Multiple Connected Residual Network for Image Enhancement on Smartphones

Jie Liu(✉) and Cheolkon Jung(✉)

School of Electronic Engineering, Xidian University, Xi'an 710071, Shaanxi, China
jieliu543@gmail.com, zhengzk@xidian.edu.cn

Abstract. Image enhancement on smartphones needs rapid processing speed with comparable performance. Recently, convolutional neural networks (CNNs) have achieved outstanding performance in image processing tasks such as image super-resolution and enhancement. In this paper, we propose a lightweight generator for image enhancement based on CNN to keep a balance between quality and speed, called multi-connected residual network (MCRN). The proposed network consists of one discriminator and one generator. The generator is a two-stage network: (1) The first stage extracts structural features; (2) the second stage focuses on enhancing perceptual visual quality. By utilizing the style of multiple connections, we achieve good performance in image enhancement while making our network converge fast. Experimental results demonstrate that the proposed method outperforms the state-of-the-art approaches in terms of the perceptual quality and runtime. The code is available at https://github.com/JieLiu95/MCRN.

Keywords: Image enhancement · Generator · Residual Network
Multiple connections · Perceptual quality

1 Introduction

Due to the demand for easy manipulation and the increase of visual quality in smartphones, numerous people choose to take photos using their phone cameras. In general, high-resolution (HR) images need a better sensor to keep image fidelity, resulting in additional cost. Image enhancement on smartphones is required to provide a higher visual quality. It can be achieved by learning the relationship between photos of smartphones and DSLR-quality images. It generates a DSLR-like image from an input image obtained by smartphones. However, it still runs on our typically used appliances with a limit of computing resources, and many GPUs are not available in a real situation. Thus, real-time processing is required for image enhancement in smartphones, and thus a lightweight solution is needed.

Up to the present, many outstanding studies have been done. In DSLR Photo Enhancement Dataset (DPED) [1], the authors successfully keep the texture and perceptual information by a generative adversarial network [2]. They also

L. Leal-Taixé and S. Roth (Eds.): ECCV 2018 Workshops, LNCS 11133, pp. 182–196, 2019.
https://doi.org/10.1007/978-3-030-11021-5_12

provide a large dataset, which contains paired images of the same scene obtained by smartphones (e.g., iPhone, Sony, and BackBerry).

(a) Original image. (b) Enhanced image.

Fig. 1. Sample image. (a): Input original image from iPhone 3GS. (b): Enhanced image by our method.

In this paper, a lightweight generator for smartphones based on CNN to keep a balance between quality and speed is proposed, called multi-connected residual network (MCRN). In Fig. 1, we pick a pair of images to show the visual effect generated by proposed method. Ignatov et al. proposed a weakly supervised photo enhancer, named WESPE [3]. They mainly focus on image enhancement for unpaired images. WESPE improves quality in texture and structure, but it is somewhat slow in inference phase. However, their model is huge even in a high-end workstation. Therefore, it is impossible to put them on smartphones, and it is hard to balance speed and quality.

To address this problem, we introduce a generative network to accelerate it, named MCRN, as shown in Fig. 2. We use a small generator to speed-up image enhancement. The generator only has 4 convolution layers and each of them has 16 channels. Because every layer of the generator learns a few features, we design multiple connected modes to maximize the flow of information [4] from different levels of features, thus improving visual quality of the images obtained by smartphones. Besides, a loss function based on a discriminator of generative adversarial network (GAN) is proposed, which strengthens and fine-tunes details of the estimated map by the generator. The proposed MCRN is composed of a two-stage generator and a discriminator to get good visual quality. The discriminator is shown in Fig. 3, and it uses adversarial loss to synthesize textures and details.

The main contributions of this paper include:

(1) We propose a lightweight end-to-end network to learn a model to map smartphone images into DSLR ones. Moreover, it consists of only 4 layers, which is very small in deep learning approaches.

(2) We use a two-stage network architecture in the generator. For the first stage, the generator keeps the structure feature by SSIM loss, while for the second stage, it keeps high dimensional semantic information.
(3) We adopt multiple connections in the generator to reuse resource and get rich features.

2 Related Work

Image enhancement methods are classified into global and local approaches in adjusting contrast and color mapping. In power-law contour detection [5] and gamma function [6], non-linear functions avoid saturation in bright regions while successfully preserving an image tone. However, they have a limit in enhancing local regions. In the past decades, histogram equalization and its variants, such as contrast limited [7] and brightness preserving [8], are widely used for enhancement to achieve better contrast. However, they are very sensitive to the change of parameters because of manually adjusting the image correction. Thus, they result in detail loss and over-exposure at local areas.

GAN. Generative adversarial network [2] is used to generate good quality images with fine details because it can learn data distributions. Recently, more and more domain translation tasks have used GAN to get an adversarial loss, which synthesizes good features and textures. However, in a specific task, GAN plays a special role, e.g. it is style loss in [9], and it is used to color loss in Gateways *et al.*'s work [10]. Chen *et al.* [11] proposed two-way GAN, in which they used U-Net [12] as global generator and an adaptive weighting scheme on WGAN [13], to improve the quality of enhanced images. Motivated by them, we propose an adversarial loss function based on GAN to fine-tune texture details in this work.

Super-Resolution. Single image super-resolution is a significant problem, which aims to generate an HR image from its low-resolution (LR) one. SRCNN [14] proposed by Dong *et al.* is the first method to solve single image super-resolution using CNN. Ledig *et al.* proposed SRGAN [15], which utilizes generative adversarial network to recover the HR images with high perceptual quality. In addition, the NTIRE 2018 Challenge on Single Image Super-Resolution [16] has achieved good results, and many teams proposed novel methods and got good scores. In this challenge, most of the methods are based on ResNet [17] and DenseNet [18] and achieve higher PSNR score. Inspired by CondenseNet [4], we use multiple connections to keep image fidelity and visual quality.

Image-to-Image Mapping. Image enhancement [1,11], style transfer [19,20] and color transfer [21–24] are the sub-tasks of image-to-image transfer [25,26]. Okura *et al.* [21] proposed a novel method based on comparing the exemplar with the source for color and texture transfer, which achieved good performance. Liu *et al.* [20] proposed a data-driven system to automatically transfer style to a user's photos. Wang *et al.* [27] proposed style and structure GANs, and achieved good performance in image generation. Two GANs were trained independently

and then learned via joint learning. Huang *et al*. [28] proposed a stacked GAN that was a multi-GAN model to learn representation from top to down for image generation. They used multi-stage models which were very big. Image enhancement on smartphones is also an image-to-image mapping operation. In consideration of speed and memory limitation, we adopt two-stage generator.

Image Denoising and Artifact Removal. The images captured by smartphones have noises which are not obvious without enlarging the image, but this phenomenon leads to severe degradation of image quality. In most methods, e.g. DPED [1], artifacts remain on their results. Thus, in this paper we achieve denoising and artifact removal based on MSE and total variation losses [29] in the training stage.

3 Proposed Method

3.1 Network Architecture

Generator. The generator of MCRN is illustrated in Fig. 2, which is composed of 4 layers. The first 1×7 layer and the second 7×1 layer are combined with a big receptive field layer to get features. The receptive field is 7×7, and the number of the first parameters is only $\frac{2}{7}$th of the second ones. Then, *Output OC* is used to keep structure information by SSIM loss as shown in Fig. 2, which is defined as follows:

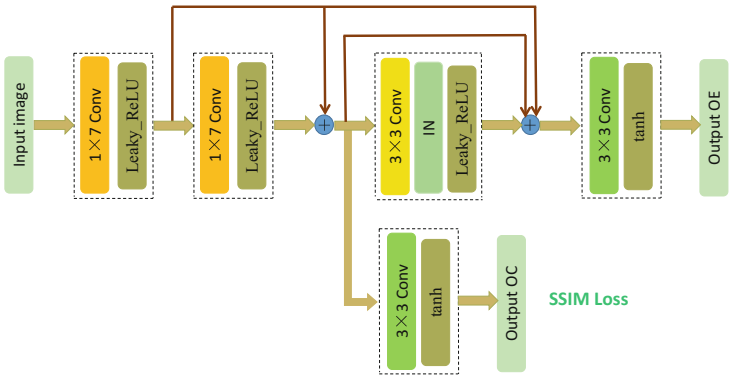

Fig. 2. Network pipeline of the proposed generator. The proposed generator consists of 5 layers in training phase and 4 layers in testing phase (remove the last layer of the first stage supervised). Input: Images captured from smartphones cameras. Output: OE composed of 4 layers. (Output *OC*: Output of the first stage output image; Output *OE*: Enhanced image by proposed method).

$$\mathcal{L}_{SSIM} = \sum_{n=1}^{N} 1 - SSIM, \tag{1}$$

where $SSIM$ (structural similarity) is calculated by $Outout\ OC$ (output of the generator) and the ground truth. $\mathcal{L}_{SSIM}$ is only used in the first stage and updated successively. In addition, instance normalization (IN) [30] operation is used after a convolution in the third layer. According to the Ulyanov *et al.*'s work [30], IN layer is defined as follows:

$$y_{bchw} = \frac{x_{bchw} - \mu_{bc}}{\sqrt{\sigma_{bc}^2 + \varepsilon}}, \tag{2}$$

where b, c, h, w are batch size, feature channel, height, and width respectively. In addition, μ_{bc} and σ_{bc}^2 are mean and covariance respectively, and they are defined as follows:

$$\begin{aligned} \mu_{bc} &= \frac{1}{HW} \sum_{w=1}^{W} \sum_{h=1}^{H} x_{bchw}, \\ \sigma_{bc}^2 &= \frac{1}{HW} \sum_{w=1}^{W} \sum_{h=1}^{H} (x_{bchw} - h\mu_{bc})^2. \end{aligned} \tag{3}$$

Since batch normalization (BN) slightly hurts color consistence, it is replaced with IN. The training phase is stable under instance normalization.

In Fig. 2, $\oplus$ is the element-wise summation operator. Since this framework is too shallow to get enough feature, we put the element-wise add operator into the generator to use the feature from the previous layers.

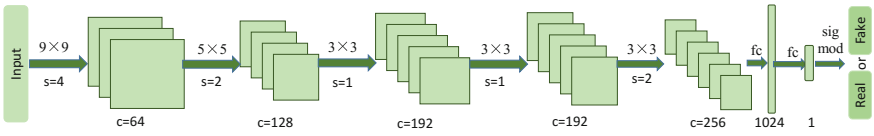

Fig. 3. Network architecture of the proposed discriminator. The discriminator consists of 5 convolutional layers and 2 fully connected layers. An activation function produces real and fake, while the generator tries to cheat the discriminator.

Discriminator. The discriminator is very simple as shown in Fig. 3. In the first step, we get the down-sampled image by stride convolution, and the number of channels increase layer by layer. The optimization function of the discriminator is defined as follows:

$$\mathcal{L}_D = -\log\left(D\left(I_g\right)\right) - \log\left(1 - D\left(G\left(I_x\right)\right)\right), \tag{4}$$

where I_g is the ground truth, and I_x is the input of the generator, which is captured by smartphones. D and G are discriminator and generator, respectively. Discriminator needs to distinguish the ground truth and the enhanced image $Output\ OE$ by the generator.

3.2 Loss Functions

Adversarial Loss. The discriminator is mainly used to optimize the generator by an adversarial loss as follows:

$$\mathcal{L}_{adv} = -\log\left(D\left(I_{OE}\right)\right), \tag{5}$$

where $I_{OE} = G\left(I_x\right)$ is the *Output OE*. Unfortunately, the adversarial loss is sensitive and unstable, and thus some other loss functions are needed.

Smoothness Loss. From the previous studies, the enhanced image has two disadvantages: color distortion and noise. Thus, we get a smoothness loss by combining MSE loss and total variation (TV) loss as follows:

$$\mathcal{L}_{smooth} = \alpha_1 \mathcal{L}_{MSE} + \alpha_2 \mathcal{L}_{TV}, \tag{6}$$

where α_1 and α_2 are coefficients of $\mathcal{L}_{MSE}$ and $\mathcal{L}_{TV}$ respectively.

Because MSE norm penalizes larger errors and is more tolerant for smaller ones, it is mainly used to reduce noise in this work as follows:

$$\mathcal{L}_{MSE} = \frac{1}{N} \sum_{x=1}^{X} \sum_{y=1}^{Y} \left\| I_g\left(x, y\right) - I_{OE}\left(x, y\right) \right\|_2^2, \tag{7}$$

where the combination of x and y is the co-ordinates of image, and $I_g\left(x, y\right)$ is the pixel at (x, y).

In the Aly *et al.*'s work [29], TV loss function acts as the image fidelity term. Bastian *et al.* [31] also used it on denoising. In this paper, TV loss is used to remove noise and artifacts as follows:

$$\mathcal{L}_{TV} = \frac{1}{HWC} \left\| \nabla_x G\left(I_x\right) + \nabla_y G\left(I_x\right) \right\|, \tag{8}$$

where H, W, C denote the dimensions of $G\left(I_x\right)$.

Style Loss. To get better texture, we introduce the pretrained VGG-19 [32] model. Style loss is calculated on the layers of poo11, pool2, pool3 and pool4 in VGG features. In this work, we get the style loss [33] through calculated squared $\mathcal{L}_2$ norm of Gramian, which is the correlation on different locations for each feature to understand the general style of the overall image. The style loss is defined as follows:

$$\mathcal{L}_{style} = \sum_{n=0}^{N-1} \left\| \left(G_l\left(I_{OE}\right)\right) - \left(G_l\left(I_g\right)\right) \right\|_2^2, \tag{9}$$

where G_l is Gramian, which is calculated by the Hermitian matrix of inner products defined as:

$$\mathrm{G}_l\left(F\right) = F_l^T F_l, \tag{10}$$

where F_l is the feature map of layer l.

Total Loss Function. The total loss function $\mathcal{L}_{total}$ combines the above loss functions with different weights as follows:

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{adv} + \lambda_2 \mathcal{L}_{smooth} + \lambda_3 \mathcal{L}_{style}, \tag{11}$$

where the weights λ_1, λ_2 and λ_3 depend on the effects of sub-loss functions on the visual quality.

4 Experiments and Results

4.1 Datasets

We use DPED dataset [1] for training and evaluation, which is provided by PIRM 2018 Enhancement on Smartphones Challenge[1]. It contains three sub-datasets, and the image datasets of smartphones are captured by iPhone, BlackBerry and Sony. The label datasets are all obtained by Canon. In PIRM 2018 Enhancement on Smartphones Challenge, we only use iPhone-Canon sub-dataset for training and evaluation. However, in our experiments we also evaluate the performance of the proposed method on BlackBerry-Canon and Sony-Canon sub-datasets.

4.2 Training Details

The proposed generator MCRN only has 4,947 and 5,394 parameters in testing and training phases respectively. For the training phase, we use a batch size of 50 whose resolution is 100×100. It is trained for 1.8×10^4 iterations from scratch with initialized learning rate as 5×10^{-4} and decreasing by the factor 10 for every 8×10^3 iterations. For every 500 iterations, we make an evaluation and save the model. Adam [34] optimization is used to optimize parameters of generator and discriminator. In training, we set hyper-parameters for the smoothness loss: $\alpha_1 = 1$ and $\alpha_2 = 23$. The coefficients of $\mathcal{L}_{total}$, λ_1, λ_2 and λ_3 are 1, 100 and 30 respectively. The training time is about 4 h on a PC with GPU GTX 1080 Ti, Tensorflow v1.8.0, CUDA v9.0 and cuDNN v7.5.

4.3 Ablation Study

To verify the effectiveness of the proposed method, we do the ablation study considering a single-connected generator, the generator without stage-wise supervision, L_{smooth} and L_{style}. Because we only have a self-evaluated iPhone dataset, we perform the ablation study on it. In addition, the results are shown in Table 1 and Fig. 4.

[1] http://ai-benchmark.com/challenge.html.

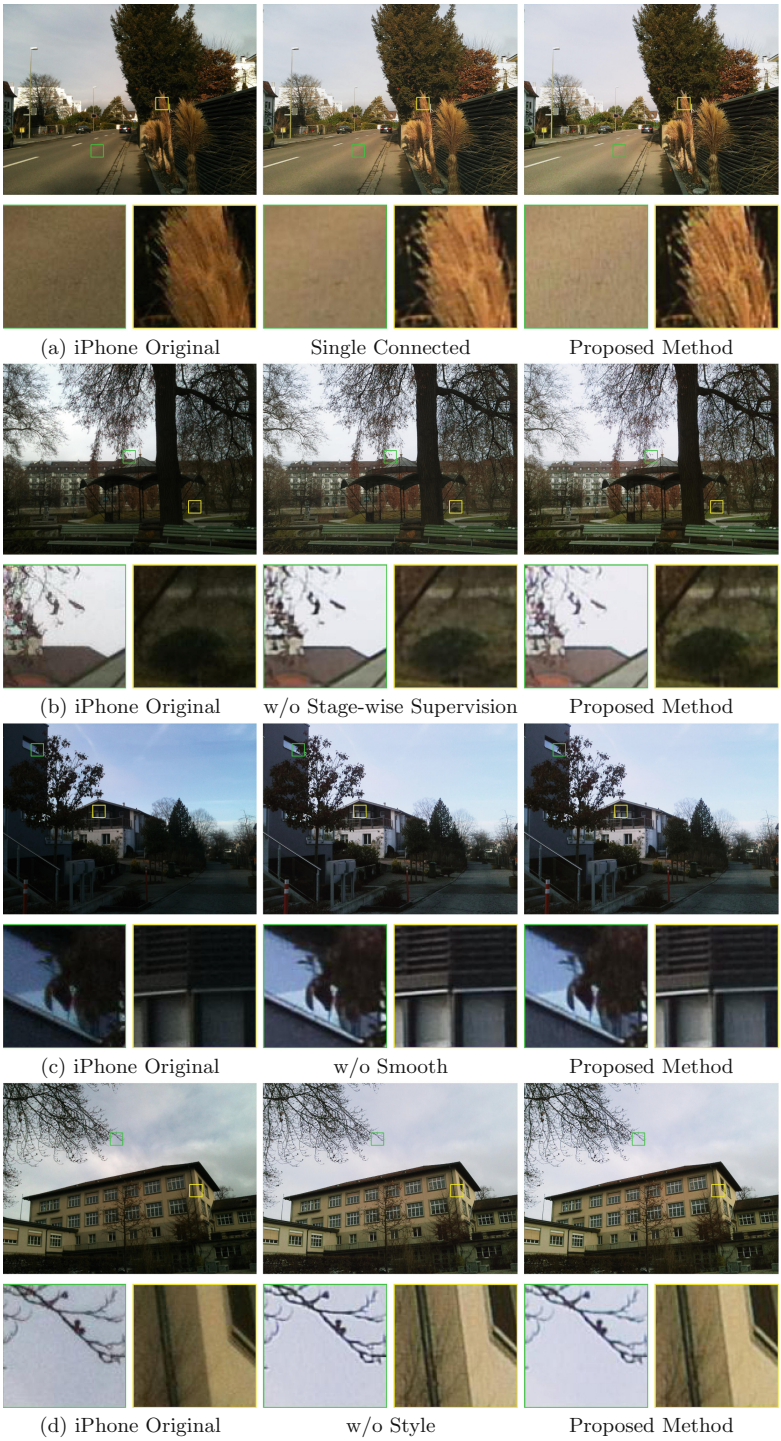

Fig. 4. The results of ablation study. We choose one example in each ablation study.

Single Connected Generator. The single-connected generator is similar to a standard Residual Network [17]. We apply the single-connected generator in the same conditions. In Table 1, the PSNR and MS-SSIM scores are lowest. As shown in Fig. 4(a), it is obvious that color of the proposed method is more natural-looking than that of the single connected generator and keeps texture details in the enlarged local image. As a result, we draw a conclusion that the proposed multiple connected residual generator outperforms the single connected generator.

Without Stage-wise Supervision. In this sub-task, we remove first stage layer and thus the generator only generates *Output OC*. The other hyper-parameters are kept the same as the proposed method. Also, we train it via the same strategy as the previous ablation study. As shown in Table 1, the enhanced images are not good at generating structure information without the first stage. In Fig. 4(b), the structures of tree and roof are clearer than the middle column ones. Thus, we adopt a two-stage generator, and use stage-wise supervision to keep structure detail.

Without Smoothness Loss. We drop out smoothness loss function to train the same network. As shown in Fig. 4(c), the middle column image generated without smoothness loss have serious noise and artifacts. Also as shown in Table 1, the PSNR socore of this experiment is lower than the proposed method, but the MS-SSIM score is similar. Thus, we add smoothness loss to keep image fidelity. Consequently, smoothness loss can improve visual quality while keeping structure.

Table 1. Results of ablation study in terms of PSNR (unit: dB) and MS-SSIM.

Strategy	PSNR	MS-SSIM
Single connected	22.35	0.9167
w/o first stage	22.50	0.9213
w/o smooth loss	22.43	0.9226
w/o style loss	22.45	0.9221
Proposed method	**22.52**	**0.9227**

Without Style Loss. The style loss is used to balance texture and structure, and we investigate the significance of this loss function by this sub-task. As shown in Table 1 and Fig. 4(d), the style loss has a significant impact on keeping image structure, because the MS-SSIM score increases evidently. The branches in third column of Fig. 4 are clearer than those in the middle column.

From the experimental results, it can be concluded that the proposed method is very effective in improving visual quality of the enhanced image. The ablation studies are based on iPhone-Canon dataset.

4.4 Analysis and Limitation

According to this challenge [35], we provide the evaluation results in Table 2, and the test dataset is not publicly available in the challenge. Because PSNR and MS-SSIM scores cannot fully represent image quality, the organizers also recommend MOS score during the test phase. Scores A, B and C are PSNR, MOS and the balance between the speed and performance, respectively.

Table 2. Partial results on Smartphones. SRCNN and DPED are baselines provided by PIRM 2018 Enhancement. The columns of CPU and GPU show the testing time per image (unit: msec/image).

Methods	PSNR	MS-SSIM	MOS	CPU	GPU	Score A	Score B	Score C
SRCNN	21.31	0.8929	2.295	3274	204	3.22	2.29	3.49
DPED	21.38	0.9034	**2.4411**	20462	1517	2.89	4.9	3.32
Ours	**21.79**	**0.9068**	2.4324	**833**	**83**	**12.0**	**12.59**	**14.95**

In Table 2, DPED [1] and SRCNN [14] are baselines. The proposed method achieves a lower MOS score as shown in Fig. 2. However, in the other metrics the proposed method outperforms SRCNN and DPED. Moreover, the proposed method is smaller and faster than SRCNN with only 3 layers. The proposed method outperforms the others in both PSNR and MS-SSIM. In Fig. 5, the enhanced images are good in visual quality, and they get high brightness keeping good textures.

However, there are also some failure cases as shown in Fig. 6. The first row is the input images obtained by iPhone 3GS, while the second row is their enhanced images by the proposed method. The test image is captured in under-exposure condition as shown in middle column of Fig. 6, and the enhanced image also has slightly lower luminance. In the Fig. 6, the proposed method produces over-enhanced results in large homogeneous regions with similar color. Thus, it can be observed that the water and blue sky in Fig. 6 look noisy.

4.5 Training and Testing on Other Smartphones' Datasets

In order to prove the universal validity of the proposed approach, we also apply this method on BlackBerry and Sony images. And the test results on this two datasets achieve good scores as shown in Table 3. We choose the same original images from BlackBerry and Sony, and the visual results are shown in Fig. 7 and Fig. 8, which are for BlackBerry and Sony respectively. From the results, it can be observed that BlackBerry images contain more noise while Sony images include more haze. Moreover, Sony images achieve higher quality with more vivid color than Blackberry images.

Fig. 5. Results by the proposed method. First column: Original iPhone 3GS image. Second column: Output of the first stage *Output OC*. Third column: Enhanced images *Output OE* by the proposed method.

Fig. 6. Failure cases. The first row is the input images obtained by iPhone 3GS, while the second row is their enhanced images by the proposed method.

(a) Original image.

(b) Enhanced image.

(c) Original image.

(d) Enhanced image.

Fig. 7. Image enhancement for BlackBerry

(a) Original image.

(b) Enhanced image.

(c) Original image.

(d) Enhanced image.

Fig. 8. Image enhancement for Sony

Table 3. Training on three smartphones and get the following results. We test the proposed method in DPED self-evaluation dataset and perform comparison in terms of PSNR (unit: dB) and MS-SSIM.

Smartphone	PSNR	MS-SSIM
BlackBerry	22.39	0.9336
iPhone 3GS	22.52	0.9227
Sony	23.86	0.9461

5 Conclusions

In this paper, we propose a generative network named multiple connected residual network (MCRN) for image enhancement on smartphones. MCRN is a lightweight generator to deal with speed and memory limitation. Moreover, the proposed method achieves good performance in image enhancement on smartphones. A two-stage generator is used in MCRN to significantly improve visual quality compared with state-of-the-art methods. In our future work, we will investigate improving the over-enhancement effect of the proposed method by considering human visual perception.

Acknowledgment. This work was supported by the National Natural Science Foundation of China (No. 61271298) and the International S&T Cooperation Program of China (No. 2014DFG12780).

References

1. Ignatov, A., Kobyshev, N., Timofte, R., Vanhoey, K., Van Gool, L.: DSLR-quality photos on mobile devices with deep convolutional networks. In: The IEEE International Conference on Computer Vision (ICCV), pp. 3277–3285 (2017)
2. Goodfellow, I., et al.: Generative adversarial nets. In: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (eds.) Advances in Neural Information Processing Systems 27, pp. 2672–2680. Curran Associates, Inc. (2014)
3. Ignatov, A., Kobyshev, N., Timofte, R., Vanhoey, K., Gool, L.V.: WESPE: weakly supervised photo enhancer for digital cameras. CoRR abs/1709.01118 (2017)
4. Huang, G., Liu, S., van der Maaten, L., Weinberger, K.Q.: CondenseNet: an efficient densenet using learned group convolutions. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2752–2761 (2018)
5. Beghdadi, A., Negrate, A.: Contrast enhancement technique based on local detection of edges. Comput. Vis. Graph. Image Process. **46**(2), 162–174 (1989)
6. Arici, T., Dikbas, S., Altunbasak, Y.: A histogram modification framework and its application for image contrast enhancement. IEEE Trans. Image Process. **18**(9), 1921–1935 (2009)
7. Reza, A.M.: Realization of the contrast limited adaptive histogram equalization (CLAHE) for real-time image enhancement. J. VLSI Signal Process. Syst. Signal Image Video Technol. **38**(1), 35–44 (2004)

8. Wang, C., Ye, Z.: Brightness preserving histogram equalization with maximum entropy. IEEE Trans. Consum. Electron. **51**(4), 1326–1334 (2005)
9. Güçlütürk, Y., Güçlü, U., van Lier, R., van Gerven, M.A.J.: Convolutional sketch inversion. In: Hua, G., Jégou, H. (eds.) ECCV 2016. LNCS, vol. 9913, pp. 810–824. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46604-0_56
10. Gatys, L.A., Ecker, A.S., Bethge, M., Hertzmann, A., Shechtman, E.: Controlling perceptual factors in neural style transfer. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 3985–3993 (2017)
11. Chen, Y.S., Wang, Y.C., Kao, M.H., Chuang, Y.Y.: Deep photo enhancer: unpaired learning for image enhancement from photographs with gans. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 6306–6314 (2018)
12. Ronneberger, O., Fischer, P., Brox, T.: U-Net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) MICCAI 2015. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-24574-4_28
13. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein GAN (2017). arXiv:1701.07875
14. Dong, C., Loy, C.C., He, K., Tang, X.: Learning a deep convolutional network for image super-resolution. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8692, pp. 184–199. Springer, Cham (2014). https://doi.org/10.1007/978-3-319-10593-2_13
15. Ledig, C., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 105–114 (2017)
16. Timofte, R., Gu, S., Wu, J., Gool, L.V., Yang, M.H., et al.: Ntire 2018 challenge on single image super-resolution: methods and results. In: CVPRW, pp. 852–863 (2018)
17. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 770–778, June 2016
18. Huang, G., Liu, Z., Weinberger, K.Q.: Densely connected convolutional networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2261–2269 (2017)
19. Ulyanov, D., Lebedev, V., Vedaldi, A., Lempitsky, V.: Texture networks: feed-forward synthesis of textures and stylized images. In: Proceedings of the 33rd International Conference on International Conference on Machine Learning, ICML 20116, vol. 48, pp. 1349–1357. JMLR.org (2016)
20. Liu, Y., Cohen, M., Uyttendaele, M., Rusinkiewicz, S.: Autostyle: automatic style transfer from image collections to users' images. In: Proceedings of the 25th Eurographics Symposium on Rendering, EGSR 2014, Aire-la-Ville, Switzerland, Switzerland, pp. 21–31. Eurographics Association (2014)
21. Okura, F., Vanhoey, K., Bousseau, A., Efros, A.A., Drettakis, G.: Unifying color and texture transfer for predictive appearance manipulation. In: Proceedings of the 26th Eurographics Symposium on Rendering, EGSR 2015, Aire-la-Ville, Switzerland, Switzerland, pp. 53–63. Eurographics Association (2015)
22. Zhang, R., et al.: Real-time user-guided image colorization with learned deep priors. ACM Trans. Graph. **36**(4), 119:1–119:11 (2017)
23. Monroe, W., Hawkins, R.X.D., Goodman, N.D., Potts, C.: Colors in context: a pragmatic neural model for grounded language understanding. Trans. Assoc. Comput. Linguist. **5**, 325–338 (2017)
24. Solli, M., Lenz, R.: Color semantics for image indexing. In: Conference on Colour in Graphics, Imaging, and Vision, vol. 2010, no. 1 (2010)

25. Liu, M.Y., Breuel, T., Kautz, J.: Unsupervised image-to-image translation networks. In: Guyon, I., et al. (eds.) Advances in Neural Information Processing Systems 30, pp. 700–708. Curran Associates, Inc. (2017)
26. Isola, P., Zhu, J., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 5967–5976, July 2017
27. Wang, X., Gupta, A.: Generative image modeling using style and structure adversarial networks. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9908, pp. 318–335. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46493-0_20
28. Huang, X., Li, Y., Poursaeed, O., Hopcroft, J., Belongie, S.: Stacked generative adversarial networks. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR), July 2017
29. Aly, H.A., Dubois, E.: Image up-sampling using total-variation regularization with a new observation model. IEEE Trans. Image Process. **14**(10), 1647–1659 (2005)
30. Ulyanov, D., Vedaldi, A., Lempitsky, V.: Instance normalization: the missing ingredient for fast stylization. arXiv:1607.08022 (2016)
31. Goldluecke, B., Cremers, D.: An approach to vectorial total variation based on geometric measure theory. In: Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 327–333, July 2010
32. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv:1701.07875 (2014)
33. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9906, pp. 694–711. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46475-6_43
34. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. CoRR abs/1412.6980 (2014)
35. Ignatov, A., Timofte, R., et al.: PIRM challenge on perceptual image enhancement on smartphones: Report. In: European Conference on Computer Vision Workshops (2018)