EESR：Edge Enhanced Super-Resolution

Ying-Yue Li[1], Yun-Dong Zhang[2], Xue-Wu Zhou[2], Wei Xu [1]*
[1]State Key Laboratory of ASIC and System, Fudan University, No.825 ZhangHeng Rd. Shanghai 201203, China
[2] Vimicro AI Chip Technology Corporation, No. 35, XueYuan Road, Beijing 100191, China
* Email: wei_xu@fudan.edu.cn

Abstract
In this paper, an edge enhanced super-resolution network (EESR) is proposed for a better generation of high-frequency structures in blind super resolution. In EESR, an edge detection network is used to extract high-frequency pixels. A generative adversarial network (GAN) is used to fine-tune the EESR network. And a more applicative criterion "perceptive texture score" (PTS) is defined and used to evaluate the quality of generated high-resolution image. Experiments show that the new EESR is able to recover textures with 4 times up-sampling, and gained PTS of 0.6210 on DIV2K test set, which is much better than the state-of-the-art methods.

(a) (b)

Figure 1. The original HR image (a) and this paper' result (b), recovered from the original's 4 times down-sampled LR image.

1. Introduction

Super-resolution (SR) is widely used in image processing. There are mainly two types of SR: blind SR and non-blind SR. For non-blind SR, a higher resolution image is generated by exploiting image priors [1], or by exploiting models with simplified degradation [2]. For blind SR, priors and degradation methods are unknown, which brings more challenges to SR processing.

Traditional interpolation-based methods (such as bilinear, bicubic and Lanczos) generate high-resolution (HR) pixel intensities by weighted neighboring low-resolution (LR) pixels. They generate good smooth regions but are less effective for modeling high-frequency structures such as textures. Hence, recent researches tend to use CNNs (Convolutional Neural Network) for a more effective generation of the high-frequency structures [3] [4] by designing different network depths and patterns. However, most of them focus mainly on minimizing the MSE (mean square error). Thus, although they are able to have high PSNR (peak signal to noise) and SSIM (structural similarity), their resulting images appear to be over-smooth. SRGAN [6], by using GAN [7] architecture, is the first framework capable of inferring photo-realistic natural images. It has a pretty generation of high-frequency structures, better than most existing blind SR methods. However, SRGAN is hard to train [8].

This paper proposes an edge enhanced super-resolution (EESR) algorithm for blind SR, with its indistinguishable SR processing results shown in Figure 1. Our major contributions are as follows:

1) An edge detection network is used to extract and preserve high-frequency structures.
2) GAN architecture is used for fine-tuning the generative network, instead of direct training.
3) A new criterion "perceptive texture score" (PTS) is raised for better evaluating an image's high-frequency structures.

The rest of this paper is as follows. In section 2, we introduce the related works. In section 3, we present the architecture and detail design of our EESR network. In section 4, we present our training and inferencing process and experimental results. Section 5 is the summary.

2. Related work

2.1 HED edge detection model

Xie *et al*. [9] developed a deep supervised edge and boundary detection network named Holistically Nested Edge Detection (HED). Instead of supervising the edge features of the final output, HED applies supervision on all the side layers. By resolving the challenging ambiguity, HED has a well acceptable performance in edge detection on the BSD500 dataset (ODS F-score of 0.782) and NYU Depth dataset (ODS F-score of 0.746). In EESR, we use the HED model for edge detection.

2.2 SRGAN

SRGAN [6] uses a discriminator network to discriminate real HR images from generated SR samples. In SRGAN, a perceptual loss function consisting of an adversarial loss and a content loss is used. The adversarial loss pushes the solution to the natural image manifold, while the content loss is motivated by perceptual similarity instead of similarity in pixel space. In addition to PSNR and SSIM, SRGAN uses mean opinion score (MOS) as its criterion to evaluate the quality of the generated HR image. Using GAN architecture, SRGAN is able to generate images with good details and vivid textures, better than most CNN methods. But due to its use of GAN architecture, SRGAN is difficult to train and converge.

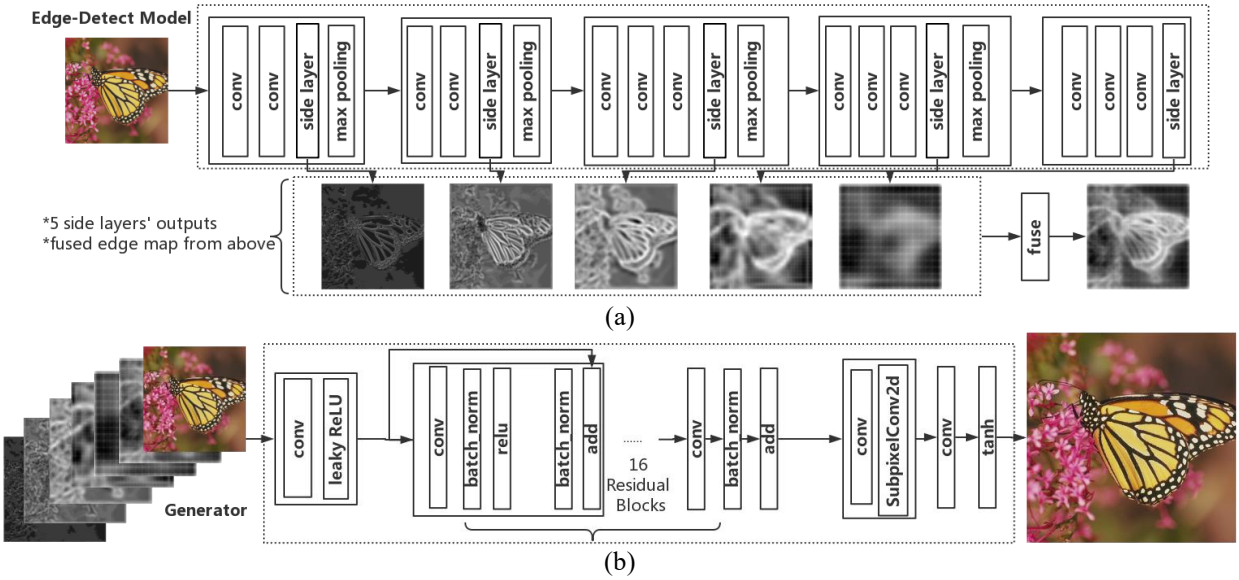

Figure 2. Generative network architecture of edge detection model (a) and Generator model (b).

3. Our approach

As shown in Figure 3, EESR has a generative network for training and a discriminative network for fine-tuning.

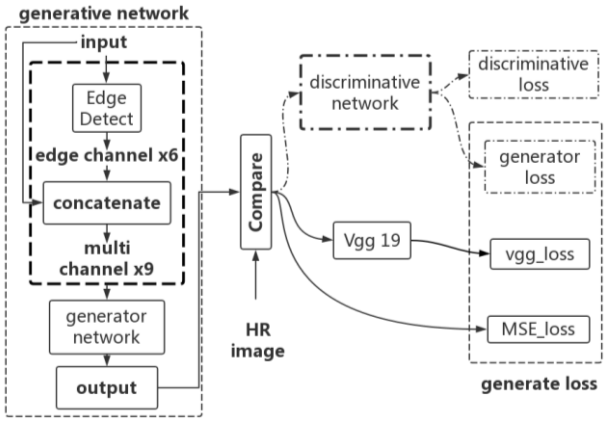

Figure 3. Main architecture of EESR

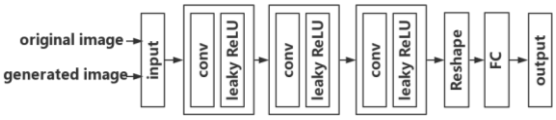

Figure 4. Discriminative network

3.1 Generative network

The generative network, consisting of an edge detection network and a generator network, is responsible for mapping LR image I^{LR} to SR image $G(I^{LR})$.

We use HED [6] model for edge detection. As shown in Figure 2(a), 5 tappeded out edge maps, together with 1 fused map, are concatenated with 3-channel RGB (from original LR image) to form a 9-channel input map of the generator network. With multi-scale and multi-level input feature maps, the generator network is therefore able to have a good preservation of high-frequency features of the original image throughout training. To fit scalable image size, subpixel convolutional layers [10] are used to form a fully convolutional network, as shown in Figure 2(b).

3.2 Discriminative network

The discriminative network discriminates SR samples from original HR images, and induces the generative network to generate indistinguishable SR images. Different from direct training in SRGAN, EESR uses the discriminative network on fine-tuning its network, which makes the network more stable and easier to converge. Besides, its reduced structure of 3 convolutional layers is more concise than that in SRGAN, as shown in Figure 4.

3.3 Loss function

EESR network has two loss functions: generative and discriminative. Discriminative loss is optimized during fine-tuning, shown as follows:

$$L_{GAN-D} = \mathbb{E}_{\delta}(D(I^{HR}), \mathbb{I}) + \mathbb{E}_{\delta}(D(G(I^{LR})), \mathbb{O}), \quad (1)$$

where $\mathbb{O}$ and $\mathbb{I}$ are matrixes with the same shape as I^{HR}, whose elements equal to 0 and 1 separately. $\mathbb{E}_{\delta}$ is a sigmoid cross entropy function to ensure stability and avoid overflow.

Instead of applying pixel-wise MSE loss only, A VGG-19 network [5] is used to extract image features for calculating losses. The generative loss contains generator loss from GAN L_{GAN-G}, MSE loss L_{MSE} and VGG loss L_{VGG}.

$$L_{GAN-G} = \mathbb{E}_{\delta}(D(G(I^{LR})), \mathbb{I}) \quad (2)$$

$$L_{MSE} = \frac{1}{r^2WHC}\sum_{x=1}^{rW}\sum_{y=1}^{rH}\sum_{z=1}^{C}(I_{x,y,z}^{HR} - G(I^{LR})_{x,y,z})^2 \quad (3)$$

$$L_{VGG} = \frac{1}{MNK}\sum_{i=1}^{M}\sum_{j=1}^{N}\sum_{k=1}^{K}(V(I^{HR})_{i,j,k} - V(G(I^{LR}))_{i,j},k)^2, \quad (4)$$

where V is the mapping procedure from image to VGG layers output in size of $M \times N \times K$. The final loss function of generative network is a weighted mean of these three losses.

4. Experiments and analysis

4.1 Evaluation

In SRGAN, MOS is obtained from 26 raters, depending too much on subjective judgment, and is hard to reproduce. Instead, we use a more general and objective operator, called perceptive texture score (PTS), to quantify our super-resolved images, with its definition as follows:

$$\text{PTS} = \frac{\sum_i^W \sum_j^H abs(G(I^{LR})\otimes \mathbb{K}_{PTS})_{i,j}}{\sum_i^W \sum_j^H abs(I^{HR}\otimes \mathbb{K}_{PTS})_{i,j}}, \quad (5)$$

where $G(I^{LR})$ is the SR image and $\mathbb{K}_{PTS}$ is a 3-channel matrix with each channel set to be as follows:

$$\mathbb{K}_{PTS}[i] = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}, (i = 0,1,2). \quad (6)$$

$\mathbb{K}_{PTS}$ is used to calculate the differences between a pixel and its neighboring pixels, which are highly relative to the details and textures of the image. Therefore, instead of concentrating on pixels' similarity and smoothness in PSNR and SSIM, PTS can be more appropriate to evaluate the preservation of the high frequency details.

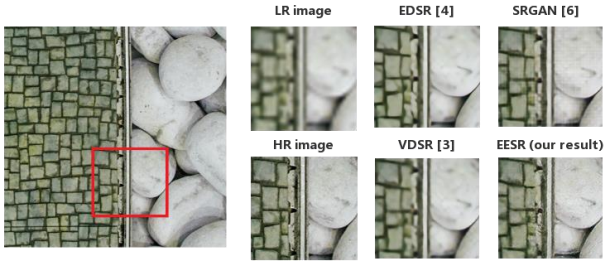

Figure 5. 4 times super resolution result of our EESR compared with existing algorithms.

Table 1. Comparison with previous works on DIV2K

	Bicubic	VDSR[3]	SRGAN[6]	EDSR[4]	EESR
PSNR	22.2467	23.5969	23.4090	**23.7904**	22.3501
SSIM	0.7414	0.8102	0.8082	**0.8288**	0.7588
PTS	0.0767	0.1870	0.2383	0.2287	**0.6210**

4.2 Experimental results

We implement our EESR on TensorFlow, and train it on DIV2K dataset using one NVIDIA TITAN XP GPU. In the experiments, the learning rate is set to 1e-4 initially and decayed to 1e-5 after 1000 epochs to increase the stability of the fine-tuning. The batch size is set to 16. The generative network is trained for 200 epochs, followed by a 2000-epoch fine-tuning.

Figure 5 shows the resulting images from 4 times up-sampling of EESR and 4 other state-of-the-art SR methods. We can see that SRGAN performs much better than EDSR and VDSR which are apparently over-smooth, but also brings in texture noise. Obviously, EESR performs the best in the preservation of high frequency details and the perceptional similarity. Table 1 also tells the same result. That is, although all the 5 methods get similar values in PSNR and SSIM, EESR scores much higher in PTS which means an overwhelming performance in SR processing.

5. Summary

In this paper, we propose a new blind SR method. We focus our approach on edge pixels by introducing an edge detection model before training the generative network. We use GAN network to fine-tune our network instead of direct training. Experimental results on DIV2K show that our EESR method reaches an outstanding performance for 4 times up-sampling in preserving high-frequency structures and generating high quality SR images.

References

[1] Yu Chen, Ying Tai, Xiaoming Liu *et al.* arXiv:1711.10703 (2017).

[2] Kai Zhang, Wangmeng Zuo, Lei Zhang. Computer Vision and Pattern Recognition (CVPR) (2018).

[3] Jiwon Kim, Jung Kwon Lee and Kyoung Mu Lee. Computer Vision and Pattern Recognition (CVPR), pp 1063-6919 (2016).

[4] Bee Lim, Sanghyun Son, Heewon Kim *et al.* CVPR workshop (2017).

[5] K. Simonyan and A. Zisserman. International Conference on Learning Representations (2015).

[6] Christian Ledig, Lucas Theis, Ferenc Husz´ar *et al.* The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 4681-4690 (2017).

[7] Ian J. Goodfellow, Jean Pouget-Abadiey, Mehdi Mirza *et al.* Conference and Workshop on Neural Information Processing Systems (NIPS), pp 2672-2680 (2014).

[8] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky *et al.* Conference and Workshop on Neural Information Processing Systems (NIPS) (2017).

[9] Saining Xie, Zhuowen Tu. IEEE International Conference on Computer Vision (ICCV), pp 1395-1403 (2015).

[10] Wenzhe Shi, Jose Caballero, Ferenc Huszár *et al.* Computer Vision and Pattern Recognition (CVPR), pp 1063-6919 (2016).