Research on Application of Deep Learning Based on Artificial Intelligence in High-noise Image Denoising

Sainan Wang1,*

1Nanjing Engineering Vocational College, Jiangsu Union Technical Institute, China

*Corresponding author e-mail: Wangsn@njvec.edu.cn

Abstract. Image denoising is to reduce or eliminate the influence of noise on the image to obtain the original image with less error. In order to remove the noise in the image more effectively, this article builds a convolutional neural network for image denoising based on the artificial intelligence-based deep learning method, and trains it through a high-noise image data set, and analyzes the training data set and model its own influence on the denoising effect, the results show that the increase in the number of layers is beneficial to denoising.

Keywords: Deep Learning, Image Denoising, Artificial Intelligence, Image Denoising

1. Introduction
With the continuous development of science and technology, image imaging methods have become more diversified, such as aerial remote sensing, handheld devices, etc. However, the image collection, transmission, and processing are often affected by subjective and objective factors such as environment and equipment [1, 2], produce "noise", thereby affecting the image quality of the image. Therefore, image denoising is to reduce or eliminate the influence of noise on the image to obtain the original image with smaller error [3-5]. Currently, the methods of processing image denoising mainly include transform domain-based and statistical information-based. Image denoising methods based on artificial intelligence deep learning have gradually become a research hotspot. In the field of image processing, deep learning methods have a high accuracy rate, can extract image features more fully and accurately, complete object detection and other tasks, and have better adaptability to large-scale data sets [6].

Based on the deep learning method of artificial intelligence, this paper constructs a convolutional neural network for image denoising, and trains it through high-noise image data sets to verify the effectiveness of the method.

2. Convolutional neural network
Convolutional Neural Network (CNN) is a type of artificial neural network. It is one of the most widely used deep learning models and is often used to solve problems such as image processing. The structure of convolutional neural network has weight sharing. This structure reduces the complexity of the network model and the number of parameters. When the input of the network is an image, this advantage is more obvious.
The auto-encoder is a neural network model. In the auto-encoder model, set the input data set as $x$, $x$ gets $h$ through the encoder, and $h$ represents the key feature of the compressed $x$; $h$ gets through the decoder, which is $x$ weight The result of the structure. Ideally, the main characteristics of $x$ can be retained. The so-called denoise autoencoder (dA) refers to such an automatic encoder: its input is noisy data, and the model After training, the expected output prediction results are lossless and pure data.

When the input data is an image, a convolutional neural network can be combined with an automatic encoder to form a convolution autoencoder (CAE). In a convolutional autoencoder, a convolutional layer is used instead of a fully connected layer for feature extraction. In the processing of medical images, such as nuclear detection on breast cancer images, the method of automatic encoding has begun to be applied (Figure 1).

![Figure 1. Typical convolutional neural network process.](image)

3. Deep learning image denoising based on artificial intelligence

3.1. Network structure
The number of parameters that need to be trained is $3 \times 3 \times 32 \times 64+64$; the C2 layer is performed with 64 cores of size $(1 \times 1)$ Convolution, corresponding to the generation of 64 feature maps, the number of parameters that need to be trained is $1 \times 1 \times 64 \times 64+64$; the D· layer undergoes $(3 \times 3)$ deconvolution operation to generate the D2 layer, which contains 32 feature maps, The number of parameters to be trained is $3 \times 3 \times 64 \times 32+32$; the size of the convolution kernel is increased to $(5 \times 5)$, the output feature map size is set to 1, and the number of parameters to be trained is $5 \times 5 \times 32 \times 1+1$, the result of deconvolution output at this time is the denoised image output by the network output layer.

3.2. Convolutional layer
In the convolutional layer, the direct convolution operation between the feature map and the convolution kernel of $(5 \times 5), (3 \times 3)$ and other sizes will reduce the size of the feature map of the next layer. Therefore, it is considered to extend the boundary of the feature map.

The calculation form of the convolutional layer is:

$$Y_j^i = f \left( \sum_{j \in M} W_{ij} \ast Y_{ij}^{i-1} + B_i \right)$$  (1)

Among them, $l$ represents the number of layers; $Y_j^i$ is the $j$-th feature map of the $l$-th layer; $W$ is the convolution kernel; $B$ is the offset corresponding to the $j$-th feature map; $\ast$ is the convolution operation. The convolution kernel $W$ is equivalent to $n_i$ filters, and the filter size is $k_i \times k_i$, and $k_i$ generally takes 1, 3, 5, etc.

$$f(x) = \max(0, x)$$  (2)
3.3. Training
This article Using mean square error as the cost function of the network, the cost function is:

\[ L = \frac{1}{n} \sum_{j=1}^{n} \left( \frac{1}{wh} \sum_{j=1}^{k} \| D_{i,j,k} - X_{i,j,k} \|^{2} \right) \]  

(3)

D is the image denoised by the network; X is the original image without noise. The weight update method is as follows:

\[ W_{i+1}^{l} = W_{i}^{l} + \Delta_{i+1}^{l} \]  

(4)
\[ \Delta_{i+1}^{l} = -\eta \frac{\partial L}{\partial W_{i}^{l}} \]  

(5)

Among them, t is the batch; l is the number of convolutional layers; \( \eta \) is the learning rate of the lth layer. The partial derivative of the error of the weight of the layer multiplied by a negative learning rate is the updated value of the weight of the layer.

4. Experiment and result analysis
4.1. Experimental Platform and Data Set
For denoising training, the GPU used is NVIDIA Tesla K40. At the same time, experiment with Tensorflow deep learning framework.

The images used for training in this article are from VOC2018, and 15,000 images are randomly selected as the training set. All the images of VOC2018 are color images, and the gray-scale image denoising network needs to first convert the color image into a gray-scale image and add noise.

The input and output data used in training are all image fragments. This article uses a sliding window method to generate image fragments. The size of the sliding window is \( f_{patch} \times f_{patch} \), and the step size is \( \lfloor w_{patch}/2 \rfloor \). In order to make the network learning ability stronger, this article will also perform out-of-order operations on the training set.

4.2. Data pre-processing
(1) Read the mrc file and star file

The data used in the experiment in this article contains two parts: mrc file and star file.

The mrc file contains the pixel information of the photo file, and the gray value of each pixel indicates the strength of the electronic signal. First, use the EM-Data module in the EMAN2 software package to read in the data and store it as an equivalent png file. The pixel value is L2 normalized, and the value of the pixel matrix is compressed to 0–255.

The star file is the file output by the mrc file after the particles are selected by the relion software. The first two columns in the file are the horizontal and vertical coordinates of the particle center point in the image. In the experiment, the file is read in the python language and the particle coordinates are saved Prepare for subsequent processing.

(2) Histogram equalization

Histogram equalization is one of the commonly used methods of image processing. The basic idea is to broaden the denser gray levels in the image, and compress the sparse gray levels in the image, so that the original value can be taken The dynamic expansion of the range of the image can effectively improve the contrast of the image and the change of gray tones, thereby making the image clearer.

(3) Image cutting

In order to provide training and test samples, you first need to obtain a local area containing particles and only noise from the original image, and save it as a 256*256 size image. For the original mrc image, according to the particle center point given in the star file ( x,y), save the area around it with a size of 256*256, and get the particle data set.
4.3. The impact of training data on denoising performance

Training data can affect the denoising effect of images. In order to compare the effect of training data on denoising performance, this paper designs experiments for three influencing factors: the size of training image fragments and the number of training images.

In order to compare the impact of the size of the training segment on the network learning ability, the network was trained with multiple scale training sets, including a $19 \times 19$ small-scale training set and a $43 \times 43$ large-scale training set. The training input sample noise adopts Gaussian noise with $\sigma=25$. In order to show the denoising performance of the network under different data sets, Lena images processed by Gaussian noise with $\sigma=25$ are used as test images. The larger the training image segment in a certain range, the more noise features it contains, and the better the denoising effect. However, the more training cycles are required for the network to stabilize, which requires longer training time.

This article uses data sets of different sizes to train the network, including a small data set containing 100 images (S-100), a medium data set containing 1000 images (M-1000), and a large data set containing 15,000 images (L-15000). The images used in various training sets are randomly selected from the VOC2012 data set, and are processed with Gaussian noise with $\sigma=25$.

4.4. Comparative experiment analysis

The denoising results are compared with the existing algorithms GSM, KSVD, CN2, MLP with superior performance. In this simulation experiment, the noise added to the test image is Gaussian white noise with $\sigma=25$, and all algorithms use the CPU for calculation processing. Two evaluation indicators, peak signal-to-noise ratio (PSNR) and processing time, are used to measure the denoising effect. Table 1 shows the average PSNR and average time consumption of each method for denoising 10 test images.

| Performance evaluation factors | GSM   | KSVD  | CN2   | MLP   | Method of this article |
|-------------------------------|-------|-------|-------|-------|-----------------------|
| Average PSNR/dB               | 29.36 | 29.65 | 29.26 | 30.17 | 30.20                 |
| Average time/s                | 4.21  | 96.35 | 4.02  | 14.23 | 3.82                  |

The PSNR of the method proposed in this paper is better than GSM, KSVD, CN, and slightly better than the multi-layer perceptron network (MLP) at a noise level of $a=25$. Compared with the time taken to process the same picture, this paper proposes The method takes the least time. Compared with GSM and KSVD, CN2 denoising takes less time. It uses a neural network with 4 layers and 24 feature maps. When denoising, you only need to input a noisy two-dimensional image, so the time is shorter. This article uses the method of feature maps to construct a symmetrical convolutional neural network, so the denoising time is less than CN2, and the deconvolution layer used in this article has a greater effect on noise removal. A good denoising effect has been achieved. MLP uses a 4-layer fully connected neural network to process image fragments, and then merges all processed image fragments to form a denoised image, which is time-consuming. The input and output of the method in this paper are the entire image, and there is no limitation on the size of the image. The processing time can be shorter when the peak signal-to-noise ratio is high.

All three denoising methods can effectively remove the noise in the image. Observing the local image, we can find that the denoising method based on deep convolutional neural network can get clearer edges and restore more texture details, which shows that the GSM and KSVD methods have better visual effects of denoising.

5. Conclusion

Based on the deep learning method of artificial intelligence, this paper constructs a convolutional neural network for image denoising, and trains it through a high-noise image data set, and analyzes the
impact of the training data set and the model itself on the denoising effect. The results show The increase in the number of layers is conducive to noise removal.

References
[1] Khaw H Y, Soon F C, Chuah J H, et al. High-density Impulse Noise Detection and Removal Using Deep Convolutional Neural Network with Particle Swarm Optimization[J]. IET Image Processing, 2018, 13(2):365-374.
[2] Byeongjoon, Kim, Minah, et al. A performance comparison of convolutional neural network-based image denoising methods: The effect of loss functions on low-dose CT images.[J]. Medical physics, 2019, 46(9):3906-3923.
[3] Ali H, Rada L, Badshah N. Image Segmentation for Intensity Inhomogeneity in Presence of High Noise[J]. IEEE Transactions on Image Processing, 2018, 27(8):3729-3738.
[4] Yi X, Babyn P. Sharpness-Aware Low-Dose CT Denoising Using Conditional Generative Adversarial Network[J]. Journal of Digital Imaging, 2018, 31(5):1-8.
[5] Wu H, Wan Y, Tang M, et al. Real-Time Denoising of Brillouin Optical Time Domain Analyzer with High Data Fidelity Using Convolutional Neural Networks[J]. Journal of Lightwave Technology, 2019, 37(3):2648-2653.
[6] Dequan, Shuyan, Dong, et al. Phase diversity algorithm with high noise robust based on deep denoising convolutional neural network.[J]. Optics express, 2019, 27(16):22846-22854.