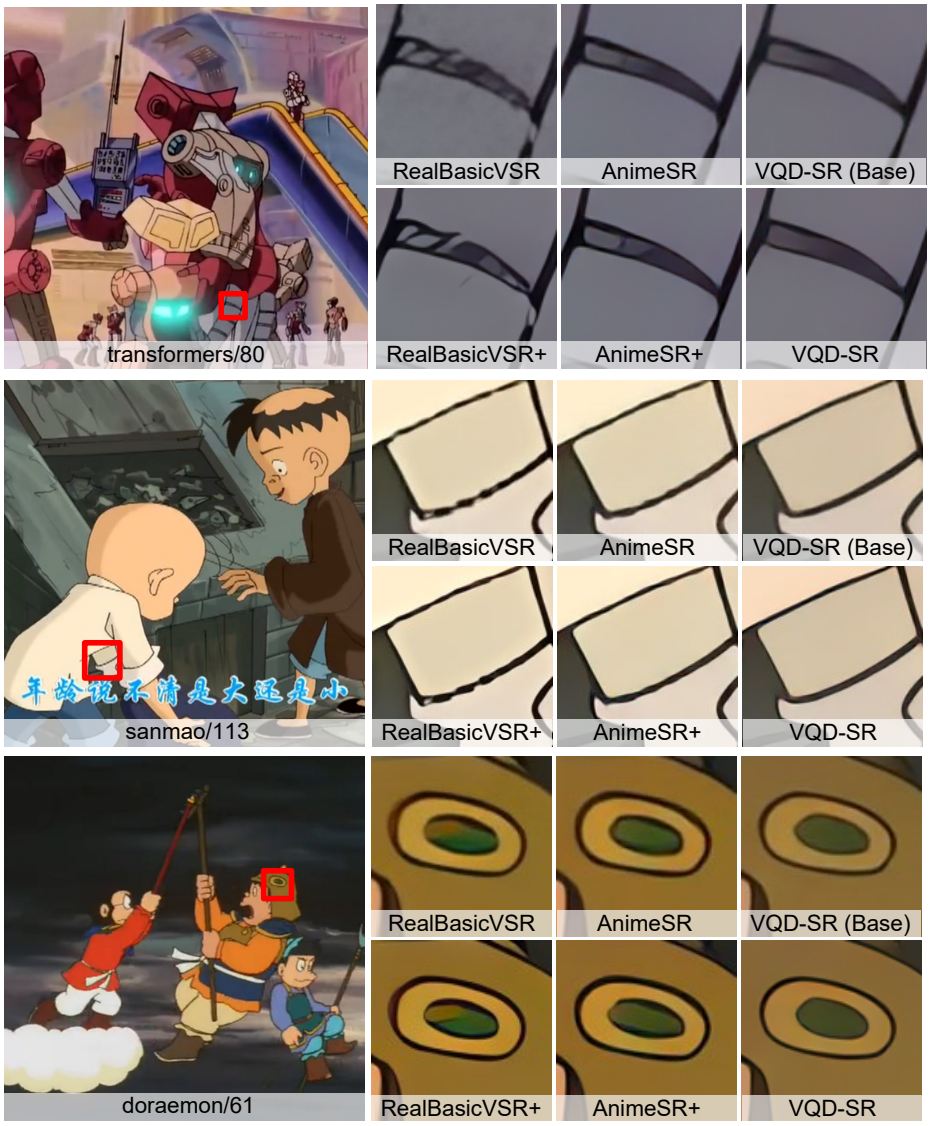

Figure 13. Ablation study of HR-SR enhancement for different VSR methods.

Figure 14. Visual comparison of different SR models for HR-SR enhancement in animation domain.

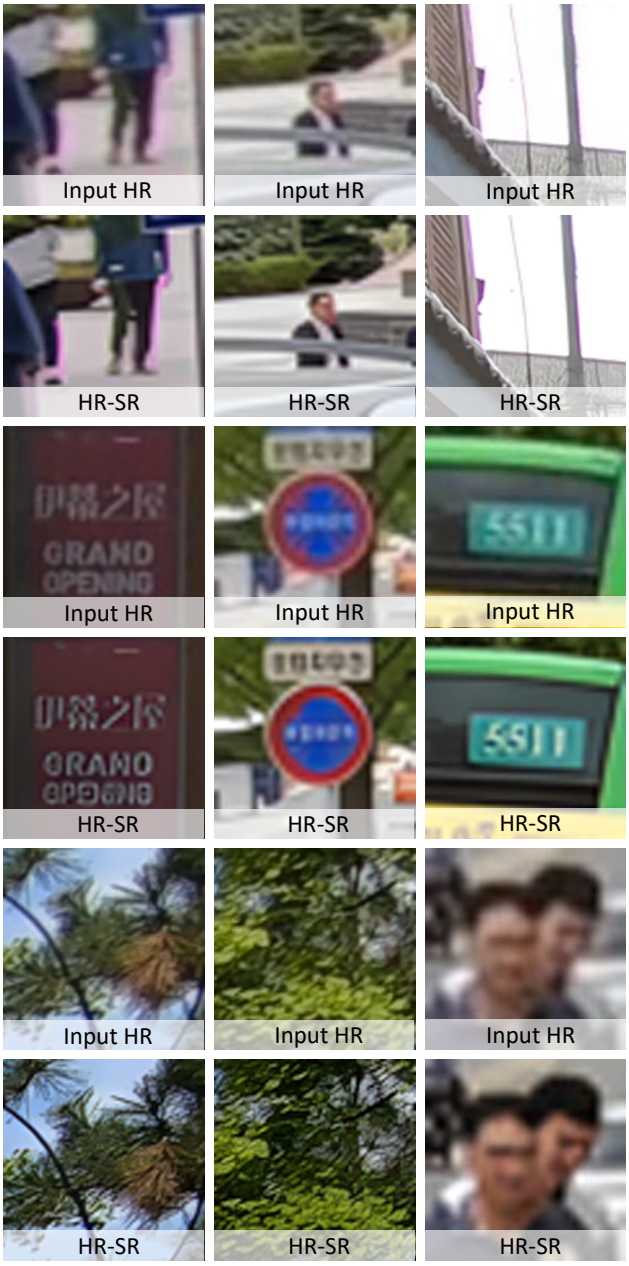

Figure 15. Extend HR-SR enhancement strategy to natural videos. Complex textures and irregular illumination conditions lead to amplified illumination artifacts (row 1), contaminated details (row 2), and over-sharp textures (row 3).