

Image Denoising with GAN Based Model

Peizhu Gong, Jin Liu* and Shiqi Lv

College of Information Engineering, Shanghai Maritime University, Shanghai, 201306, China

*Corresponding Author: Jin Liu. Email: jinliu@shmtu.edu.cn

Received: 05 August 2020; Accepted: 12 September 2020

Abstract: Image denoising is often used as a preprocessing step in computer vision tasks, which can help improve the accuracy of image processing models. Due to the imperfection of imaging systems, transmission media and recording equipment, digital images are often contaminated with various noises during their formation, which troubles the visual effects and even hinders people's normal recognition. The pollution of noise directly affects the processing of image edge detection, feature extraction, pattern recognition, etc., making it difficult for people to break through the bottleneck by modifying the model. Many traditional filtering methods have shown poor performance since they do not have optimal expression and adaptation for specific images. Meanwhile, deep learning technology opens up new possibilities for image denoising. In this paper, we propose a novel neural network which is based on generative adversarial networks for image denoising. Inspired by U-net, our method employs a novel symmetrical encoder-decoder based generator network. The encoder adopts convolutional neural networks to extract features, while the decoder outputs the noise in the images by deconvolutional neural networks. Specially, shortcuts are added between designated layers, which can preserve image texture details and prevent gradient explosions. Besides, in order to improve the training stability of the model, we add Wasserstein distance in loss function as an optimization. We use the peak signal-to-noise ratio (PSNR) to evaluate our model and we can prove the effectiveness of it with experimental results. When compared to the state-of-the-art approaches, our method presents competitive performance.

Keywords: Image denoising; generative adversarial network; convolutional and deconvolutional neural networks; Wasserstein distance

1 Introduction

Noise often appears in the image as isolated pixels or pixel blocks that causes strong visual effects. Generally, the noise signal is irrelevant to the object to be studied. It appears in the form of useless information, disturbing the observable information of the image, which is a great obstacle to image processing. Therefore, image denoising is a very promising task in graphics, whose purpose is to restore original signal and preserve details from the noisy image.

In recent decades, various image denoising methods, no matter based on traditional methods or machine learning methods, have been proposed. The current traditional algorithms can be roughly divided into two categories, namely spatial noise reduction and frequency noise reduction. Based on the rule that the sum of random noise in the spatial domain is zero, the former method obtains a new center pixel by calculating other adjacent pixels in the gray space within a certain size receptive field, thereby offsetting the image noise. Differently, the latter method first transforms the image from the spatial domain to the frequency domain, and then divides the noise into high, medium and low frequency noise. In this way, the noise of different frequencies can be separated. However, many traditional filtering methods are poor



targeting, not having optimal expression and adaptation for specific images. Moreover, in the process of noise reduction, details or edge information of images are often lost.

Recent studies regarded that deep learning has revolutionized the computer vision and pattern recognition community. Among them, Convolutional Neural Networks (CNN) have been demonstrated that have excellent performance in image processing tasks, especially in the applications of image recognition, object detection, lossy image compression, image super-resolution and image denoising. Besides, Generative Adversarial Networks (GAN) [1] have generally been applied to image processing after it was proposed. Many summaries of training techniques and articles for improving the model itself gradually began to be published in the second half of 2015 and early 2016, which greatly stimulated the subsequent development of GAN, especially in the application of data generation and other related issues.

Inspired by the idea mentioned above, we research GAN and its variants, and then propose a novel model for generating denoised images when it is inputted with noisy images. The model aims to construct generative adversarial networks to learn how to offset the picture noise, which can be divided into a generator and a discriminator. Specially, the architecture of our proposed generator consists of two parts: encoder and decoder [2]. The encoder adopts convolutional neural networks to extract features, while the decoder outputs the noise in the images by deconvolutional neural networks. Shortcuts are added between designated layers which is used in Res-Net [3] for preserving image texture details and preventing gradient explosions. We add Wasserstein distance in loss function as an optimization to improve the training stability of the model. We validate our method using 100 screen-shots and showed that it has competitive performance compared to the state-of-the-art approaches.

The remaining of this article is structured as follows: In Section 2, we introduce the current research situation of image denoising with neural networks. In Section 3, we present the proposed GAN based networks model. Finally, we evaluate the experimental results with peak signal-to-noise ratio (PSNR) and draw a conclusion in Section 4.

2 Related Work

This section briefly describes the existing research in image denoising and related deep learning techniques.

2.1 Image Denoising

One can model the observed noisy image as the superposition of two images - one corresponding to noisy and the other corresponding to the clear background image. Hence, the input image I_n can be expressed as $I_n = I_c + N$, where I_c is the non-noisy signal, and N is the noise signal.

At present, Non-Local Means (NLM) [4] and Block Method of 3-Dimension (BM3D) [5] have high usage rates. The NLM algorithm estimates the position of the center point of the reference block by performing weighted average operations on the self-similar structure blocks, thereby reducing noise. The most serious drawback of NLM is that the details in the image are removed along with the noise. The BM3D takes usage of the similarity between image blocks, achieving higher signal-to-noise ratio and excellent visual effects. The shortcoming of this method is the high time complexity.

Additionally, Gu et al. [6] proposed a new idea. They use the set of similar patches to evaluate whether there is noise in the image. Elad et al. [7] proposed a method named K-SVD. The method removes noise with a sparse dictionary. Liu et al. [8] raised a novel parallel method for denoising and deduplicating mass web documents.

2.2 CNN Based Image Processing

The first complete convolution neural network model is raised by Lecun et al. [9]. The model of multilevel deep learning has the basic construction of CNNs. Szegedy et al. presents a new complex CNN [10]. They propose a new construction named Inception using dense matrix to approximate sparse matrix. In addition, enlightened by network in network [11], they use 1x1 convolution before traditional

convolution to reduce parameters of calculation. Taigman et al. present a CNN method for face alignment and weight non-sharing convolution layers to realize face recognition [12]. Considering that statistics features in different face area are not same. They use local-convolution in high layers instead of traditional convolution.

In recent years, the advances in computer hardware have provided the soil for the development of deep learning. There are plenty of researchers had tried to do image processing using convolutional neural network. One class of methods is used CNN to do denoising on blind image. Lerenhan et al. [13] proposed a data-driven priori denoising method. Nithish et al. [14] proposed a novel CNN-based model. The model consists of a multi-scale feature extraction layer, a l_p regularizer, and a three-step training approach.

2.3 Generative Adversarial Networks

Goodfellow et al. [1] proposed a novel adversarial network for image generation, which has had a profound impact on deep learning. The network provides a new idea for image processing. Scott et al. [15] proposed a novel model built on generative adversarial networks. The author believes that the excellent performance of GANS in image processing can be applied to text modeling. Liu et al. [16] applied multi-scale and multi-class conditions to GANS to generate handwritten characters.

As we all know, at present, the best model for image processing applications in deep learning is CNN. DCGAN [17] is one of the best attempts at combining CNN with GAN, and Yan et al. use DCGAN to do image super-resolution, denoising and deconvolution [18]. However, although DCGAN has an excellent architecture. It still does not solve the problem from the source root for the training stability of GAN. During the DCGAN training process, it is still necessary to carefully balance the G and D.

Different from DCGAN, WGAN [19] has made an important contribution to improving the stability of training of GANs through the loss function, which can get good performance on a fully connected layer. WGAN proposed using Wasserstein Distance as an optimization training method. In order to meet the conditions for using the Wasserstein distance, the network needs to clip weights in a range to force the Lipschitz continuity. WGAN-GP [20] further enhances the stability of GANs.

Related research on denoising and neural networks is continuing. In this paper, we propose a novel image denoising model, which has achieved competitive results.

3 Methodology

In this part, we first introduce the overview of the proposed model, and then describe the structure and loss functions in detail.

3.1 Overview of the Proposed Model

The proposed model consists of a generator network and a discriminator network. Fig. 1 shows the architecture of the model we proposed. The upper part shows the framework of generator, and the lower one shows the framework of discriminator.

The generator network consists of 12 blocks. The second and fourth convolutional block are connected with the 10th and 8th deconvolution block respectively. It means that the intermediate calculation results obtained by the second and fourth convolutional layer are sent to the 10th and 8th deconvolution layer. Finally, the characteristic map obtained by the 12th layer is subtracted from the noise map to obtain an output result. The discriminator network is composed of 6 convolution units, with the picture processed by the generator as input. The discriminator network scores the picture and the score is output.

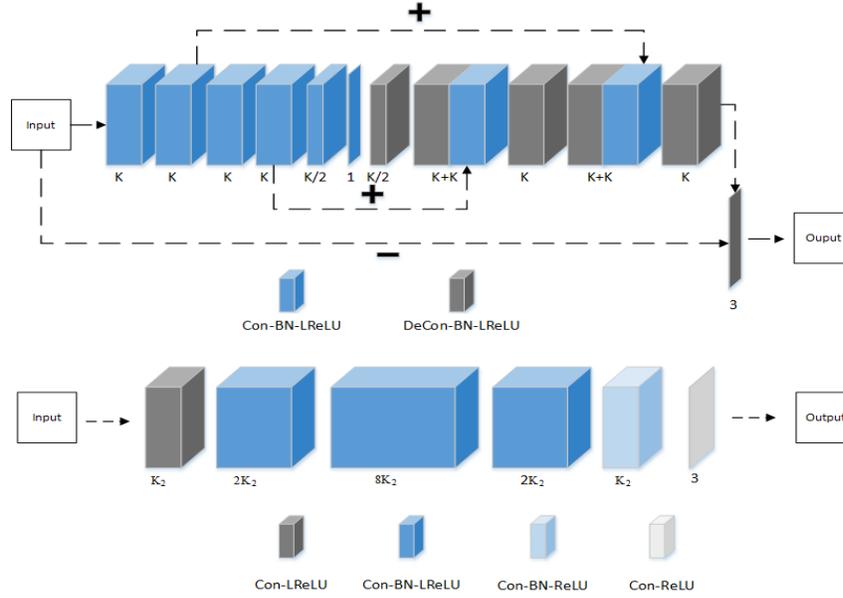


Figure 1: The framework of the network model

3.2 Structure of Generator Network

Inspired by the symmetrical structure of U-net, we propose a novel symmetrical encoder-decoder based generator network. The network consists of 12 blocks, divided into an encoder and a decoder. In figure 1, the encoder consists of 6 blue blocks, while the decoder consists of 6 gray blocks. The input of the generator network are noisy images and the output are denoised images.

The encoder extracts image features in the noisy images, and then the decoder outputs the noise in the images according to the features. Instead of adding fully-connected layer to the encoder and decoder, we use max pooling to reduce the dimensions of feature maps in the encoder. And to repair the dimensions in the decoder, we use upsampling. In order to preserve as much image texture details as possible, shortcuts are added between the encoder and the decoder.

The final objective of the prediction of U-net is to approximate the labeled color map. The loss function contains the pixel-wised-soft-max [7] of the whole image to describe the deviation of each pixel:

$$p_i(x) = \frac{e^{a_i(x)}}{\sum_j^I e^{a_j(x)}} \quad (1)$$

where $a_i(x)$ is the value calculated by activation function in the channel i of the feature map at the pixel position $x \in \Omega$ with $\Omega \subset Z^2$.

According to the deviation, the loss function uses cross entropy to penalize the network:

$$E = \sum_{x \in \Omega} w(x) \log(p_{l(x)}(x)) \quad (2)$$

where $l: \Omega \rightarrow \{1, \dots, K\}$ describes the true label of each pixel and $w: \Omega \rightarrow R$.

3.3 Objective Function

Tradition GAN consists of a generator G and a discriminator D . GAN always uses a mini-max object function to guide the confrontational training between G and D :

$$\min_G \max_D E_{x \sim P_r} [\log(D(x))] + E_{\tilde{x} \sim P_g} [\log(1 - D(\tilde{x}))] \quad (3)$$

where P_g is the distribution of the generated data by G , and P_r is the distribution learned by D . P_g is defined by $\tilde{x} = G(z), z \sim p(z)$. In the original GAN, the input z is some random noise in noisy images.

One of the most prominent problems of GAN is that the training of GAN is more difficult than other neural networks. In order to overcome this shortcoming, Wasserstein GAN (WGAN) is proposed. The most important innovation of WGAN is the modified object function [21]:

$$\min_G \max_D E_{x \sim P_r} [D(x)] - E_{\tilde{x} \sim P_g} [D(\tilde{x})] \quad (4)$$

Besides, Gulrajani et al. [20] further improved WGAN. They proposed to use a gradient penalty algorithm GP to replace the gradient clipping algorithm in WGAN. The improved gradient clipping algorithm reduces the risk of failure of converge or other undesired behaviors of GAN. In order to ensure the reliability and stability when training the model, the GP algorithm is introduced into the object function:

$$L = E_{x \sim P_g} [D(G(x))] - E_{y \sim P_{data}(y)} [D(y)] + \lambda E_{x \sim P_x} [(\|\nabla_x D(x)\|_2 - 1)^2] \quad (5)$$

During the training process, the model is minimizing the value of the above formula, which means that the output of the generator is close to non-noisy images; and the discriminator network expects the target value to be maximized, to predict the authenticity of the input more and more accurately.

3.4 Denoising with GAN

The flow chart of two kinds of GANS are shown in Fig. 2. In our model, G receives a noisy image x , learns the characteristics of noise and depresses the noise. Finally, G generates a picture $G(x)$ without noise. The output $D(x)$ represents the confidence whether x is not generated by G . The higher the $D(x)$ is, the higher the confidence that the picture is a real image. Conversely, the lower the $D(x)$ is, that x is less likely to be a real image.

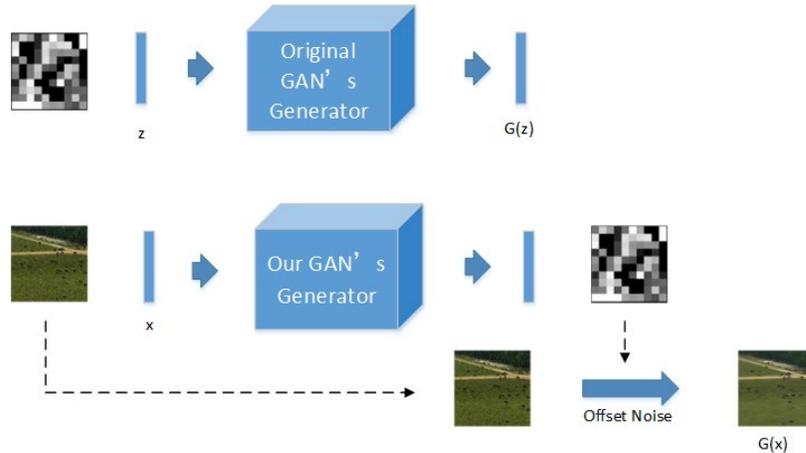


Figure 2: The flow charts of our network and the original GANS

In the training process, the clean images and the denoised images generated by the generator are used as the input of discriminator network, and the discriminant results generated by the two types of pictures are respectively added to the object function. We truncate the generator discriminant result and calculate the overall mean of the truncated matrix to prevent numerical divergence:

$$loss_real = \frac{1}{mn} \sum_{i=1}^n \sum_{j=1}^m a_{ij}, a_{ij} = \begin{cases} a_{ij} = 0, a_{ij} < 0 \\ a_{ij}, 0 < a_{ij} \leq 100 \\ a_{ij} = 100, a_{ij} > 100 \end{cases} \quad (6)$$

$$loss_fake = \frac{1}{mn} \sum_{i=1}^n \sum_{j=1}^m b_{ij}, b_{ij}' = \begin{cases} b_{ij} = 0, b_{ij} < 0 \\ b_{ij}, 0 < b_{ij} \leq 100 \\ b_{ij} = 100, b_{ij} > 100 \end{cases} \quad (7)$$

where $loss_real$ is obtained from clean pictures, $loss_fake$ is obtained from denoised images, a_{ij} and b_{ij} are the elements in the discriminant result matrix, which are the truncated value.

In order to prevent gradient disappearance and gradient explosion, we also add gradient penalty to the loss function of D :

$$M = \mu \times input + (1 - \mu) \times G(x) \quad (8)$$

$$grad = \nabla_M f(A) = \frac{\partial f(A)}{\partial M} \quad (9)$$

$$g\rho = \frac{1}{n} \sum_{i=1}^n \frac{1}{(\sqrt{grad_i^2} - 1)^2} \quad (10)$$

where $G(x)$ is the image generated by G , $input$ is the clean image corresponding to $G(x)$, μ is a parameter in the constructed model, and A is a result obtained by M passing through the discriminator network.

Built on the structure of DCGAN, we replace the G and D network with CNN since convolution layer has an excellent effect on either image feature extraction or image production. Convolutional layers with weight sharing reduce the parameters that are required to be trained, and obtain powerful image feature mapping. In G network, we first process image data through a set of convolutional layers in order to learn image features. Then G network uses a set of deconvolution layers to reconstitution images. In D network, multiply convolution layers acquire knowledge among images generated by G network and those real ones. The generator network can be defined as follows:

$$CBL(K) - CBL(K) - CBL(K) - CBL(K) - CBL(K/2) - CBL(1) - RBL(K/2) - RBL(K) - RBL(K) - RBL(K) - RBL(K) - RBL(3) \quad (11)$$

where $CBL(K)$ refers to a block which consists of a convolutional layer with K kernels, a BN layer and a LReLU layer [22]. In $RBL(K)$, the convolutional layer is replaced with a residual layer.

The discriminator network can be defined as follows:

$$CL(K_2) - CBL(2K_2) - CBL(8K_2) - CBL(2K_2) - CBR(K_2) - CBR(3) \quad (12)$$

where $CL(K_2)$ refers to a block which consists of a convolutional layer with K_2 kernels, and a LReLU layer. $CBR(K_2)$ refers to a block composed of a convolutional layer with K_2 kernels, a BN layer and a ReLU layer.

The LReLU used in the G network can be defined as follows:

$$y_{ji} = \begin{cases} x_{ji}, x_{ji} \geq 0 \\ a_{ji} x_{ji}, x_{ji} < 0 \end{cases} \quad (13)$$

In the D network, apart from use as LReLU, we also use ReLU as activation function:

$$ReLU(x) = \begin{cases} x, x > 0 \\ 0, x \leq 0 \end{cases} \quad (14)$$

4 Experiment Results and Analysis

We use different training sets and validation sets to evaluate the model's noise reduction capabilities in the same experimental environment. In the experiment, we take 100 screen shots from two

documentaries (marine and grassland) as training sets. Then we conduct the same for two training sets: adding Gaussian noise of constant standard deviation. For validation, we use 10 images completely distinct from the training set and add Gaussian noise. For testing, we have both added Gaussian images and real noisy images.

We use peak signal-to-noise ratio (PSNR) to measure the similarities between the outputs of our methods with the original image. Our code utilized TensorFlow library v1.6, and the training process was performed in HP Z840 desktop workstation with NVIDIA K80 GPU. Each task took 12 hours to train the model in 10K iterations. The GAN parameters used in the experiments are the follows: Adversarial Loss Factor is 0.5, Pixel Loss Factor is 1.0, Feature Loss Factor is 1.0 and Smoothness Loss Factor is 0.0001.

Fig. 3 shows the result obtained when we apply the model to images with similar pixel thresholds. From left to right, there are the added Gaussian noise result, the denoising result and the original image. We can see that our model achieves effective results.

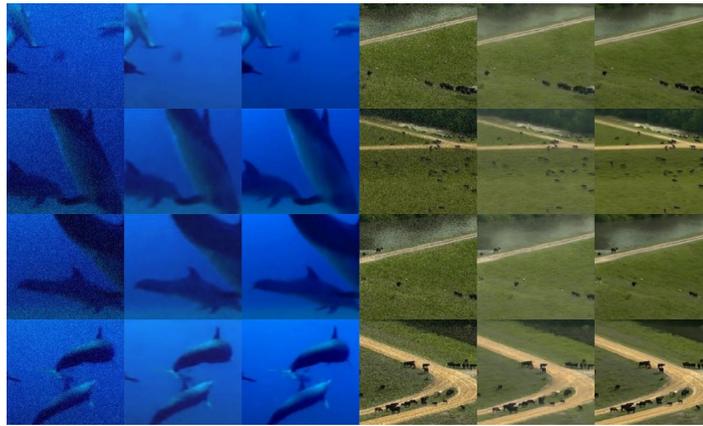


Figure 3: The denoising result of our model on images with similar pixel thresholds

PSNR is calculated from the maximum value of signals and the mean square error between signals to evaluate the denoising performance of the model. It can be expressed as:

$$MSE = \frac{1}{H \times W} \sum_{i=1}^H \sum_{j=1}^W (X(i, j) - Y(i, j))^2 \quad (15)$$

$$PSNR = 10 \log_{10} \left(\frac{(2^n - 1)^2}{MSE} \right) \quad (16)$$

The value of PSNR is proportional to the signal quality. The greater the PSNR value, the less distortion is represented.

Tab. 1 shows the PSNR measurements for the denoising results. We calculate the PSNR value comparison between our proposed model and other denoising models. Our denoising model is better than the traditional GAN denoising effect, but the PSNR value is still lower than the classic filter algorithms.

Table 1: Image denoising result PSNR

Model	PSNR Mean (dB)
Our Model	29.5263
Native DCGAN	27.1454
NLM	32.0913
BD3M	33.6711

We speculate that, the noisy image and the original image are relatively close in pixel distribution. However, in the later stage of training, the discriminator loses the guiding effect, making it difficult to further optimize.

5 Conclusion

In this paper, we propose a novel GAN based model for image denoising. With the novel generator network, the proposed model provides a more efficient and universal image denoising method, and can be easily trained to learn how to offset the picture noise. Experiments' result shows the effectiveness of our proposed model. We believe there is still space to further enhance the experimental results.

At present, color saturation loss that caused by our model is a bit high. At the same time, we are also thinking about whether GAN based noise reduction methods can have wider applications than CNN based methods or traditional methods, which may be the direction of our future research.

Funding Statement: This work is supported by the National Natural Science Foundation of China (61872231, 61701297) and the Major Program of the National Social Science Foundation of China (Grant No. 20 & ZD130).

Conflicts of Interest: We declare that we do not have any commercial or associative interest that represents a conflict of interest in connection with the work submitted.

References

- [1] I. Goodfellow, J. Pouget-Abadie, M. Mirza, Bing Xu, *et al.*, "Generative adversarial nets," *Advances in Neural Information Processing Systems*, vol. 27, pp. 2672–2680, 2014.
- [2] C. Xia, J. Peng, Z. Ma, X. J. J. O. I. H. Li and P. Protection, "A multi-scale network with the encoder-decoder structure for CMR segmentation," *Journal of Information Hiding and Privacy Protection*, vol. 1, no. 3, pp. 109, 2019.
- [3] K. He, X. Zhang, S. Ren and J. Sun, "Deep residual learning for image recognition," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 770–778, 2016.
- [4] A. Buades, B. Coll and J. M. J. I. J. O. C. V. Morel, "Nonlocal image and movie denoising," *International Journal of Computer Vision*, vol. 76, no. 2, pp. 123–139, 2008.
- [5] K. Dabov, A. Foi and V. Katkovnik, "Image denoising by sparse 3-D transform-domain collaborative filtering," *IEEE Transactions on Image Processing*, vol. 16, no. 8, pp. 2080–2095, 2007.
- [6] S. Gu, L. Zhang, W. Zuo and X. Feng, "Weighted nuclear norm minimization with application to image denoising," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 2862–2869, 2014.
- [7] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Transactions on Image Processing*, vol. 15, no. 12, pp. 3736–3745, 2006.
- [8] J. Liu, J. Song, L. Kong, J. Kim and J. Wang, "A novel parallel method for denoising and deduplicating mass web documents," *Journal of Internet Technology*, vol. 17, no. 5, pp. 889–896, 2016.
- [9] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.
- [10] C. Szegedy, W. Liu and Y. Jia, "Going deeper with convolutions," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1–9, 2015.
- [11] M. Lin, Q. Chen and S. Yan, "Network in network," arXiv preprint arXiv:1312.4400, 2003.
- [12] Y. Taigman, M. Yang, M. A. Ranzato and L. Wolf, "Deepface: Closing the gap to human-level performance in face verification," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 1701–1708, 2014.
- [13] L. Li, J. Pan, W. S. Lai, C. Gao, N. Sang *et al.*, "Learning a discriminative prior for blind image deblurring," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pp. 6616–6625, 2018.
- [14] N. Divakar and R. Venkatesh Babu, "Image denoising via CNNs: an adversarial approach," in *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition Workshops*, pp. 80–87, 2017.

- [15] S. Reed, Z. Akata, X. Yan, L. Logeswaran, B. Schiele and H. Lee, “Generative adversarial text to image synthesis,” arXiv preprint arXiv:1605.05396, 2016.
- [16] J. Liu, C. Gu, J. Wang, G. Youn and J. Kim, “Multi-scale multi-class conditional generative adversarial network for handwritten character generation,” *The Journal of Supercomputing*, vol. 75, no. 4, pp. 1922–1940, 2019.
- [17] A. Radford, L. Metz and S. J. Chintala, “Unsupervised representation learning with deep convolutional generative adversarial networks,” arXiv preprint arXiv:1511.06434, 2015.
- [18] Q. Yan and W. Wang, “DCGANs for image super-resolution, denoising and deblurring,” *Advances in Neural Information Processing Systems*, pp. 487–495, 2017.
- [19] M. Arjovsky, S. Chintala and L. Bottou, “Wasserstein gan,” arXiv preprint arXiv:1701.07875, 2017.
- [20] I. Gulrajani, F. Ahmed, M. Arjovsky, V. Dumoulin and A. C. Courville, “Improved training of wasserstein gans,” in *Advances in Neural Information Processing Systems*, pp. 5767–5777, 2017.
- [21] M. Cuturi, “Sinkhorn distances: Lightspeed computation of optimal transport,” *Advances in Neural Information Processing Systems*, vol. 26, pp. 2292–2300, 2013.
- [22] K. He, X. Zhang, S. Ren, and J. Sun, “Delving deep into rectifiers: Surpassing human-level performance on imagenet classification,” in *Proc. of the IEEE Int. Conf. on Computer Vision*, pp. 1026–1034, 2015.