Методы и алгоритмы улучшения качества изображения
Method and algorithm for image quality enhancement

Сунь Чжунао ВМК

Main Task:

Single image super-resolution (SR) is an active research topic, which aims at reconstructing a high- resolution (HR) image from its low-resolution (LR) counterpart.

Bicubic | ESRGAN | RealSR | Real-ESRGAN

Improvement of this research

- in this work, 1) they propose a high-order degradation process to model practical degradations, which is replaced the traditional Bicubic to do the down-sample,   and utilize sinc filters to model common ringing and overshoot artifacts.

- 2) they employ several essential modifications (e.g., U-Net discriminator with spectral normalization) to increase discriminator capability and stabilize the training dynamics.

Classical Degradation Model

$$x = \mathcal{D}(y) = [(y \circledast k) \downarrow_r + n]_{\mathrm{JPEG}}, \tag{1}$$

- y – the ground truth
- k – the blur kernel
- r - the scale factor of downsampling operation
- n – noise
- JPEG compression is also adopted, as it is widely-used in real-world images.

second-order degradation

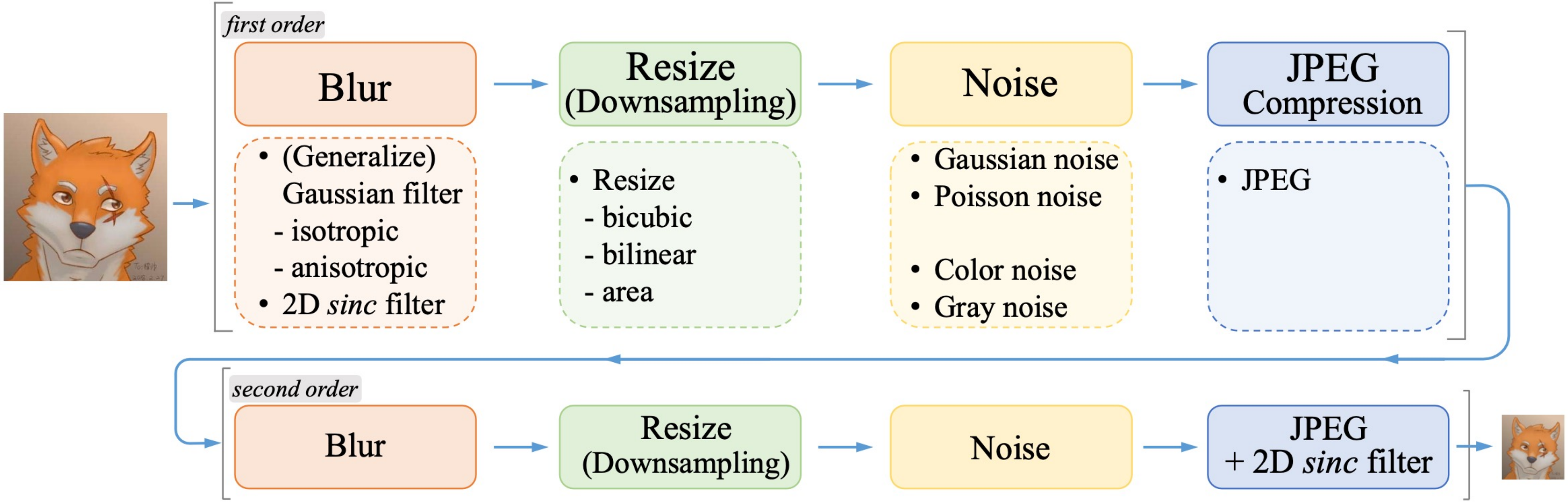


Figure 2: Overview of the pure synthetic data generation adopted in Real-ESRGAN. It utilizes a second-order degradation process to model more practical degradations, where each degradation process adopts the classical degradation model. The detailed choices for blur , resize, noise and JPEG compression are listed. We also employ *sinc* filter to synthesize common ringing and overshoot artifacts.

Ringing and overshoot artifacts

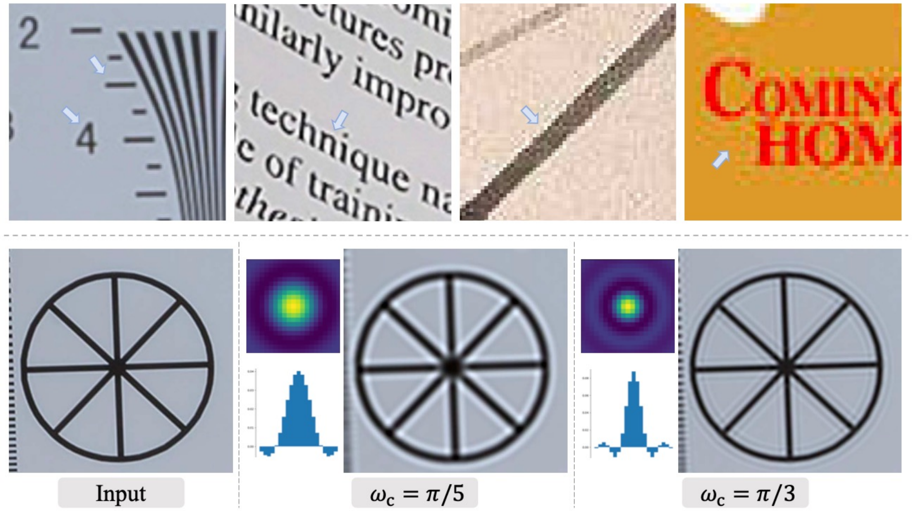


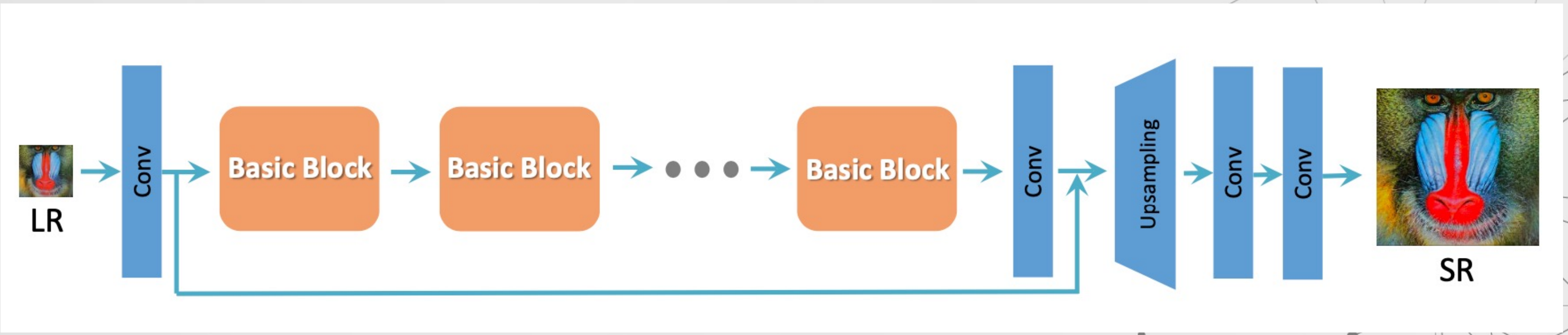
Figure 5: **Top**: Real samples suffering from ringing and overshoot artifacts. **Bottom**: Examples of *sinc* kernels (kernel size 21) and the corresponding filtered images. Zoom in for best view

- We employ the *sinc* filter, an idealized filter that cuts off high frequencies, to synthesize ringing and overshoot artifacts for training pairs. The *sinc* filter kernel can be expressed as:

$$\boldsymbol{k}(i,j) = \frac{\omega_c}{2\pi\sqrt{i^2+j^2}} J_1(\omega_c\sqrt{i^2+j^2}), \qquad (6)$$

Networks structure

- **ESRGAN generator：** They adopt the same generator (SR network) as ESRGAN [ESRGAN: Enhanced Super-Resolution Generative Adversarial Networks], *i.e.*, a deep network with several residual-in-residual dense blocks (RRDB),



- **U-Net discriminator with spectral normalization (SN)：**
- The U- Net outputs realness values for each pixel, and can provide detailed per-pixel feedback to the generator.

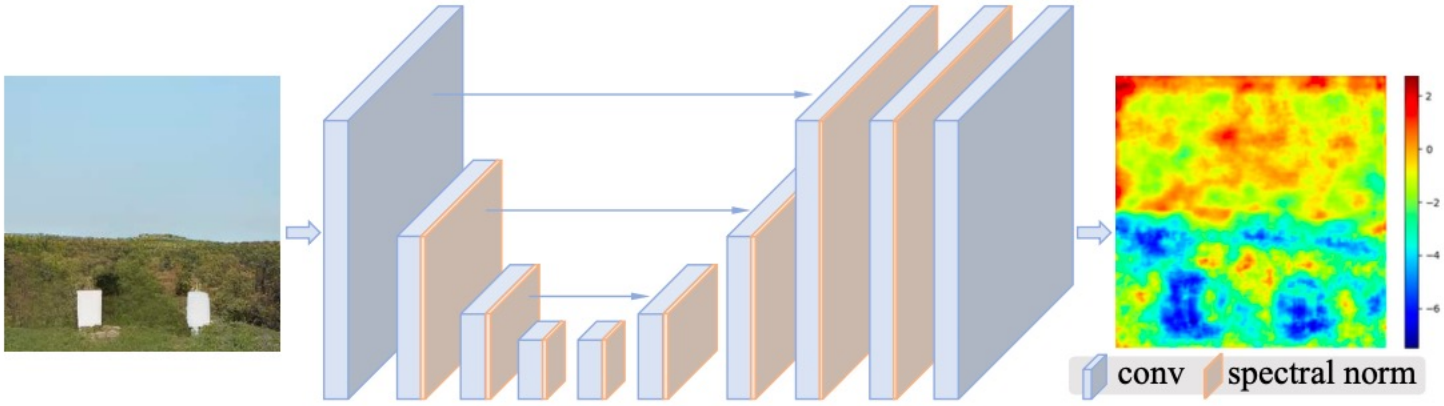


Figure 6: Architecture of the U-Net discriminator with spectral normalization.

The training process

- The training process is divided into two stages. First, they train a PSNR-oriented model with the L1 loss. The obtained model is named by *Real-ESRNet*. We then use the trained PSNR-oriented model as an initialization of the generator, and train the *Real-ESRGAN* with a combination of L1 loss, perceptual loss *[Perceptual losses for real-time style transfer and super-resolution. In ECCV, 2016]* and GAN loss [Photo- realistic single image super-resolution using a generative ad- versarial network. In *CVPR*, 2017].

2.2. Perceptual loss function

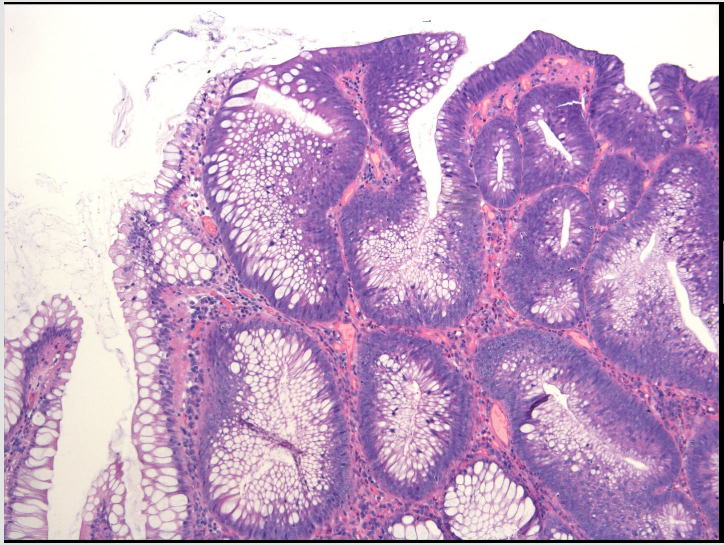
The definition of our perceptual loss function l^{SR} is critical for the performance of our generator network. While l^{SR} is commonly modeled based on the MSE [10, 48], we improve on Johnson et al. [33] and Bruna et al. [5] and design a loss function that assesses a solution with respect to perceptually relevant characteristics. We formulate the perceptual loss as the weighted sum of a content loss (l_X^{SR}) and an adversarial loss component as:

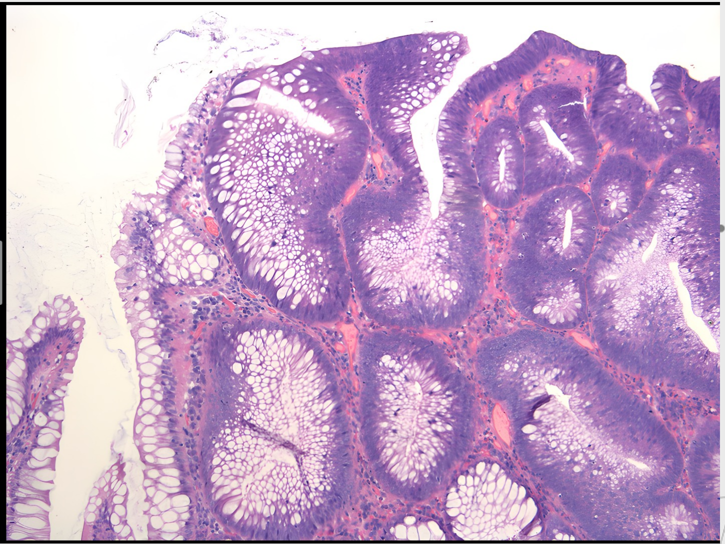
$$l^{SR} = \underbrace{\underbrace{l_X^{SR}}_{\text{content loss}} + \underbrace{10^{-3} l_{Gen}^{SR}}_{\text{adversarial loss}}}_{\text{perceptual loss (for VGG based content losses)}} \tag{3}$$

In the following we describe possible choices for the content loss l_X^{SR} and the adversarial loss l_{Gen}^{SR}.

$$l_{Gen}^{SR} = \sum_{n=1}^{N} -\log D_{\theta_D}(G_{\theta_G}(I^{LR})) \tag{6}$$

Here, $D_{\theta_D}(G_{\theta_G}(I^{LR}))$ is the probability that the reconstructed image $G_{\theta_G}(I^{LR})$ is a natural HR image. For better gradient behavior we minimize $-\log D_{\theta_D}(G_{\theta_G}(I^{LR}))$ instead of $\log[1 - D_{\theta_D}(G_{\theta_G}(I^{LR}))]$ [22].



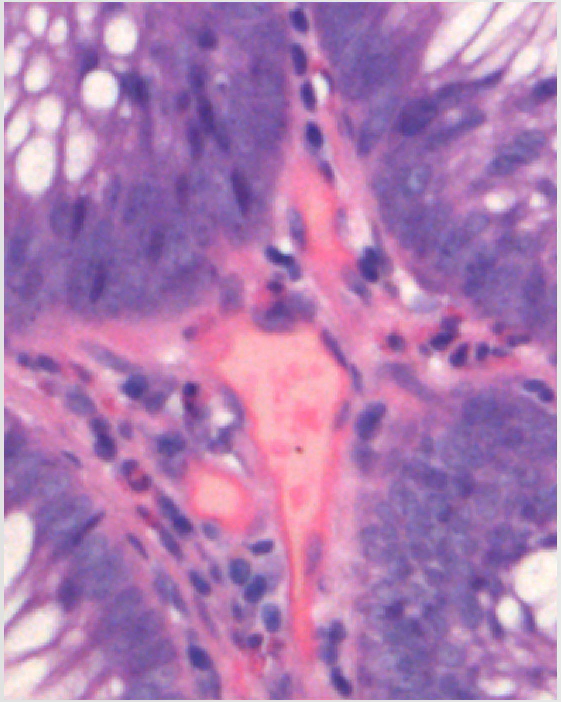


- 3632 × 2442

- 3630 × 2440

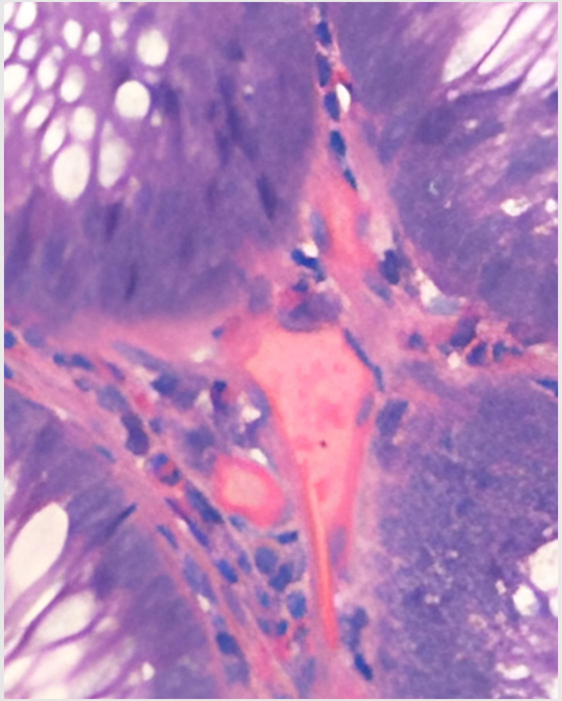
One Part of Image

reference



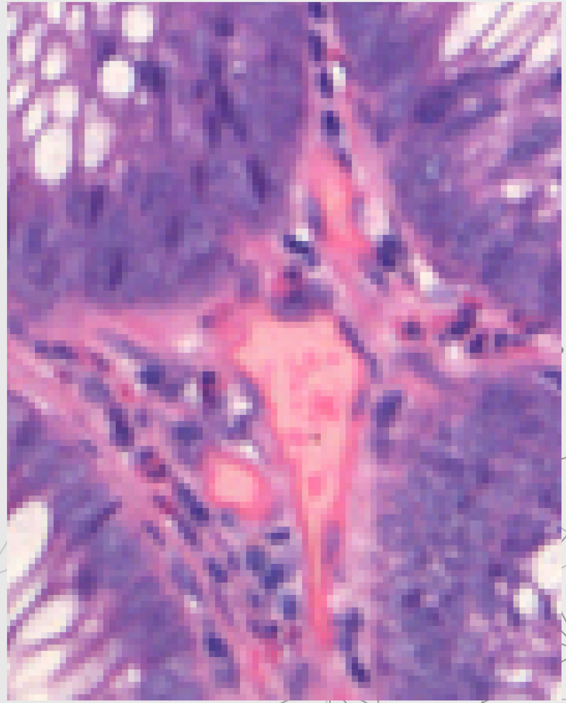
- 3632 × 2442

Result



- 3630 × 2440

LR



- 815 × 610

THANKS

Сунь Чжунао ВМК