Journal of Electronic Research and Application

Research Article

A New Image Denoising Method with Gan Models

Xueji Huang

Southwest petroleum university, Xindu road no. 8, Xindu district, Chengdu 610500, Sichuan province, China

Abstract: In order to obtain clear images and solve the problems of low image quality caused by noise disturbance, a lot of researches have been done on image denoising techniques. In the theoretical system of algorithms studied so far, many algorithms can effectively remove noise in low-dimensional images, but at the same time, the results are slightly inferior when processing high-dimensional images. This paper proposes a q-GAN, which uses multi-scale in generating networks. The convolution kernel extracts image features and transforms the denoising problem into the feature domain. In the feature domain, a residual structure is used to denoise, and the noise distribution is removed from the feature distribution. There are residual noise features in the obtained denoising features, which are removed by subsequent feature filtering of the network structure, and finally a denoised image is generated by fusing the noiseless features.

Keywords: Image Denoising; GANs; Neural Networks

Publication date: March, 2020

Publication online: 31 March, 2020

****Corresponding author:*** Xueji Huang, kuseet123@gmail.com

1 Introduction

Ordinary artificial neural networks have been widely used in regression problems. It minimizes the optimization function to map a continuous input vector to another continuous output vector. A multilayer perceptron neural network was proposed to solve the regression problem of image denoising[1]. Use paired noisy and raw data to estimate network parameters to minimize the difference between noisy and raw data. Compare the output value with the real value. The network parameters are updated using backpropagation and a loss function to minimize the mean square error. Jain and Seung proposed an unsupervised learning method using convolutional networks for image denoising. In contrast to the typical structure of a convolutional network, this network recovers a denoised image from a Gaussian noise model [2]. The network is trained using a Berkeley segmentation dataset that contains noise-free images, so unsupervised learning is used to complete the training process. During training, noise with different changes is added to the input, noisy training samples are synthesized from the noise-free image, and then the network is trained. The loss function is the reconstruction error of the noise samples the most.

This paper proposes an enhanced generative adversarial network (q-GAN), which consists of an identification network and a generative network. The generation network is trained to generate an image from the input data that is difficult to distinguish from the original data. Its goal is to deceive the authentication network of the reconstructed image, and the goal of the authentication network is to distinguish the reconstructed image from the real image. A new loss function is proposed. The loss function of the network consists of a weighted sum of multiple loss functions.

2 Model

A GAN network based on residual structure was established for image denoising. It generates noise-free images by generating networks to achieve image denoising, and at the same time, the decision network that is against it improves the noise-free images generated, and the residual structure helps to train deeper networks. The residual structure uses skip connections and batch normalization, which can train deeper convolutional neural networks in a short time while ensuring training quality. Only three residual blocks are used in this paper. More residual blocks will significantly improve the training accuracy and increase the training time.

2.1 Generator

Since the image of the input generation network is a noise image, the multi-scale features output by the feature extraction layer are also features polluted by noise. For these noisy features, this paper uses stacked convolution layer structure to extract the noise in the features, and through cross-layer connection, subtracts the noisy features from the extracted noise to obtain the denoised image features for subsequent use. Network processing. In image denoising networks, it is usually necessary to extract texture features and recover the denoised images through the features. However, in DnCNN, a deep convolutional network is used to extract the noise distribution in the image, and the cross-layer connection is used to remove the noise distribution in the noisy image to achieve the purpose of image denoising. For generating network G, it is necessary to constantly deceive and discriminate network D:

$$\max\left[\log\left(D\left(G(z)\right)\right)\right] \quad (1)$$

To discriminate network D, we must continuously learn to prevent being deceived by the generated network G:

$$\max[\log] \mathbf{D}(x) + \log \quad (2)$$

Training using gradient descent (GD), for generating gradients of network G: (3)

For the gradient of discriminative network D: (4)

The probability density distribution function of X is defined as Pdata (x), and the probability density distribution function of G (Z) is defined as Pg (x). Then for each training, if G is fixed, the optimal output D value is as follows:

$$D_G \mathbf{YxY} = \frac{Pdata(x)}{Pdata(x) + Pg(x)} \quad (5)$$

The network uses a symmetric structure, similar to the traditional CNN framework, with three convolutional layers, as well as batch normalization and Lrelu[3] activation functions. Then there are three residual blocks, each containing two convolutional layers, batch normalization and Lrelu activation function. The residual structure makes the network effective in training and has better convergence performance. Next are three deconvolution layers, each corresponding to a convolutional layer at the front end of the network. Each deconvolution layer resizes the image. The generator network is shown in Figure 1.

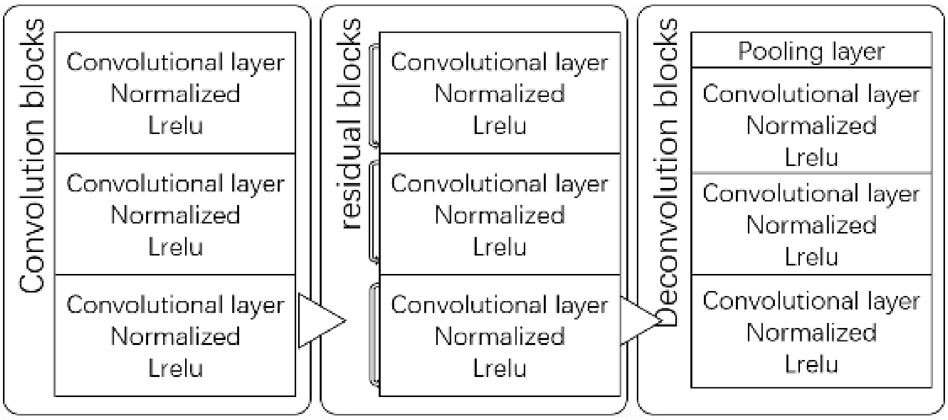

Figure 1. Generator model

2.2 Discriminator

This paper uses a convolutional neural network to build a discriminative network. The image information is extracted through four layers of convolutional layers. Each layer of convolutional layers uses batch normalization processing, uses ReLU as the activation function, and uses a maximum pooling operation. The discriminative network extracts image features through a convolution operation and uses maximum pooling to reduce the dimensionality of the convolution information. When selecting the pooling function of the network, the average pooling function averages the features, which can well retain the background information of the image, but cause loss of image texture details. In the task of discriminating the network, the judgment is mainly based on the texture details, so the maximum value pooling is selected for the downsampling operation to retain more texture features, while achieving the goal of reducing the amount of parameters. The batch normalization operation is used to solve the initialization and optimization problems of the network, so that during the training process of the network at each layer, the input distribution does not fluctuate with the change of the previous layer. In the selection of the activation function, the ReLU function is selected because it has a linear unsaturated nature, is easy to calculate, and provides the ability of neural networks to sparsely express. The structure of Discriminator train model is shown in Figure 2.

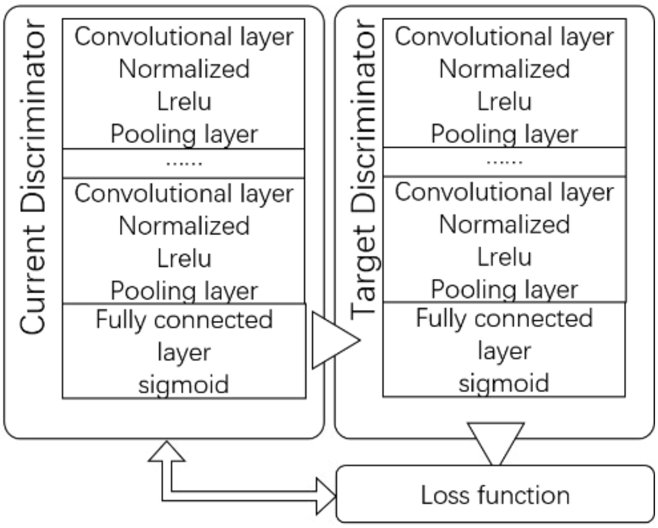

Figure 2. Discriminator train model

In order to improve the quality of the denoised images generated by the network, an identification network is also needed to determine whether each denoised image produced is qualified or unqualified. Construct a decision network based on deep reinforcement learning, and build two neural networks. The target decision network is a decision network that generates an adversarial network. It does not update parameters in a timely manner. The current decision network is used to determine whether the denoised image is qualified. This neural network has the latest neural network parameters.

2.3 Comprehensive loss function

In order to ensure the good performance of the trained network, a new comprehensive loss function is proposed in this paper. Specifically, the point-wise pixel loss, smoothing loss, and adversarial loss are combined with appropriate weights to form a new comprehensive loss function.

Adversarial loss refers to generating the output produced by the network to deceive the errors caused by the identification network. Pixel loss refers to the error generated by comparing each pixel of the generated image with the ground truth image (Euclidean distance). Adding a new loss to these existing loss functions is called smoothing loss. In order to prevent the possible checkerboard effect of the image[4], the difference between adjacent pixels is calculated as a loss of smoothness. Slide a copy of the generated image one unit to the left, and then take a Euclidean distance between the moving images. The new loss function is defined as follows:

$$L = \lambda a La + \lambda p Lp + \lambda s Ls \quad (6)$$

Among them, LA stands for adversarial loss (loss from discriminator D), LP is pixel loss (pixel-to-pixel Euclidean distance between the generated image and the real image), and Ls is smoothness loss. λa, λp, and λs are predefined weights for adversarial loss, pixel loss, and smoothing loss, respectively.

3 Experimental results and analysis

3.1 Experimental data set and training details

Due to the lack of large-scale data sets that can be used for training and evaluation of single image denoising, this chapter synthesizes a new set of training and test samples in experiments. This paper intercepts 100 movie image frames and adds Gaussian noise as the data set. Gaussian noise with different standard deviations is added to generate different training and test sets, and each image has 5 kinds of intensity noise. The training set contains a total of 500 images and the test set has 50 images. All training and test samples are adjusted to 256 × 144.

In the course of the experiment, because the network is more complicated, directly training the entire network cannot effectively converge to good results, so the two-step training method is adopted for the GAN network proposed in this paper. First, pre-train the generated network to extract global information, then fix the network parameters of this branch and continue training the remaining other branches. The optimization algorithm adopted by the network is Adam (Adaptive Moment Estimation) algorithm, the batch size is 7, and the number of training iterations is 10k. In training, set $\lambda a = 0.5$, $\lambda p = 1.0$, $\lambda s = 0.0001$. The first convolutional and deconvolutional layer of the generated network consists of kernels of size 3 * 3, with a step size of 1. All other convolution and deconvolution layers in the generative network consist of kernels of size

3X3, with a step size of 1. The first three layers of all the convolution and deconvolution layers of the identification network are composed of kernels of size 4 × 4, with a step size of 2 and no zero padding. The last two layers consist of 4 × 4 kernels with a step size of 1 and zero padding.

3.2 Comparison with other methodsy

This section compares the algorithms proposed in this chapter with other algorithms, including traditional algorithms, such as the BM3D[5] denoising algorithm, and deep learning-based algorithms, such as the DnCNN algorithm[6]. For fair comparison, algorithms based on convolutional neural networks, such as SRCNN[7], are trained on the training data set in this paper. In the following denoising simulation, Gaussian noise is added to the initial image, and then the denoising simulation is performed. In the comparison experiment, DnCNN, SRCNN, and BM3D denoising algorithms are used to process the same noisy image to better illustrate the denoising effect of the algorithm in this chapter. Figures 3 and 4 show the denoising results of various methods.

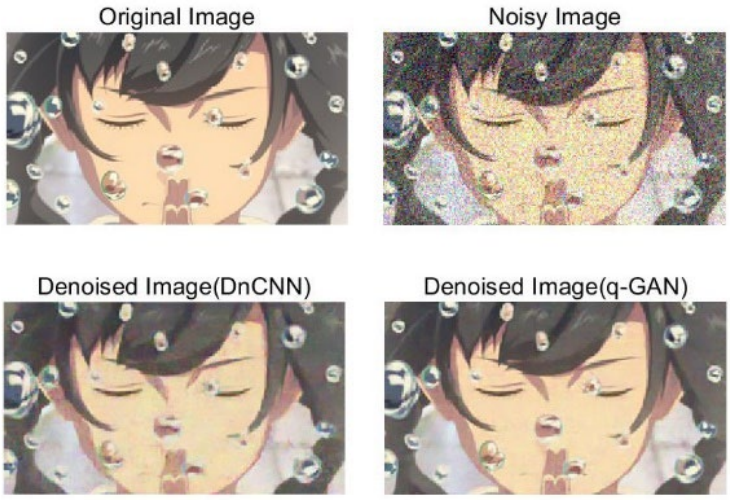

Figure 3. Denoising effect comparison between q-GAN network and other methods

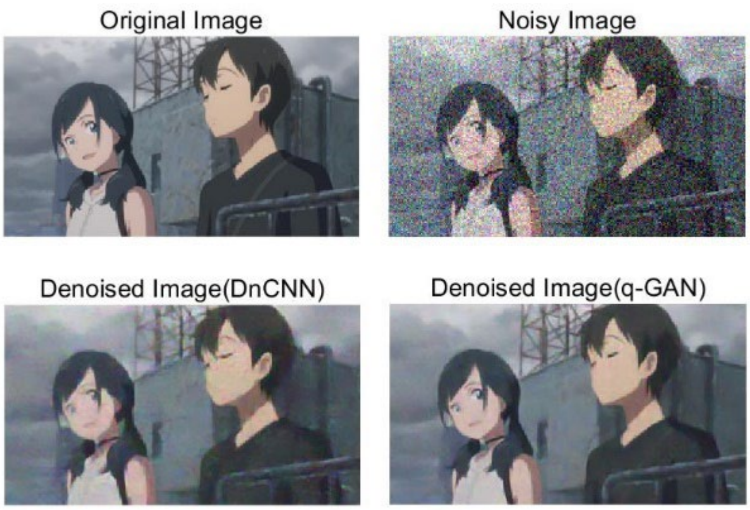

Figure 4. Denoising effect comparison between q-GAN network and other methods

In order to objectively verify the effectiveness of the algorithm in this chapter, the quantitative comparison of PSNR and SSIM values is evaluated below. As shown in Table 1, the underline with the highest PSNR and SSIM is marked. Compared with the deep learning algorithm represented by DnCNN, the results of q-GAN proposed in this chapter have higher PSNR and SSIM.

Table 1. PSNR / SSIM comparison table between q-GAN and other methods

Test Image	Noisy Image		DnCNN		q-GAN	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
1	18.8479	0.7067	30.0871	0.8489	31.0321	0.8489
2	19.7487	0.7221	31.9325	0.8332	32.1271	0.8334

4 Conclusion

This paper details the rationality of generating adversarial networks for denoising. The generation network uses multi-scale feature extraction, and denoising and fusion in the image feature domain can better restore the image texture. The structure of discriminant network and the advantages of adversarial training are explained. The loss function of the generative adversarial network is given. The similarity between the denoised image and the noiseless image is used to measure the distance between the two image distributions to guide the training of the generator network. It also describes the details that need to be paid attention to during training. Through adversarial training, more detailed denoising results such as texture features and edge information are generated. For different intensity noise images, the network parameters obtained by training are different. For input noise images, appropriate parameters need to be selected according to the noise intensity to make the denoising effect better.

References

[1] Burger H C, Schuler C J, Harmeling S. Image denoising: Can plain neural networks compete with BM3D?[C]. 2012 IEEE Conference on Computer Vision and Pattern Recognition, 2012: 2392-2399.

[2] Yang M, Li G, Shu C, et al. GAN based sample simulation for SEM-image super resolution[C]. 2nd Chinese Conference on Computer Vision, CCCV 2017, October 11, 2017 - October 14, 2017, 2017: 383-393.

[3] Dong C, Loy C C, He K, et al. Image Super-Resolution Using Deep Convolutional Networks[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2016, 38(2): 295-307.

[4] Jain V, Seung H S. Natural image denoising with convolutional networks[M]. 2008: 769-776.

[5] He K, Zhang X, Ren S, et al. Deep Residual Learning for Image Recognition[C]. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016: 770-778.

[6] He K, Zhang X, Ren S, et al. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification[C]. 2015 IEEE International Conference on Computer Vision (ICCV), 2015: 1026-1034.

[7] Gan Y, Angelini E, Laine A, et al. BM3D-based ultrasound image denoising via brushlet thresholding[C]. 12th IEEE International Symposium on Biomedical Imaging, ISBI 2015, April 16, 2015 - April 19, 2015, 2015: 667-670.