Inception Model of Convolutional Auto-encoder for Image Denoising

Diangang Wang[1], Wei Gan[1], Chenyang Yan[2(✉)], Kun Huang[1], and Hongyi Wu[3]

[1] State Grid Sichuan Information and Communication Company, Shanghai 610041, China
[2] School of Computer Science and Technology, Shanghai University of Electric Power, Shanghai 200090, China
857322130@qq.com
[3] School of Information and Electronic Engineering, Zhejiang University of Science and Technology, Zhejiang 310023, China

Abstract. In order to remove the Gaussian noise in the image more effectively, a convolutional auto-encoder image denoising model combined with the perception module is proposed. The model takes the whole image as input and output, uses the concept module to denoise the input noise image, uses the improved concept deconvolution module to restore the denoised image, and improves the denoising ability of the model. At the same time, the batch normalization (BN) layer and the random deactivation layer (Dropout) are introduced into the model to effectively solve the model over fitting problem, and the ReLu function is introduced to avoid the model gradient disappearing and accelerate the network training. The experimental results show that the improved convolution neural network model has higher peak signal-to-noise ratio and structure similarity, better denoising ability, better visual effect and better robustness than the deep convolution neural network model.

Keywords: Convolutional auto-encoder · Inception module · Image denoising · Peak signal to noise ratio · Structural similarity

1 Introduction

With the rapid development of computer technology and Internet technology, people's daily life is full of all kinds of information. According to the investigation and research, among all the external information obtained by human beings, vision system accounts for more than 70% [1], so the acquisition, processing and use of image information is particularly important. Image denoising is an important research topic in the field of image processing. While removing the noise, we try to keep the important information in the image. Digital image processing can be generally divided into space-based processing and transform based processing [2]. The denoising method based on the spatial domain is to operate on the gray space of the original image, and process the gray value of the pixel directly. The common methods include mean filter, median filter and image

J. Liu et al. (Eds.): MobiCASE 2020, LNICST 341, pp. 174–186, 2020.
https://doi.org/10.1007/978-3-030-64214-3_12

denoising based on partial differential. Median filter can effectively filter salt and pepper noise and mean filter is suitable for filtering Gaussian noise. The denoising method based on transform domain is to transform the source image first, such as Fourier transform, wavelet transform, etc. Subsequent paragraphs, however, are indented.

At present, many image denoising methods have been proposed by scholars at home and abroad. At present, the BM3D (block matching and 3D) [3] algorithm with better denoising effect is to divide the image into blocks of certain size, merge the blocks with similar characteristics into three-dimensional arrays, process the three-dimensional arrays by three-dimensional filtering method, and obtain the denoised image by inverse transformation; Schuler [4] and others put forward MLP (multilayer perceptron) model, which uses image preprocessing and multilayer perceptron Through the combination of network learning model. The algorithm proposed by Burger [2] uses MLP in image denoising. Chen et al. [5] Proposed TNRD (traditional nonlinear reaction diffusion) model, expanded sparse coding and iterative methods into forward feedback network, and achieved good image denoising effect.

In recent years, research shows that as a typical representative of deep learning, Auto-encoder (AE) is mainly used to learn the compression and distributed feature expression of given data through unsupervised learning, so as to reconstruct the input data [6]. Based on the auto-encoder, researchers have derived a variety of auto-encoders. Hinton [7] and others improved the original shallow structure, proposed the Inception and training strategy of deep learning neural network, and then produced the Denoising Auto-Encoder (DAE); in 2007, Benjio [8] proposed the Inception of Sparse Auto-Encoder (SAE); in addition, there were Marginalized Denoising Auto-Encoder (MDA) and Stacked Sparse Denoising Auto-Encoder (SSDA) [9].

In this paper, a neural network denoising model based on a convolutional autoencoder is used to speed up the operation speed of the network. This network model has changes in the size of the convolution layer. The Inception module is used to expand the network width to better extract noise image features. The network structure of the improved Inception deconvolution module is used. Batch normalization and random inactivation are used to prevent overfitting. The length and width of the convolution layer are inversely proportional to the number of feature maps, which greatly reduces the number of network parameters. The data set uses VOC2012. Due to the huge content of the data set, 1000 pictures are randomly selected as the training set, 700 of which are used as the training set, and 300 pictures are used as the test set. Classical image data is used for comparative experiments. It is proved by experiments that the algorithm structure in this paper is more robust to denoising and has better denoising effect.

2 Network Structure of Convolutional Auto-encoder

Image denoising is the process of processing and restoring the noisy image. In this paper, the lightweight network structure is used to achieve excellent denoising effect and the deep learning network structure of four-layer convolutional auto-encoder is used. In order to speed up network training, the data in the data set is divided into 20×20 sizes. After adding the noise, the original image content is stored in different H5 files to speed up file reading and complete network training better.

2.1 Generate Noise Image

Gaussian noise is a kind of random noise which accords with normal distribution, and it is also the most common noise distribution. As shown in formula (1),

$$\begin{aligned} &Z \sim N(\mu, \sigma) \\ &T_{(h,w,c)} = X_{(h,w,c)} + k \cdot Z \\ &X_{(h,w,c)} = \begin{cases} 0, & T_{(h,w,c)} < 0 \\ 255, & T_{(h,w,c)} > 255 \\ T_{(h,w,c)}, & others \end{cases} \end{aligned} \tag{1}$$

Among them, Z is the noise data, which conforms to the normal distribution with the expectation of μ and the variance of σ. k is the noise intensity, and $X_{(h,w,c)}$ is the image pixel. Finally, the value of image pixel after noise is added into the formula to limit, so as to avoid data overflow [10]. In this paper, we use the Gaussian noise data set with noise level of 25 to denoise the data. As shown in Fig. 1, we can see the difference between the noisy image and the original image. In this paper, we remove the image noise based on the noise level of 25.

Fig. 1. Comparison between the original image and the noise image

2.2 Multi Feature Extraction Inception Module

Inspired by popular image processing algorithms such as VGG Net and GoogLeNet, the InceptionV3 module is used to extract image features and restore images, and good results are achieved. In order to solve the problem of increasing the depth and width of the network while reducing the parameters, the Inception module mainly improves the traditional convolution layer in the network, the structure diagram is shown in Fig. 2. The inception module calculates several different transformation results on the same input map in parallel, and connects their results into one output. Using the inception module is conducive to extract as much feature information as possible from different convolution kernel sizes of noisy images, and provides better generalization ability for the model network. Therefore, this paper improves on the basis of inception, changes the original convolution layer to deconvolution layer for up sampling operation, uses the combination of small convolution kernels of 1×1, 3×3, 5×5 to reduce the channel dimension of the feature image, better restores the feature image, and makes it closer to the original image.

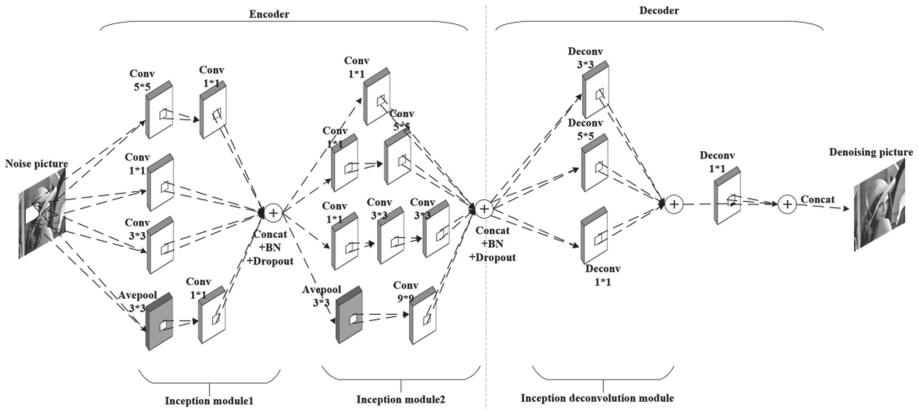


Fig. 2. Network structure of convolutional auto-encoder denoising

Different from the up sampling operation, this paper uses two-layer inception module to process noise image. This brings another problem: the number of feature maps in each layer increases, and the cost of computation increases greatly. Therefore, this paper makes the following settings for the inception module:

1) Each convolution layer of Inception is added to the ReLu activation function, which simplifies the calculation process. The dispersion of activity makes the calculation cost of Inception module decrease;
2) Add batch normalization (BN) and random deactivation layer (Dropout). BN layer can accelerate the training speed of inception network many times, improve the generalization ability of the network, normalize the output to the normal distribution of n (0, 1), reduce the distribution of internal neurons, accelerate the training speed, and produce more stable nonlinear output. In the experiment, it is found that the training PSNR is not stable when only BN layer operates, and the problem of non data set and non verification set is considered After that, the Dropout layer is used to solve the over fitting phenomenon in the model training process. The results show that the Dropout layer can reduce the PSNR instability. In Dropout learning process, part of the weight or output of the hidden layer is randomly zeroed to reduce the interdependence between nodes.

2.3 Design of Convolutional Auto-Encoder Network Based on Inception Module

In order to make the denoising network model be able to process natural images, the data of each image is transformed into a three-dimensional matrix. The convolutional auto-encoder is divided into two parts: decoder and encoder. There are four layers in total. The structure of the convolutional auto-encoder denoising network based on the Inception module is shown in Fig. 2. The advantage of the network is that it uses auto-encoder structure, encoder and decoder of coding layer, the first two layers of InceptionV3 classic structure in Encoder and deconvolution module in Decoder. The advantage of this module is that it can restore the noise image features extracted from the encoder to a greater extent,

and it can restore the original image features better than one layer deconvolution. The specific network settings are as follows:

Encoder:

The first layer: it consists of five different scale convolution layers and an average pooling layer to form the Inception module. It can enlarge the width of convolution auto-encoder, extract information of different sizes of image using multiple convolution kernels, and fuse them to get better representation of image. The first layer of convolution layer is $5 \times 5 \times 32$, $1 \times 1 \times 64$, and the image output channel is 64; the second layer is $3 \times 3 \times 64$, and the image output channel is 64; the third layer is $1 \times 1 \times 64$, and the image output channel is 64; the fourth layer is the average pooling layer, with a step size of 1, and the pooling layer is 3×3, followed by a layer of $1 \times 1 \times 32$ convolution layer, and the image output channel is 32. Input of each layer is added with standardization, and padding is same using ReLu function to prevent the gradient from disappearing. Finally, the Concat layer is used to connect, the standardized BN layer is added, and Dropout is used to prevent over fitting. At this time, the output channel of the picture is $64 + 64 + 64 + 32 = 224$;

The second layer: using the structure of the second module in Inception V3, the first layer of Conv is $1 \times 1 \times 64$, the second layer is $1 \times 1 \times 48$, $5 \times 5 \times 64$; the third layer is $1 \times 1 \times 64$, $3 \times 3 \times 96$, $3 \times 3 \times 96$, the fourth layer is the average pooling layer, the pooling layer size is 3×3, and the step size is 1. After the pooling layer is connected with a convolution layer, the convolution core size is 1×1, and the channel size is 32. At last, Concat layer is used to connect, standard BN layer is added, and Dropout is used to prevent over fitting. After the Inception module of this layer, the output picture channel is $64 + 64 + 96 + 32 = 256$.

Decoder:

The first layer: the upper sampling layer is implemented by the improved Inception module using deconvolution. It is composed of four different dimensions of anti convolution layers, which are $3 \times 3 \times 16$, $5 \times 5 \times 16$, $1 \times 1 \times 16$, step size is set to 2, and Concat layer is used for connection. Using the improved Inception module for deconvolution can make the feature fusion better. At this time, the shape of the picture is $20 \times 20 \times 64$, adding BN layer for standardization operation;

The second layer: use deconvolution to realize the upper sampling layer, and use the upper sampling layer to process the image of decoder. In order to get the same size of the original image, use the upper sampling layer of the first layer to realize, and restore the image to the original size. At this time, the image shape is $20 \times 20 \times 1$.

In conclusion, in order to improve the robustness of image denoising, the convolution operation of the Inception module is introduced to improve the convolution operation in the Inception module, better feature extraction of noise image, use the ReLu function to prevent the gradient from disappearing, introduce BN and Dropout operation to prevent network over fitting, improve the overall denoising performance of the model, and shorten the training time.

The flow of image denoising using the network is shown in Fig. 3. With the increase of training times, the verification set is used to evaluate whether the model is over fitted. The specific operation is: set the number of nodes to 500, after training the corresponding parameters through the training set, the verification set is used to detect the error of the

model, and then change the number of nodes. If the error of the model is greater than 100% or less than 0%, stop the network immediately and make corresponding modification.

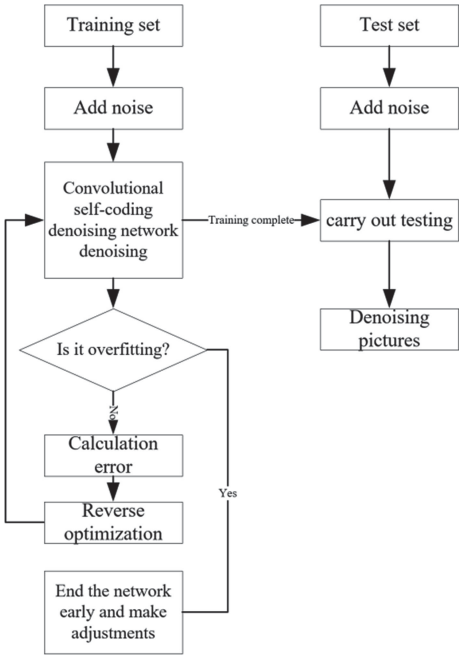


Fig. 3. Training flow of decontamination network of convolutional auto-encoder

3 Experiments and Results

3.1 Experimental Data

In the image denoising experiment based on convolutional auto-encoder, VOC2012 data set is very large, so 1000 of them are randomly selected as data sets, 700 of them are training sets and 300 are test sets. At the same time, 10 standard images commonly used in the field of image denoising are used as reference images for comparative experiments. All the images in voc2012 are color images, while the gray image is used in this paper. Therefore, it is necessary to convert the color image into gray image and add Gaussian noise with noise level of 25. In order to facilitate training, the input image is cut into 20×20 sub image blocks, and the cut-out image is stored in an H5 file every five original images, which is convenient for model reading and training.

3.2 Experimental Environment and Evaluation Criteria

The experimental environment system is configured as windows 10 system, the processor is Intel Core i7-3370 CPU, and the memory is 8 GB.

Peak Signal to Noise Ratio (PSNR) and Structural Similarity (SSIM) are used as noise reduction evaluation indexes. As shown in formula (3) and (4).

1) Mean Square Error (MSE)

$$MSE=\frac{1}{M\times N}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(\hat{f}(i,j)-f(i,j)\right)^2 \tag{2}$$

MSE represents the mean square error of the current image $\hat{f}(i,j)$ and the reference image $f(i,j)$. M and N are the height and width of the image respectively.

2) Peak Signal to Noise Ratio (PSNR)

$$PSNR=10\log_{10}\left(\frac{(2^n-1)^2}{MSE}\right) \tag{3}$$

Where MSE (formula (2)) is the mean square error between the original image and the denoised image, n is the number of bits per pixel, generally 8, that is, the number of pixel gray scale is 256. The unit is dB, the larger the PSNR value is, the less the representative distortion is.

3) Structural Similarity (SSIM)

$$SSIM(x,y)=\frac{(2\mu_x+C_1)(2\sigma_{xy}+C_2)}{\left(\mu_x^2+\mu_y^2+C_1\right)\left(\sigma_x^2+\sigma_y^2+C_2\right)} \tag{4}$$

Where μ_x is the average of x, μ_y is the average of y, σ_x^2 is the variance of x, σ_y^2 is the variance of y, σ_{xy} is the covariance of x and y. $c_1=(k_1L)^2$ $c_2=(k_2L)^2$ is a constant used to maintain stability, L is the dynamic range of pixel value, $k_1=0.01$, $k_2=0.03$. The range of similar structure is 0–1. The larger the value of similar structure is, the closer the two images are.

3.3 Influence of Network Model on Denoising Performance

The Influence of Inception Module on Noise Reduction Performance. In this paper, the network uses the Inception module to extract the features of images and in order to show the feature extraction ability of multiple Inception structures in this paper, the common convolutional auto-encoder, one layer of Inception module and multiple Inception modules in this paper are used for PSNR comparison. The experimental results are shown in Fig. 4. In the contrast experiment, the same decoding layer is set up, which is $3\times3\times32$ and $3\times3\times1$ anti convolution layer respectively. Different methods are used in the encoder. The encoder adopts the structure of two layers of convolution layer, and the general auto-encoder uses two 3×3 convolution layers as the encoder; the first layer of Inception module uses the first Inception module and the second layer uses the 3×3 convolution layer; the two layers of Inception module uses the module used in this paper. Using the same experimental environment and training set, the training process outputs the training PSNR. After 500 times of training, it can be seen from Fig. 4 that the PSNR value of the algorithm in this paper keeps rising during the training process,

up to 25 dB or more, while the initial stage of the ordinary convolutional auto-encoder is poor, and it slowly drops in the early stage of the sudden rise and the later stage, and finally it is about 19 dB; the first layer of Inception rises slowly after the fluctuation in the early stage, and finally it is about 21 dB. The two-layer Inception module used in this paper is superior to the other two methods from the beginning, and finally it is gentle at about 23 dB. From the experimental results, we can see that the more training times, the better stability and robustness of this algorithm, so it shows better denoising results.

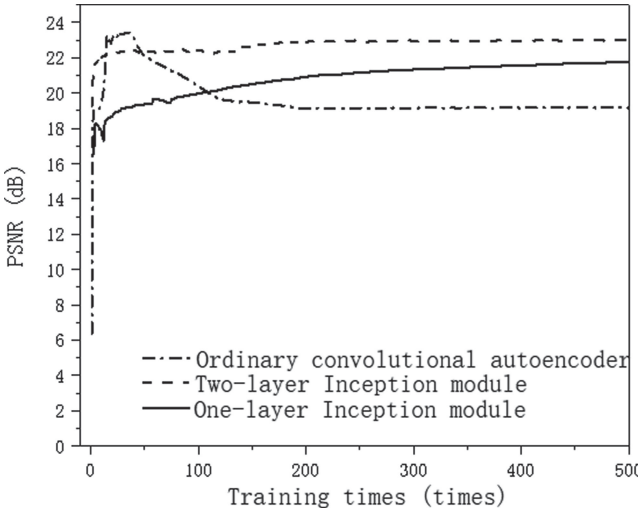


Fig. 4. Comparison of the effects of different coding layers on the model

The Influence of the Deconvolution Module of Inception on the Performance of Denoising. In this paper, we use the Inception deconvolution to feedback the extracted features. Compare the influence of one layer deconvolution and Inception deconvolution module on image denoising. The design of the coding layer of the comparison experiment is consistent with the encoder of this paper. The decoding layer uses a deconvolution layer with a convolution core of 3×3 and the deconvolution module of this paper to carry out the comparison experiment. The experimental results are shown in Fig. 5. From Fig. 5, it can be seen that the ordinary one-layer deconvolution in the early training is relatively stable, while the Inception deconvolution module fluctuates briefly. In the later training, the effect of using the Inception module is better than that of the ordinary one-layer deconvolution, and the final stability is about 24 dB. The overall effect is very good when using the Inception deconvolution module.

3.4 Comparative Experimental Analysis

In order to verify the robustness of the method in this paper, 10 classic test images are selected for simulation experiment, and compared with literature [11], literature [12] and

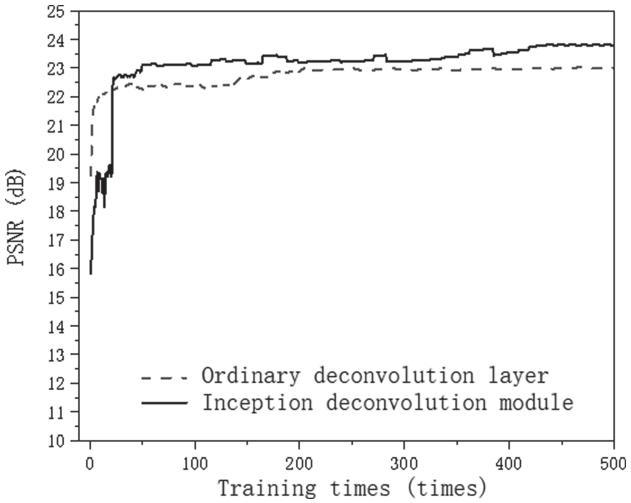


Fig. 5. Comparison between one layer deconvolution and Inception deconvolution

literature [13], as shown in Table 1 and Table 2, Table 1 is the peak signal-to-noise ratio of ten images, and Table 2 is the structural similarity of each method. Both literature [11] and literature [12] use deep convolution neural network for image denoising, as shown in Table 1 and Table 2, the algorithm in this paper shows good denoising effect, with an average increase of PSNR by 11.088 and SSIM by 0.451 compared with the original image; literature [11] and literature [12] use 5-layer deep convolution neural network. The difference is that the first three layers of literature [11] are convolutions, the last two layers are anti convolutions, and literature [12] uses five convolutions to denoise. Compared with literature [11] and literature [12], PSNR and SSIM increased by 2.813 and 0.821, respectively, and 5% and 1.9% respectively. In reference [13], the PSNR and SSIM were increased by 2.626% and 1.1% respectively under the same experimental environment by using one-layer Inception module and five-layer convolution layer.

Table 1. Peak signal to noise ratio (PSNR) of each method for ten images

PSNR	House1	Woman1	Woman2	Man	Camera	Lena	Barbara	Boat	House2	Girl
Original drawing	20.18	20.21	20.34	20.22	20.58	20.23	20.31	20.30	20.35	20.63
Document [11]	28.13	28.36	30.38	28.01	27.99	29.19	26.58	27.68	30.08	29.7
Document [12]	29.19	29.32	30.85	30.11	30.28	30.21	29.58	30.22	30.77	30.09
Document [13]	29.46	29.15	29.22	28.84	28.15	28.86	29.03	28.26	28.69	28.31
This paper	30.12	31.55	32.08	31.69	30.59	32.54	30.04	32.13	31.82	31.67

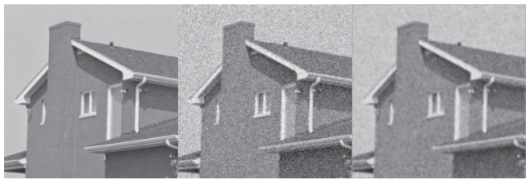
Table 2. SSIM of ten images

SSIM	House1	Woman1	Woman2	Man	Camera	Lena	Barbara	Boat	House2	Girl
Original drawing	0.43	0.49	0.42	0.53	0.49	0.49	0.57	0.53	0.42	0.47
Document [11]	0.90	0.89	0.92	0.87	0.89	0.91	0.85	0.87	0.85	0.90
Document [12]	0.92	0.90	0.92	0.93	0.92	0.92	0.94	0.92	0.89	0.90
Document [13]	0.93	0.92	0.93	0.93	0.93	0.92	0.93	0.94	0.91	0.90
This paper	0.94	0.93	0.93	0.94	0.92	0.93	0.95	0.96	0.93	0.92

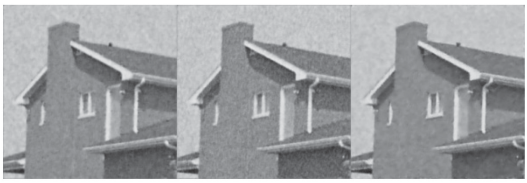
Select five of the images for output comparison. The comparison figure is shown in Fig. 6. The image in this paper has a good visual effect and a clear edge. Through the details, it can be seen that the denoising algorithm in this paper has a good effect and the details are processed in place, showing the image after denoising more clearly.

4 Epilogue

The algorithm of this paper adopts the structure of convolutional autoencoder, using coding layer and decoding layer structure to clearly divide the network into two parts. Among them, the coding layer uses multiple Inception modules for feature extraction, and the decoding layer improves the traditional Inception module, modifying the convolution network into a deconvolution network, so that the image can make full use of the advantages of Inception module feature extraction in the deconvolution network. Better integrate image features, restore original image information. From the experimental results, it can show good robustness in image denoising. However, there are also deficiencies. Compared with the convolutional neural network and the convolutional autoencoder without the Inception module, the four-layer network proposed by the algorithm of this paper takes a long time. After preliminary experimental tests, it is known that the use of the Inception module causes the network to become wider, and the volume The cumulative neural network has experienced more operations, so how to shorten the model training time is the focus of future research.



House1 Original Drawing Noise Figure Document[12]



Document [13] Document [14] This Paper

Woman2 Original Drawing Noise Figure Document [12]

Document [13] Document [14] This Paper

Camera Original Drawing Noise Figure Document [12]

Fig. 6. Rendering of each algorithm

Document [13] Document [14] This Paper

Lena Original Drawing Noise Figure Document[12]

Document [13] Document [14] This Paper

House2 Original Drawing Noise Figure Document [12]

Document [13] Document [14] This Paper

Fig. 6. (*continued*)

References

1. Gai, S., Bao, Z.Y.: New image denoising algorithm via improved deep convolutional neural network with perceptive loss. Exp. Syst. Appl. **138**, 112815 (2019)

2. Schuler, C.J., Christopher Burger, H., Harmeling, S.: A machine learning approach for non-blind image deconvolution. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 1067–1074 (2013)
3. Li, Y.J., Zhang, J., Wang, J.: Improved BM3D denoising method. IET Image Process. **11**(12), 1197–1204 (2017)
4. Burger, H.C., Schuler, C.J., Harmeling, S. Image denoising: can plain neural networks compete with BA-I3D? In: Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), Providence, RI, USA, pp. 2392–2399. IEEE (2012)
5. Chen, Y., Pock, T.: Trainable nonlinear reaction diffusion: a flexible framework for fast and effective image restoration. IEEE Trans. Pattern Anal. Mach. Intell. **39**(6), 1256–1272 (2016)
6. Niu, W.H., Meng, J.L., Wang, Z.: Image denoising based on adaptive contraction function contourlet transform. J. Graph. **4**, 17 (2015)
7. Hinton, G.E., Osinder, S., The, Y.W.: A fast learning algorithm for deep belief nets. Neural Comput. **18**(7), 1527–1554 (2006)
8. Bengio, Y., Lamblin, P., Popovici, D., et al.: Greedy layerwise training of deep networks. In: Proceedings of the 20th Annual Conference on Neural Information Processing System, pp. 153–160 (2006)
9. Ma, H.Q., Ma, S.P., Xu, Y.L., et al.: Image denoising based on improved stacked sparse denoising autoencoder. Comput. Eng. and Appl. **54**(4), 199–204 (2018)
10. Chen, Q., Pan, W.M.: Design and implementation of image denoising based on autoencoder. J. Xinjiang Normal Univ. (Nat. Sci. Ed.) **37**(02), 80–85 (2018)
11. Li, C.P., Qin, P.L., Zhang, J.L.: Research on image denoising based on deep convolution neural network. Comput. Eng. **43**(03), 253–260 (2017)
12. Zhang, K., Zuo, W., Chen, Y., et al.: Beyond a gaussian denoiser: residual learning of deep cnn for image denoising. IEEE Trans. Image Process. **26**(7), 3142–3155 (2017)
13. Li, M., Zhang, G.H., Zeng, J.W., Yang, X.F., Hu, X.M.: Image denoising method based on convolution neural network combined with Inception model [J/OL]. Comput. Eng. Appl., 1–8 (2019)
14. Zhou, M.J., Liao, Q.: Knowledge push based OR attribute similarity. Comput. Eng. Appl. **47**(32), 135–137 (2011)
15. Huang, Z., Li, Q., Fang, H.: Iterative weighted sparse representation for X-ray cardiovascular angiogram image denoising over learned dictionary. IET Image Process. **12**(2), 254–261 (2018)
16. Xiang, Q., Pang, X.: Improved denoising auto-encoders for image denoising. In: 2018 11th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Beijing, China, pp. 1–9 (2018)