Multimedia Tools and Applications
https://doi.org/10.1007/s11042-019-7556-x

# A generative adversarial network for image denoising

**Yue Zhong**[1,2] **· Lizhuang Liu**[2] **· Dan Zhao**[2] **· Hongyang Li**[2]

Received: 6 December 2018 / Revised: 30 January 2019 / Accepted: 28 March 2019 /
Published online: 02 May 2019


**Abstract**

Recent studies have shown that the performance of image denoising methods can be improved significantly by using deep convolutional neural networks(CNN). The traditional CNN ways mainly focus on minimizing the Mean Squared Error (MSE), resulting in a feeling that the images lack of high-frequency details. So we apply a generative adversarial network (GAN) in image denoising. A very deep convolutional densenet framework is acting as our generator, which benefits in easing the vanishing-gradient problem of very deep networks. Moreover, we use Wasserstein-GAN as our loss function to stabilize the training process. Also, the Wasserstein distance between real and generated images from discriminator can be regarded as an indicator that has been proved highly relevant to the quality of the generated sample. A photo-realistic image with higher quality can be produced through our work than in traditional ways.

**Keywords** Image denoising · Generative adversarial network · Densenet · Wasserstein-GAN

## 1 Introduction

The image denoising is an ill-posed inverse problem, which is the same as single image super-resolution(SR). They are both aimed to recover a clean image from the corresponding input image. We believe the same mathematical model can both work well on SR and

✉ Lizhuang Liu
liulz@sari.ac.cn

Yue Zhong
zhongyue@sari.ac.cn

Dan Zhao
zhaodan@sari.ac.cn

Hongyang Li
lihongyang@sari.ac.cn

[1] Shanghai University, Shanghai 200444, People's Republic of China

[2] Shanghai Advanced Research Institute, Chinese Academy of Sciences, Shanghai 201210, People's Republic of China


**Fig. 1** From left to right: The denoised image almost looks indistinguishable with the neighboring ground truth image. The Gaussian noise with standard deviation $\sigma = 30$ is directly added to the original image to form the input noisy image

image denoising by adjusting the structure to satisfy the different tasks. Recently, significant progress has been achieved in the single image super-resolution(SR) through generative adversarial networks (GANs). A representative work is SRGAN [17], which is able to generate a photo-realistic single image. It is amazing that we find GANs do well in capturing perceptually relevant differences, such as high texture detail. Based on recent studies on single image super-resolution, we design our generative adversarial network – an method using DenseNet [13] built on conditional generative adversarial networks. In addition, we use Wasserstein-GAN [2] with the gradient penalty [10] to accelerate our training, whose loss function optimizes Earth Mover (EM) distance, which can effectively drive the generator to work. It not only enables the training process almost fail-free, but produces the same quality image as the previous GANs do (Fig. 1).

In this work, we propose a Densely Connected Denoising Network(DenseNet) based on a Generative Adversarial Network, which generates a photo-realistic image with high quality. The Densely Connected Denoising Networks(DenseNet) [13], regarded as the state of the art convolutional net, repeatedly using skip connections from input to the reconstruction layer change the exploding/vanishing gradients problems in positive ways. In this case, high-level features at top layers can acquire more high-frequency details from the features at low levels. Relying on a Generalized Adversarial Network (GAN), our work shows outstanding performances and produce convincing images. To prevent training from crashing, we apply the very deep GAN with stable Wasserstein-GAN loss function.

## 2 Related work

Convolutional Neural Networks(CNN) have already been widely used to denoise images [6, 7, 29]. The recent work has recognized image denoising as a regression problem. H. C. Burger et al. [6] propose a Multi-Layer Perceptron neural network, which attempts to learn the mapping from a noisy image to a noise-free image by directly applying in image patches, dividing the image into overlapping patches as continuous input vectors to train the MLP. The noisy patches with the corresponding clean patches are send to MLP for estimating the network parameters by minimizing the difference between the noisy and clean patches. The average of image patches is calculated to produce the denoised image. Nithish Divakar et al. [7] apply a multi-scale feature extraction layer in the GANS, which helps in reducing the effect of noise on feature maps, while using lp regularizer to select the appropriate maps. Another one of the most well-known models for image restoration is stacked denoising auto-encoder [20], which learns end-to-end mappings from corrupted images to the original ones. Jaakko Lehtinen et al. [19] use basic statistical reasoning to map corrupted observations to clean signals by machine learning, which can give a clean image by observing the corrupted examples. Stamatios Lefkimmiatis et al. [18] propose a novel way based on the proposed network architecture: on one hand, convolutional layers is involved as a core component in the first network, on the other hand, the inherent non-local self-similarity property of natural images can be found by takeing advantage of non-local filtering layers in the second network. Many neural networks for image denoising based on CNN and GAN, have achieved great process in the past few years.

Another relevant problem using the convolutional networks that recently has made breakthroughs on is the single image super-resolution. It's a highly ill-posed problem to generate a high resolution (HR) version image from a lower resolution (LR) version image. Recovering the high-frequency details, especially, is a big challenge in single image super-resolution (SR), which is as tricky as image denoising since the high-frequency details are easily smoothed. Many classic convolutional networks are proposed in [8, 17, 26] for single image super-resolution. The input image is downscaled first to the desired resolution from the ground truth image. After series of convolution and last deconvolution operations, a high-resolution output image can be composed. The parameters are all learned during the training by minimizing of the mean squared error (MSE) between the recovered HR image and the ground truth image.

The most relevant works to ours are SRDenseNet [26] and SRGAN [17]. The architecture of work SRDenseNet improves the flow of information through the network, alleviating the gradient vanishing/exploding problem by applying the dense connected convolutional networks [13]. Furthermore, the reuse of feature maps from all previous Densely Blocks has been proved to be more efficient for avoiding the relearning of redundant features. Another work based on GAN is SRGAN. Unlike previously proposed convolutional networks for image super-resolution, it proposed an architecture relying on a very deep residual net architecture(ResNet) [11, 12], which is based on generative adversarial network (GAN) containing Discriminator and Generator network. It allows Generator network to be trained to generate an indistinguishable image from the ground truth image. Then, it fools a differentiable Discriminator network that is trained to distinguish the reconstructed image with the real image. The two networks confront each other and eventually reach the Nash equilibrium. More importantly, the training of the network is operated by minimizing the perceptual loss function, instead of minimizing only the Mean Squared Error (MSE). The proposed perceptual loss function consisting of content loss and adversarial loss put the Generator network forward to produce a photo-realistic image.

## 3 Method

The generative adversarial networks(GANs), proposed by Goodfellow et al. [9], is to train a Generative model G with the purpose of fooling the Discriminator model D. The two competing networks: the former(Generator) is used to create solutions to an indistinguishable image with ground truth image by receiving a noisy image as input while the latter(Discriminator)receives a ground truth image and a generated image and is aimed to distinguish the generated image from ground truth image. The competition between the Generator G and Discriminator D can be expressed as the following objective:

$$\min_G \max_D \mathop{\mathbb{E}}_{x\sim\mathbb{P}_r}[\log(D(x))] + \mathop{\mathbb{E}}_{\tilde{x}\sim\mathbb{P}_g}[\log(1 - D(\tilde{x}))] \tag{1}$$

where $\mathbb{P}_r$ stands for the real data x distribution and $\mathbb{P}_g$ stands for the model distribution, defined by $\tilde{x} = G(z)$. The input noise variables z has distribution P(z). GANs are famous for their ability to generate a photorealistic image with high quality. The training of a GAN, however, suffers from its own challenges, such as vanishing gradients, mode collapse [23] etc. During the period of training, caution is needed to keep a balance between Generator G and Discriminator D for a successful training. Otherwise, the training might end up in failure. Though a lot of efforts have been made to avoid those problems, but the approaches relying on the new architectures like DenseNet don't necessarily work well. Arjovsky et al. [2] observed the failure of GAN training caused by JS divergence approximation. To address this issue, they proposed Wasserstein-GAN, which optimized the Earth-Mover(EM) distance as loss function and in turn made the training fail-free and generated the same quality image as original GAN do. In addition, EM distance between real and generated images from discriminator can be regarded as an indicator that has been proved to be highly relevant to the quality of the generated sample. In our work, WGAN is employed to guide the training.

### 3.1 Generator network

The core part of GAN is Generator Network, which generates the final result. To generate a photo-realistic image with high quality, the Generator Network should gain more detailed information from the neighboring pixels. In this case, the critical section should apply an outstanding architecture with deep convolutional neural networks to generate the high-quality images.

Our structure of Generator Network is relevant to SRDenseNet [26], the architecture of which is shown in Figs. 2 and 3. It contains one convolution block, eight DenseBlocks, one bottleneck block and one output block, with each block consisting of a batch normalization layer [14], a Relu activation [21] and a convolution layer. We feed each layer with all of the previous layers(except the last one) by using skip connections [13], which effectively alleviates gradient vanishing/exploding problem and enhances the feature propagation in

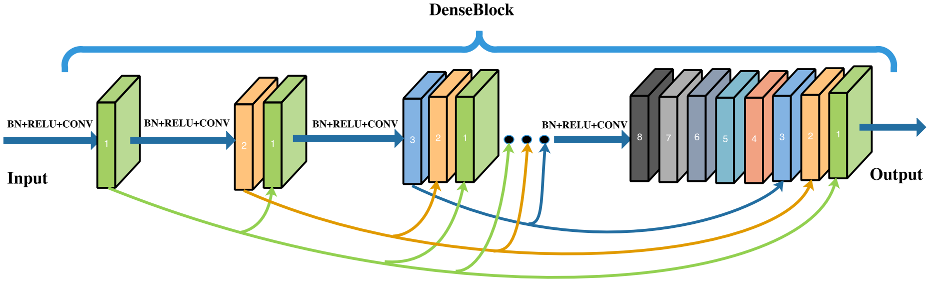



**Fig. 2** Architecture of a DenseBlock, which consists of 8 convolution layers. Each convolution layer receives the previous layer, whose channels is set to 16

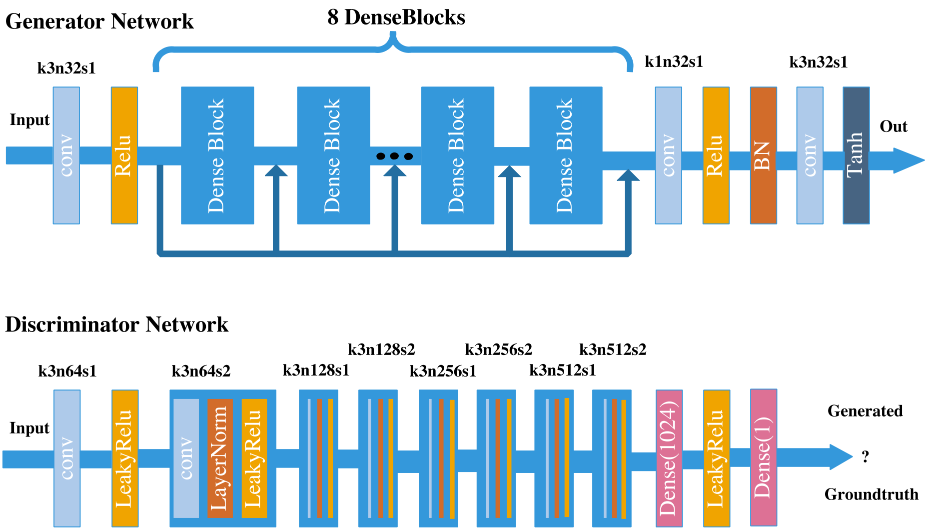



**Fig. 3** The structure of generator and discriminator network contains the kernel size(k), number of feature maps (n)and strides(s). The Generator Network is made of three convolution layers and eight DenseBlocks. The Discriminator Network is the same with SRGAN, except that the BatchNorm is replaced by LayerNorm

deep networks. The first convolution layer extracts low-level features from the input noisy image. Then, eight DenseBlocks are adopted learn the high-level features. After that, a Bottleneck layer, which is indispensable, is added. It has been proved [24] that a $1 \times 1$ convolution layer is very suitable for reducing the number of input feature maps, which makes it possible to get a feature fusion at a small computing cost. The last part is a a $3 \times 3$ convolution layer to construct output images. The Generator Network learns a residual correction between the noisy image and ground truth image, which helps accelerate the training.

### 3.2 Discriminator network

The function of Discriminator Network is to distinguish if input image is real or generated, which helps making the denoised result visually appealing. So the Discriminator Network must make the probability value assigned to actual image data as close as possible to one, while the value of generated samples is close to zero.

The Discriminator Network of ours shares the same nature with that of SRGAN. In accordance with the suggestion given by Radford et al. [22], the LeakyReLU [27] activation ($\alpha = 0.2$)and Layer Normalization [3] are included. But in our work, batchnorm layers are replaced by layernorm layers due to WGAN-GP [10]. It includes eight convolutional layers with $3 \times 3$ kernels. The last two layers are fully connected layers to give a probability of image coming from generator network or ground truth input. Another difference in the last layer is that Sigmod activation is not used because of WGAN-GP. The architecture of the Discriminator Network is shown in Fig. 3.

### 3.3 Loss function

GAN has always faced the following problems and challenges: On one hand, difficulties in training require careful design of the model structure and careful coordination of the training level of Generator Network and Discriminator Network. On the other hand, the loss function of Generator Network and Discriminator Network cannot indicate the training process, lacking a meaningful indicator associated with the quality of the generated image. To make the training stablely work, we adopt the WGAN [2], which only make the following three simple changes, compared to the traditional GAN: removing the last layer of Discriminator Network, taking no log on the loss of Generator Network and Discriminator Network and clipping gradient after updating the parameters of Discriminator Network.

WGAN proposes to use the Wassertein distance as an optimization method to train GAN, but there are still differences between mathematics and real code implementation. Using Wasserstein distance needs to satisfy the strong continuity condition—Lipschitz continuity [2]. In order to satisfy this condition, the author restrains the weights to a certain range to a range for guarantee Lipschitz continuity, but this also creates hidden problems, such as vanishing-gradients or exploding-gradients. Although the theory is proved to be very amazing, yet the results are not as good as expected. So Gulrajani et al. [10] propose to add a gradient penalty term instead. The approach enables the training of GAN to adopt new neural network architectures and requires almost no hyper-parameter tuning, which fits well with our network. The value function of WGAN is defined as follow:

$$\min_{G} \max_{D \in \mathcal{D}} \mathop{\mathbb{E}}_{x \sim \mathbb{P}_r} [D(x)] - \mathop{\mathbb{E}}_{\tilde{x} \sim \mathbb{P}_g} [D(\tilde{x})] \tag{2}$$

where $\mathcal{D}$ is the set of 1-Lipschitz functions. The goal here is making value approximates $K \cdot W(\mathbb{P}_r, \mathbb{P}_\theta)$, where $K$ is a Lipschitz constant and $W(\mathbb{P}_r, \mathbb{P}_\theta)$ is a Wasserstein distance. To make sure gradient of Discriminator Network does not exceed $K$, a gradient penalty term is added:

$$\lambda \mathop{\mathbb{E}}_{\tilde{x} \sim \mathbb{P}_{\tilde{x}}} [(||\nabla_{\tilde{x}} D(\tilde{x})||_2 - 1)^2] \tag{3}$$

Our loss function has two parts: content loss and adversarial loss :

$$\mathcal{L}oss = \lambda \cdot \mathcal{L}oss_{GAN} + \mathcal{L}oss_{Content} \tag{4}$$

where $\lambda$ is a hyperparameter and we define it as 0.0001.

**Content loss** The content loss generally chooses L1 loss(MAE) or L2 loss(MSE). We adopt the L2 loss, which is defined as the difference between the generated image and ground truth image. The pixel-wise MSE loss is calculated as:

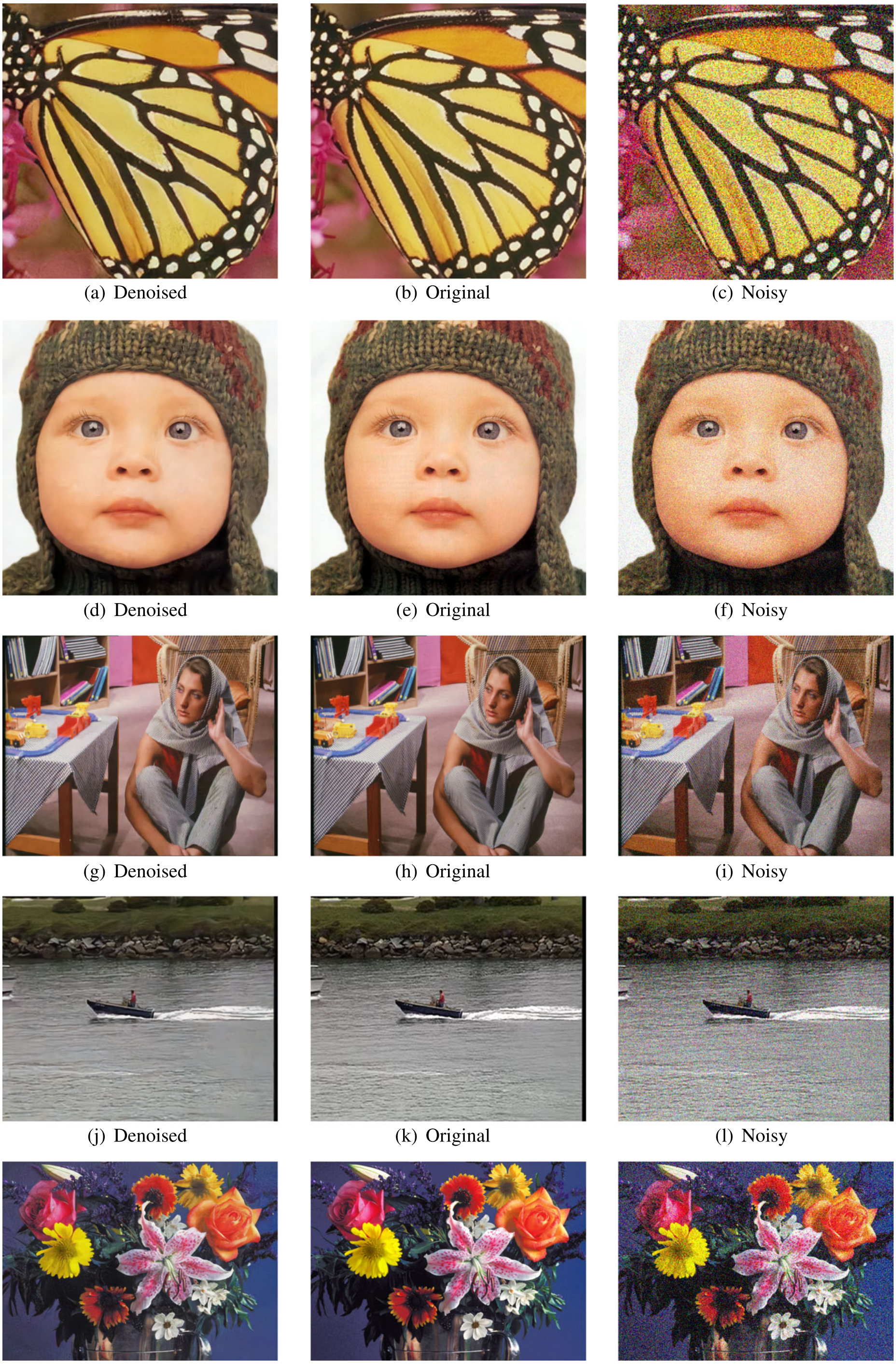
$$\mathcal{L}oss_{Content} = \frac{1}{WH} \sum_{x=1}^{W} \sum_{y=1}^{H} \left( I_{x,y}^{Origin} - G(I^{Input})_{x,y} \right)^2 \tag{5}$$

Here $W$ and $H$ describe the dimensions of the images, $I^{Origin}$ represents the ground truth image and $I^{Input}$ represents the noisy image.

**Adversarial loss** We use WGAN-GP as the critic function, which is found stablely in our work. The additional loss to the image denoising is calculated as the following:

$$\mathcal{L}oss_{GAN} = -D_{WGAN,\theta}\big(G(I^{Input})\big) \tag{6}$$

where $D_{WGAN,\theta}$ is the discriminator's output digit from WGAN-GP for image denoising.



(a) Denoised (b) Original (c) Noisy

(d) Denoised (e) Original (f) Noisy

(g) Denoised (h) Original (i) Noisy

(j) Denoised (k) Original (l) Noisy

(m) Denoised (n) Original (o) Noisy

**Fig. 4** The results of test set of images with different standard deviation

# 4 Experiments

## 4.1 Dataset and data preparation

To demonstrate the generalization of the deep learning model, we used the following database.

**Training data** The training data consists of images from DIV2K [25] - a competition for the single image super-resolution. There are 800 images included. Due to the lack of datasets for training and evaluation of single image denoising, we have randomly selected a $64 \times 64$ crop extracted from the original images for the training. All the pixels are normalized to [-1,1]. During the training process, Gaussian noise with three different levels is added to generate the noisy images. The noisy images form the set of input images and the corresponding original images form the set of ground truth images.

**Test data** We evaluate the model performance on the datasets:Set 5 [4] and Set 14 [28]. These images were used for the test of image super-resolution, which now can simply be fed to our model after the Gaussian noise is added. After that, the denoised images can be generated to complete the training of image denoising (Fig. 4).

## 4.2 Image denoising performance

The peak signal-to-noise ratio (PSNR) is usually calculated for gauging image denoising performance. Table 1 shows PSNR results between the original images and the denoised images, the involving images are corrupted with Gaussian noises with zero mean and standard deviation $\sigma = 30$ for a nosiy input. Comparing our methods with other denoising algorithms on 10 standard test images, we outperform better on 7 test images. Though the

**Table 1** The table gives the standard deviation $\sigma = 30$ is PSNR values of denoised images on 10 test images. The gaussian noise with directly added to form the input nosiy images. None of color space transforms is chosen. Only DnCNN [30] and our method use networks. Bold entries are used to show the best value of several methods convolutional neural

| Image | NLM [5] | KSVD [16] | BM3D [15] | DnCNN [30] | Ours |
|---|---|---|---|---|---|
| baboon | 21.66 dB | 23.04 dB | 23.30 dB | **24.30 dB** | 23.12 dB |
| baby | 30.30 dB | 30.85 dB | 31.13 dB | 27.40 dB | **32.50 dB** |
| barbara | 29.40 dB | 29.08 dB | 29.70 dB | 28.83 dB | **29.74 dB** |
| butterfly | 27.93 dB | 27.98 dB | 26.97 dB | 28.55 dB | **28.85 dB** |
| coastguard | 27.34 dB | 28.42 dB | 27.86 dB | 28.54 dB | **29.54 dB** |
| comic | 25.90 dB | 25.59 dB | 25.04 dB | 26.08 dB | **26.53 dB** |
| lenna | 29.60 dB | 29.61 dB | 29.32 dB | **29.98 dB** | 28.74 dB |
| pepper | 29.23 dB | 29.72 dB | **29.79 dB** | 28.81 dB | 26.74 dB |
| woman | 29.59 dB | 30.73 dB | 29.51 dB | 29.93 dB | **31.12 dB** |
| zebra | 27.26 dB | 28.70 dB | 27.58 dB | 22.63 dB | **29.10 dB** |

(a) NLM (b) KSVD (c) BM3D (d) DnCNN (e) Ours

**Fig. 5** The denoised images from left to right are produced in the method of NLM, KSVD, BM3D, DnCNN and ours. Our picture has a more realistic vision than other ways do

remaining images have the larger PSNR value, our results show the higher visual quality, which you can find from the Fig. 5: Comparing each hat in the follow pictures, more details can be found in our result picture.

At the initial phase of adversarial training, the ground truth images and denoised images can be distinguished easily by Discriminator D, which show the higher adversarial loss value. At this time, one part of loss function(adversarial loss) will influence Generator G to produce natural looking images, while we find the adversarial loss value has a slow decline trend. It's the reason that contributes our work based on GAN to producing a photo-realistic picture.

### 4.3 Training details

Our models are implemented in Tensorflow [1] deep learning framework and trained on a Nvidia GTX 1080-TI GPU. Our DenseBlock is similar to SRDenseNet, where all convolutional layers had filters with size $3 \times 3$, growth rate k=16. To get a better optimization, we abide the rules of [26] and run five gradient descent steps on Discriminator network, then one step on Generator network. Adam optimizer with a learning rate $10^{-4}$ is used in both Generator and Discriminator network. The same hyper-parameters are used according to WGAN-GP. The batch size is 4 and number of training iterations is 10k. After each 2k iterations, the learning rate decays linearly by 0.1. It takes 6 days to train our model.

## 5 Discussion and future work

In this work, we proposed a novel method for image denoising, which adopt a generative adversarial network. The simple architecture gives very competitive denoising results by applying DenseBlock in the Generator Network. The competition between the Generator Network and the Discriminator Network make the generator network produce images that look more photo-realistic. The processed images preserve clear edges with less high-frequency details blurriness. Our network is able to process different types of noises besides the Gaussian noise. It is grounded solely on the training data we use. Though our result is inspiring, but the model can't perform well against unknown real noises. This may due to the lack of ground truth images with the corresponding real noisy images in our training data. So real noisy images and ground truth images will be considered to obtain as the training data in our future work.

**Algorithm 1** Steps for training of image denoising, use WGAN with gradient penalty, $X$ is a batch size of input images in the training data.

**procedure** DENOISING($X$)
  $M \leftarrow$ *number of iterations*
  **for** $t < M$ **do**
    Sample $(X)$ from real data
    $\tilde{x} = addGaussianNoise(X)$
    Sample $y = G(\tilde{x})$
    Train Discriminator Network to make $X$ as positive samples and y as negative samples, add gradient penalty to Discriminator Network
    Compute reward r using Discriminator Network to update Generator Network, train Generator Network
    Update loss fuction according to equation (4)
  **end for**
**end procedure**

**Acknowledgements** This work was supported by the External Cooperation Program of BIC, Chinese Academy of Sciences, Grant No. 184131KYS820150003.

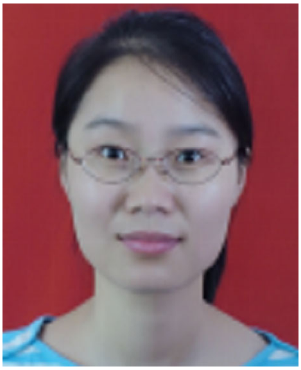
## References

1. Abadi M, Barham P, Chen J et al (2016) TensorFlow: a system for large-scale machine learning. OSDI 16:265–283
2. Arjovsky M, Chintala S, Bottou L (2017) Wasserstein GAN. arXiv:1701.07875
3. Ba JL, Kiros JR, Hinton GE (2016) Layer normalization. arXiv:1607.06450
4. Bevilacqua M, Roumy A, Guillemot C et al (2012) Low-complexity single-image super-resolution based on nonnegative neighbor embedding. In: British machine vision conference (BMVC)
5. Buades A, Coll B, Morel JM (2011) Non-local means denoising. Image Processing On Line 1:208–212

6. Burger HC, Schuler CJ, Harmeling S (2012) Image denoising: can plain neural networks compete with BM3D? In: 2012 IEEE conference on computer vision and pattern recognition (CVPR). IEEE, pp 2392–2399
7. Divakar N, Babu RV (2017) Image denoising via CNNs: an adversarial approach. In: 2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW). IEEE, pp 1076–1083
8. Dong C, Loy CC, He K et al (2016) Image super-resolution using deep convolutional networks. IEEE Trans Pattern Anal Mach Intell 38(2):295–307
9. Goodfellow I, Pouget-Abadie J, Mirza M et al (2017) Generative adversarial nets. In: Advances in neural information processing systems, pp 2672–2680
10. Gulrajani I, Ahmed F, Arjovsky M et al (2017) Improved training of Wasserstein GANs. In: Advances in neural information processing systems, pp 5769–5779
11. He K, Zhang X, Ren S et al (2016) Deep residual learning for image recognition. In: IEEE conference on computer vision and pattern recognition (CVPR), pp 770–778
12. He K, Zhang X, Ren S et al (2016) Identity mappings in deep residual networks. In: European conference on computer vision (ECCV), pp 630–645
13. Huang G, Liu Z, Weinberger KQ et al (2017) Densely connected convolutional networks. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4700–4708
14. Ioffe S, Szegedy C (2016) Batch normalization: accelerating deep network training by reducing internal covariate shift. In: International conference on machine learning, pp 448–456
15. Lebrun M (2012) An analysis and implementation of the BM3D image denoising method. Image Processing On Line 2:175–213
16. Lebrun M, Leclaire A (2012) An implementation and detailed analysis of the K-SVD image denoising algorithm. Image Processing On Line 2:96–133
17. Ledig C, Theis L, Huszar F et al (2017) Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 4681–4690
18. Lefkimmiatis S (2018) Universal denoising networks: a novel CNN architecture for image denoising. In: Proceedings of the IEEE conference on computer vision and pattern recognition, pp 3204–3213
19. Lehtinen J, Munkberg J, Hasselgren J et al (2018) Noise2Noise: learning image restoration without clean data. arXiv:1803.04189
20. Mao X, Shen C, Yang YB (2016) Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In: Advances in neural information processing systems, pp 2802–2810
21. Nair V, Hinton GE (2010) Rectified linear units improve restricted boltzmann machines. In: Proceedings of the 27th international conference on machine learning (ICML-10), pp 807–814
22. Radford A, Metz L, Chintala S (2015) Unsupervised representation learning with deep convolutional generative adversarial networks. arXiv:1511.06434
23. Salimans T, Goodfellow I, Zaremba W et al (2016) Improved techniques for training GANs. In: Advances in neural information processing systems, pp 2234–2242
24. Szegedy C, Vanhoucke V, Ioffe S et al (2016) Rethinking the inception architecture for computer vision. In: IEEE conference on computer vision and pattern recognition, pp 2818–2826
25. Timofte R, Agustsson E, Van Gool L et al (2017) Ntire 2017 challenge on single image super-resolution: Methods and results. In: 2017 IEEE conference on computer vision and pattern recognition workshops (CVPRW). IEEE, pp 1110–1121
26. Tong T, Li G, Liu X et al (2017) Image super-resolution using dense skip connections. In: IEEE international conference on computer vision (ICCV). IEEE, pp 4809–4817
27. Xu B, Wang N, Chen T et al (2015) Empirical evaluation of rectified activations in convolutional network. arXiv:1505.00853
28. Zeyde R, Elad M, Protter M (2010) On single image scale-up using sparse-representations. In: International conference on curves and surfaces. Springer, Berlin, pp 711–730
29. Zhang K, Zuo W, Chen Y et al (2017) Beyond a gaussian denoiser: Residual learning of deep CNN for image denoising. IEEE Trans Image Process 26(7):3142–3155
30. Zhang K, Zuo W, Chen Y et al (2017) Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising. IEEE Trans on Image Process

**Yue Zhong** received his B.S. degree in Electronic and Information Engineering from Xidian University, Xi'an China, in 2015, and currently pursuing the M.S. degree in Computer Science and Technology from ShangHai University. His research interests include image processing, deep learning, etc.

**Lizhuang Liu** received the Ph.D. degree in biomedical engineering from Xi'an JiaoTong University, Xi'an, China, in 2003, respectively. He is currently a professor of Shanghai Advanced Research Institute, Chinese Academy of Sciences. His research interests are computer vision, real-time computing, multiple-sensors data and image processing, and smart system application.



**Dan Zhao** received her M.S. degree in Power electronics and power transmission from WuHan University, WuHan, China, in 2006. She is currently a professor of Shanghai Advanced Research Institute, Chinese Academy of Sciences. Her research interests include computer vision, multiple-sensors data and image processing, etc.

**Hongyang Li** received her B.S. degree in Electronic and Information Engineering from Northeastern University(NEU), Qinhuangdao, China, in 2017, and currently pursuing the M.S. degree in Electronics and Communication Engineering, Shanghai Advanced Research Institute, Chinese Academy of Sciences. Her research interests include image processing, machine learning, etc.