

Convolutional Neural Network for MR Image Noise Removal

Po-Ting Chen(陳柏廷) Chiou-Shann Fuh (傅秋善) Jyh-Horng Chen(陳志宏)

¹The Graduate Institute of Biomedical Electronics and Bioinformatics,
National University of Taiwan,
E-mail: r07945040@ntu.edu.tw fuh@csie.ntu.edu.tw jyhhchen2@gmail.com

ABSTRACT

Because Magnetic Resonance Imaging (MRI) scans often take too much time, how to save time has become a major issue for studying MR images. Usually MR scan time is highly positively correlated with resolution and Number of EXcitation (NEX) of images. As long as a map with a low NEX number can be reconstructed into a map with a high NEX number, the scanning time can be effectively reduced. In this article, the post-processing method of Convolutional Neural Network (CNN) will be used to reconstruct the image. Let low-NEX images have the same quality as high-NEX images. Important image parameters used in this article are Signal-to-Noise Ratio (SNR), Peak Signal-to-Noise Ratio (PSNR), Structural SIMilarity (SSIM).

Keywords: MRI CNN denoise

1. INTRODUCTION

In medicine, MRI is an indispensable source of human images, but long scanning time is the major problem. It is sometimes necessary to wait in line to scan the MRI, thus causes delays the diagnosis. Many factors affect MR scanning, such as resolution, NEX times, and so on. Of course, research on resolution is in full swing, such as super resolution. This article hopes to reduce the SNR of the image produced by the machine to speed up the scan. Then use CNN post-processing to reduce image noise to get high-quality images. Of course, not only is it easy to restore the interpretation of the image, MRI-related analysis is also expected to be similar to the original image.

In order to solve the problem that MR scanning is too time-consuming, this article starts from the NEX number of images. Low NEX will cause low SNR problems, so it is necessary to reconstruct from low SNR images to high SNR, and try to avoid image errors in the process.

2. Method

In this paper, we regard image denoising as a common discriminant learning problem, that is, to separate noise from noisy images by convolutional neural network (CNN) instead of using explicit images to learn discriminant models.

2.1 Model Based Network

There are three reasons for using CNN. First, CNN with a very deep architecture is effective in improving the ability and flexibility to utilize image features. Secondly, considerable progress has been made in the regularization and learning methods used to train CNNs, including Rectifier Linear Unit (ReLU), batch normalization and residual learning. CNN can use these methods to speed up the training process and improve denoising performance. Third, CNN is very suitable for parallel computing on modern powerful GPU, which can be used to improve runtime performance.

2.2 Related Work

In 2016, DnCNN proposed to use conventional traditional filters to denoise images. Because the structure of medical imaging is too complicated and there is almost no room for fault tolerance, the traditional filter is somewhat inadequate. The reason is that traditional filtering will use a filter for the entire image, but for medical images with complex structures, using only one filter will cause part of the image structure to become blurred. For example, using BM3D has very good results in smooth areas, but complex structure areas (such as the cerebellum) will appear to be insufficient, so this is why we chose the neural network architecture this time.

In the past literature, the based-CNN network architecture proposed in the field of image denoising is not often used directly in medical imaging. Therefore, we access and retrain some mainstream and novel network architectures on medical images.

In the past, FCN, DnCNN, IRCNN, and BRDNet were commonly used in the mainstream, and they have good performance in general image denoising, but the problem of typical small data sets such as medical imaging needs to be corrected, including network bandwidth and complex background noise of medical imaging. The network bandwidth needs to be increased to accurately learn the boundary and uniformity on the image.

Networks with narrow bandwidths, such as DnCNN and IRCNN, will still have problems like traditional filters. Although they have been relatively corrected, they still seem to be insufficient.

Therefore, we will try our best to widen the width of the network when designing the network to avoid past problems. However, although BRDNet has tried to widen the width of the network, the selected networks are all similar to DnCNN. We believe that this will make the network unable to exert the best results, and the subnet of one party should be modified.

2.3 Data Pre-processing

The data set used this time is the actual images collected from the MRI machine, a total of 1576 images, of which 1400 are training data, 100 are validation data, 76 are test data, and the data size is $256 * 256$. In addition, the data sets collected this time are MR T1 images, and are concentrated in the middle slice, which is the slice that covers part of the brain. On the one hand, we think that the general interest is in this area. After all, the CNN architecture is a good way to extract image features. On the other hand, it is to obtain better training results.

Using deep learning methods, deep architectures can provide competitive results, but usually the premise is that a large amount of data can be obtained for training. In other words, in order to provide better model performance, a large number of data sets are required, and the acquisition of data is a difficult problem when it comes to medical images. Therefore, in addition to seeking suitable noise reduction techniques, the size of the data set must also be considered, and the designed model must be able to adapt to small data sets. So before starting training in this article, we used some common techniques to expand the training data set.

After collecting clean images, different noise masks were added, and the STDs were 10, 15, and 25, respectively. The input image is a noise image with a mask, and the output is noise, rather than taking a clean image as the output. The reason for this is that we found that the high-frequency signal of the image can be more perfectly retained. Although using noise as an output can avoid this problem, the quality of SSIM and PSNR parameters also needs to be considered.

3. Network Design

The model architecture is based on CNN. The most commonly used ones are BRDnet, IRCNN and DnCNN. All three are CNN-based model architectures.

There are different effects for different model architectures, and some of these model features are complementary. We found that although DnCNN has good noise reduction performance, there is still a problem of blur at the image boundary. The reason we guess is that the surface information will be lost when it is transmitted to the deep layer. Therefore, we connected the surface information with the deep information (similar to U-net) and found that the characteristics of the image boundary can be preserved after sacrificing part of the anti-aliasing effect.

3.1 Network Architecture

The network architecture we designed is divided into two parts, we call it MDNet as shown in Fig. 1. The upper layer of MDNet is the connected DnCNN, and the lower layer is the use of deep DnCNN. Using this to connect the upper and lower layers can increase the width of the model and can effectively improve the performance of reducing noise. We use Batch Normalization (BN) and ReLU to avoid the problem of gradient vanishing and gradient explosion, and can speed up the model convergence speed, more effectively used in the application of small data sets. For example, this application is used in medical imaging.

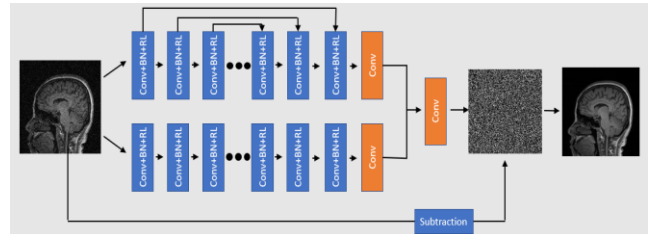


Fig. 1. Architecture of our proposed MDNet network. Conv = Convolutional layer, BN=Batch Normalization, RL = ReLU

In the upper layer network, we use a 26-layer CNN. After deducting the first and last layers, the rest are connected one after the other to ensure the retention of surface information. Here we use channel first, so the size of the first and last layer is $1 * C * 256 * 256$, the rest are $32 * C * 256 * 256$, the number of filters used is 32, and C is the number of channels. The MR image is a grayscale image, so here $C = 1$.

3.2 Boundary Artifact

Past literature has shown that deep CNN network architecture often causes boundary artifacts. Because in

many vision applications, it is usually required that the output image size should be the same as the input image size. This may cause border artifacts. Therefore, we directly fill zeros before convolution to ensure that each feature map of the middle layer has the same size as the input image. We found that a simple zero-fill strategy does not cause any boundary artifacts.

3.3 Loss Function

For the convenience of calculation, we used the mean square error (MSE) as the loss function. Take x as a noise image, y as a clean image, and z as noise. Can be obtained as Eq. (1).

$$X - Y = Z \quad (1)$$

Use residual learning to make the model predict the noise Z and then calculate the difference between the clean image and the original image to obtain the MSE value of the image. As seen in Eq. (2). In this way, the loss value of the model can be found, so that the model can be successfully trained.

$$MSE = \sum_{i=1}^n (y_i - y_i^p)^2 \quad (2)$$

We also tried to use SSIM as a loss function. It is expected that more visually complete images will be obtained. The SSIM formula is divided into three parts. Compare image brightness Eq. (3), contrast Eq. (4), and structural similarity Eq. (5). We found that the training results often make the proportions of the three lose their balance. For a good training effect, additional weights may need to be added. Therefore, it is not used here. But it is undeniable that SSIM is an extremely important parameter.

$$l(x, y) = \frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1} \quad (3)$$

$$c(x, y) = \frac{2\sigma_{xy} + c_2}{\sigma_x^2 + \sigma_y^2 + c_2} \quad (4)$$

$$s(x, y) = \frac{\sigma_{xy} + c_3}{\sigma_x\sigma_y + c_3} \quad (5)$$

In addition, this optimization function uses Adam, a known convenient and effective method. It can more effectively allow the model to extract features and accelerate the training speed of the model.

In summary, the network architecture we proposed this time has the following main advantages: (1) Using two sub-networks to connect instead of deepening the network can enhance the denoising performance. (2) Use BN and ReLU to realize the application of small data sets and avoid gradient explosion and vanishing. (3) Connect the network back and forth to preserve the image boundary information. (4) Get better PSNR and mathematical metric in the experiment.

4. Result

The loss function in the model used in this article is MSE, and the optimization function is Adam, a method that can effectively and quickly allow the model to converge. In addition, batch normalization is often used in the previous literature. If batch normalization is not used, it may cause the loss of loss or disappear. However, batch normalization may also lead to poor learning performance. In the past literature, it is mentioned that batch normalization is best used together with residual learning.

4.1 Model with ReLU Compared

Fig. 3 is a comparison graph of the results of batch normalization and residual learning.

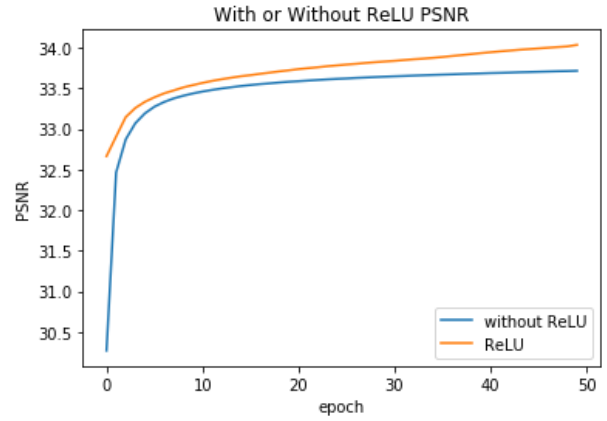


Fig. 3. It can be seen ReLU is used to allow the model to converge more quickly and efficiently.

The noise of the model used this time is Gaussian noise. We chose three different standard deviation noise masks, st. dev. = 10, 15, 25. The SNR of the original image is 118, and the SNRs obtained after adding Gaussian noise are 20.6, 13, and 7.2 dB, respectively. And the Fig.4 shown 3 different St.dev noise were added at clear brain image.

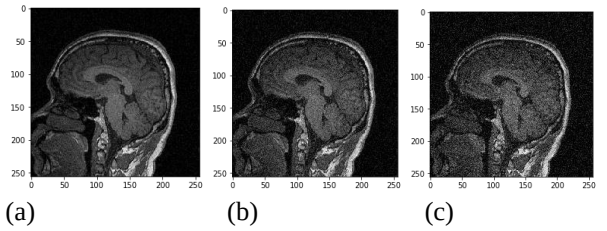


Fig.4 Dataset in each st. dev. (a) st. dev=10 (b) st. dev=15 (c) st. dev=25

4.2 Model PSNR

In Fig. 5, we can see the comparison between the PSNR of MDNet and the mainstream image denoising model in the past. Although higher results can be obtained, high PSNR does not necessarily mean an absolute improvement in image quality. There are many ways to get high PSNR (such as smooth), which is also a blind

spot that often falls into when using PSNR as an indicator, so we will use other image quality indicators for more verification in the future.

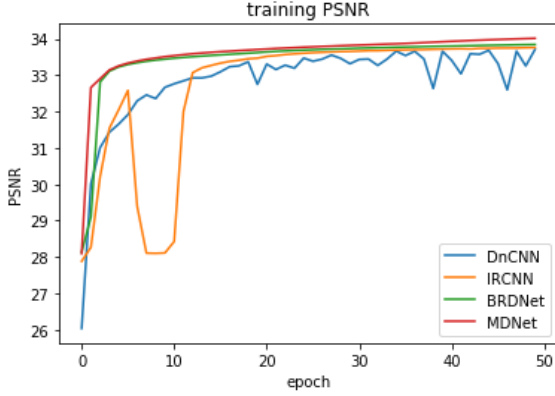


Fig. 5. It can be shown that MDNet can perform better than other models and get higher PSNR. And std=10

In Fig. 6, we could see the PSNR results of the four networks under three different standard deviations of Gaussian noise. It can be seen that MDNet has quite good training results, which can get 34.01dB at st. dev.=10 and 31.37dB at st. dev.=15, st. dev.=25 gets 28.18, which is higher than other network results.

	DnCNN	IRCNN	BRDNet	MDNet
Std=10	33.70	33.76	33.84	34.01
Std=15	31.13	31.14	31.20	31.37
Std=25	27.79	27.84	27.83	28.18

Fig. 6 Comparison of network training results under Gaussian noise of different st. dev.

4.3.1 Model output St. dev. = 10

In Fig.7, we can see that compared with the previous model, MDNet can present better details (such as the cerebellum). In a large area where the signal is strong (such as the gyrus and brainstem), the background noise can be more effectively eliminated, providing a more effective denoising effect.

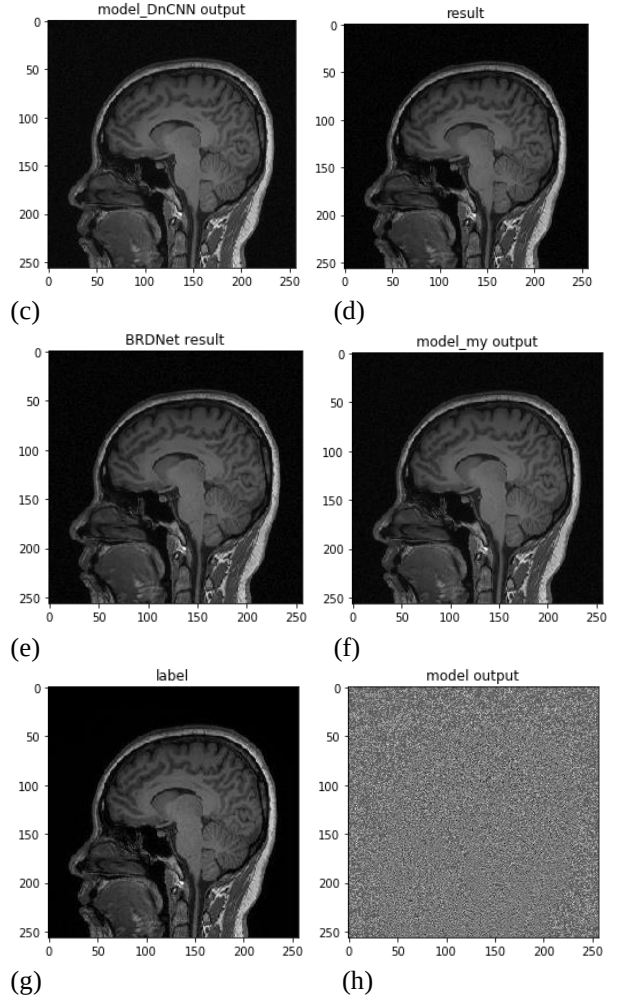
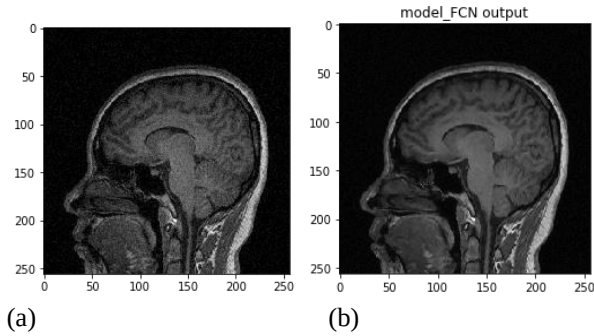
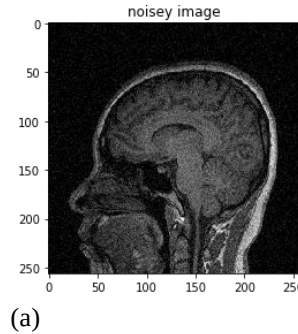


Fig. 7 (a) The noise image as input.(noise std=10) (b) The original simple FCN model output. (c) The DnCNN model output. (d)The IRCNN model output (e) The BRDNet model output. (f) The MDNet model output. (g) The original clear image. (h) The noise map obtained MDNet.

4.3.2 Model output st. dev. = 15

Fig. 8 shown each model output in noise st. dev.=15.



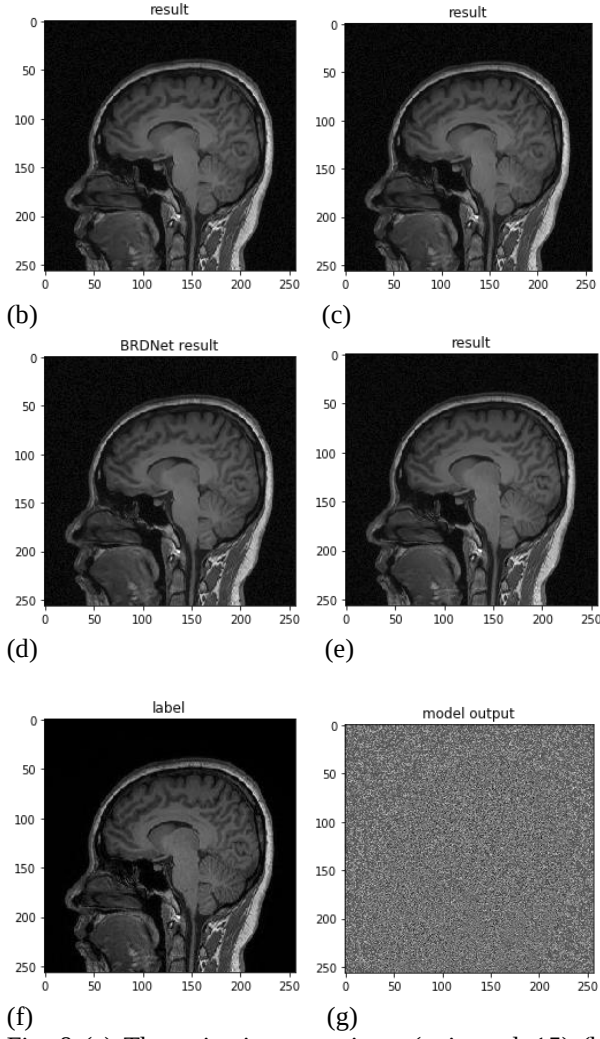


Fig. 8 (a) The noise image as input.(noise std=15) (b) The DnCNN model output. (c)The IRCNN model output (d) The BRDNet model output. (e) The MDNet model output. (f) The original clear image. (g) The noise map obtained MDNet.

4.3.3 Model output st. dev. = 25

Fig. 9 shown each model output in noise st. dev.=25.

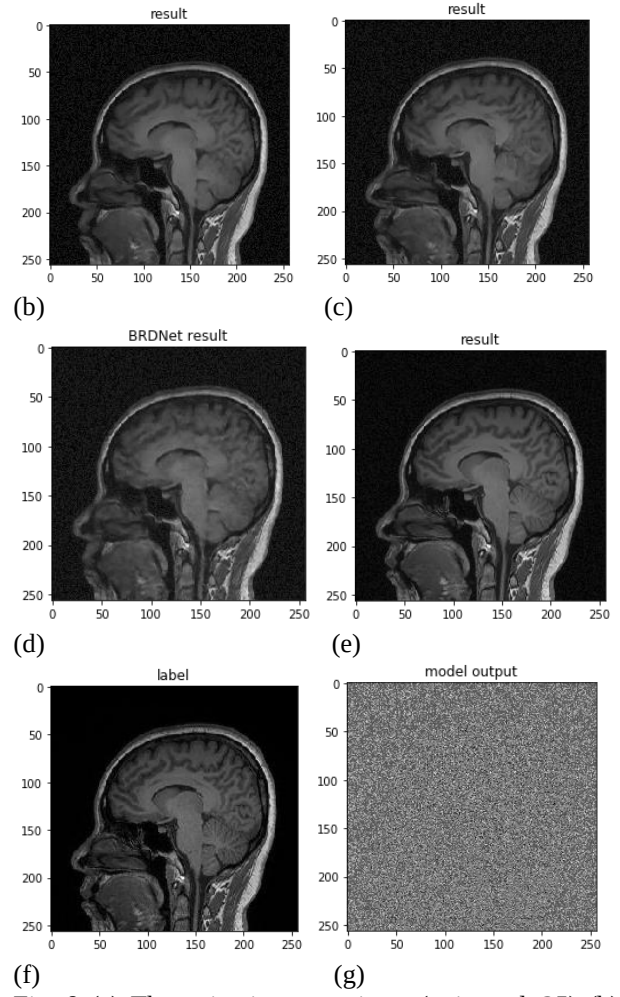
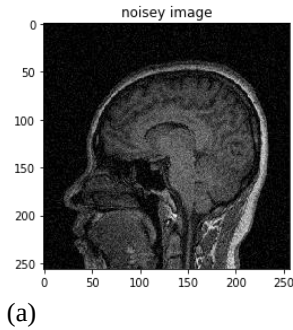


Fig. 9 (a) The noise image as input.(noise std=25) (b) The DnCNN model output. (c)The IRCNN model output (d) The BRDNet model output. (e) The MDNet model output. (f) The original clear image. (g) The noise map obtained MDNet.

4.4 SSIM Evaluation on Real Images

Although the image with noise added is easy to restore and explain, it does not necessarily work on real images. Because the background noise of real images is very complicated, it is not only Gaussian noise. Therefore, when using real images, we will use SSIM, which is closer to the visual effect, to measure the image quality. In the past, many documents suggested that PSNR does not completely represent the visual effect of the image. Even though PSNR is still an important reference, we still first pursue the visual effect of the image on the real image.

4.4.1 Real Image Resolution=1*1*(cm)

In order to verify the effectiveness of the model on real images, we obtained an additional set of T1 images from the machine (the detailed parameters are TR=2300ms, TE=2.4ms, Matrix size=256*256 pixels,

Average=1, resolution=1*1*1cm). Put it directly into the model without adding noise and compare the results with the images with high NEX times.

It can be seen in Fig. 10 that the background noise at the brainstem position can be effectively removed. It can also become clearer in places with complex structures such as the cerebellum. Another interesting thing is that the brain boundaries are also vaguely seen in the noise map trained by the model. This coincides with the Rician noise generated by the common background noise of MRI. In Fig. 11, the 152 sections of the whole brain are listed in SSIM, and most of them fall from 0.965 to 0.990. The destructuring effect of the middle section is slightly lower than that of the head and tail sections. We believe that the structure of the middle section is more complicated. This will cause SSIM to have some errors in calculations.

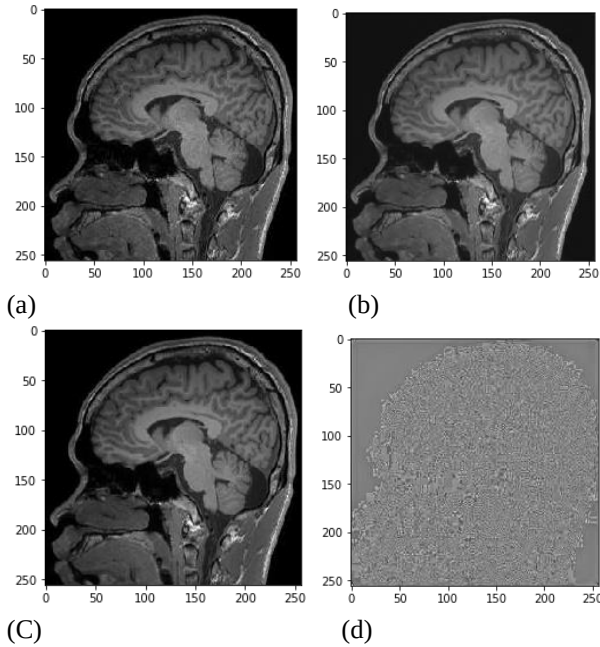


Fig. 10(a) Real MRI T1-weighted brain image (NEX=1) without any additional noise. (b) The subtraction result of the model after (a) after learning through the residuals. (c) Real MRI T1-weighted brain image (NEX=4). (d) The noise map obtained by model learning.

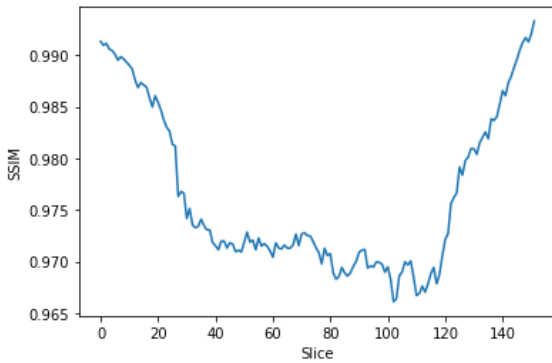


Fig. 11 Denoising results for 152-slice NEX=1 images of the whole brain.

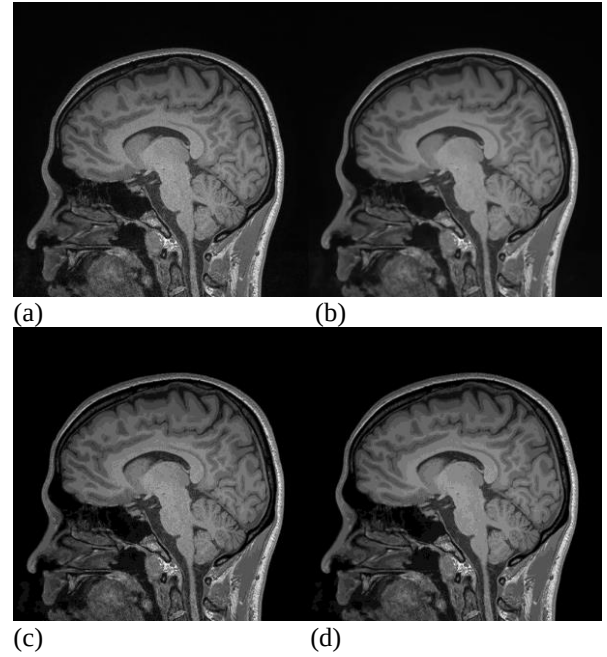
4.4.2 Real Image Resolution=0.5*0.5*1(cm)

Next, we tested the real image with more detailed resolution. In the principle of MR imaging, increasing the resolution will cause the signal intensity to decrease, and the overall SNR will become lower. In principle, it will reduce the SNR by 2 times the previous set of data. Our goal is also to reconstruct high-quality images.

The data and parameters used this time are TR = 2300ms, TE = 2.4ms, matrix size = 512 * 512 pixels, average value = 1, resolution = 0.5 * 0.5 * 1cm.

This time we add a comparison of other methods to actually take a look at the visual effects produced by each method. Comparison methods include BM3D, simple FCN, and our model MDNet. In Fig. 12, could see the original image, the products of each method and the drawing of partial enlargement.

It could be seen that MDNet still has better results in detail than the other two. Although BM3D can have a good denoising effect, it will make the image slightly blurred and distorted. General FCN will have residual noise. MDNet can completely remove the noise cake and retain the characteristics of the original image.



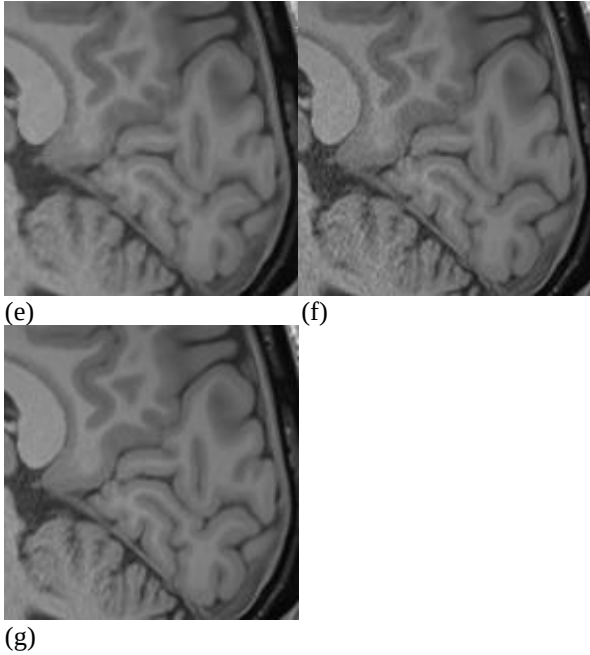


Fig.12 (a)High resolution real MRI T1-weighted brain image (NEX=1) without any additional noise. (b) BM3D denoising results. (c) Simple FCN denoising results. (d) MDNet denoising results. (e) Drawing of partial enlargement by BM3D. (f) Drawing of partial enlargement by simple FCN (g) Drawing of partial enlargement by MDNet

5. Discussion

In the simulation results, we have tested the results of several different methods, which are not very different from each other, and MDNet is not particularly prominent. This means that each model can effectively and completely deal with Gaussian noise. Although MDNet has achieved significant advantages in real images. But we still found two big problems.

The first is that our results are cleaner than the target image (NEX=4). Whether this is also an image distortion must be considered. The second is that our data is a healthy brain image. It remains to be considered whether it is capable of denoising without affecting the imaging of the diseased area when encountering a patient's brain image.

6. Conclusions

In this paper, we proposed a new model called MDNet, which uses upper and lower two-layer network connections, respectively connected to the front and back DnCNN and general DnCNN, to enhance the denoising performance. In addition, MDNet uses BN and ReLU to enhance learning ability and avoid the gradient explosion and vanishing problems common in deep networks. And solve the problem of small data sets. The experimental results show that MDNet is very competitive in medical image denoising compared with

other methods. In the future, we hope that we could develop a network architecture with better performance on this basis.

In addition, as mentioned earlier, we chose MSE rather than SSIM in the setting and selection of loss function. We would continue to try to find the balance weights of the three sub-functions of SSIM in the future, and look forward to making better models to remove the MR Image background noise.

7. References

- [1] Li, S., Yin, H., Fang, L.: 'Group-sparse representation with dictionary learning for medical image denoising and fusion', *IEEE Trans. Biomed. Eng.*, 2006, 59, (12), pp. 3450–3459
- [2] Zhang, L., Zuo, W.: 'Image restoration: from sparse and low-rank priors to deep priors', *IEEE Signal Process. Mag.*, 2017, 34, (5), pp. 172–179
- [3] Zhang, K., Zuo, W., Chen, Y., et al.: 'Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising', *IEEE Trans. Image Process.*, 2017, 26, (7), pp. 3142–3155
- [4] Malfait, M., Roose, D.: 'Wavelet-based image denoising using a Markov random field a priori model', *IEEE Trans. Image Process.*, 1996, 6, (4), pp. 549–565
- [5] Mairal, J., Bach, F., Ponce, J., et al.: 'Non-local sparse models for image restoration'. *Proc. IEEE Int. Conf. Computer Vision*, 2009, pp. 2272–2279
- [6] Dabov, K., Foi, A., Katkovnik, V., et al.: 'Image denoising by sparse 3-D transform-domain collaborative filtering', *IEEE Trans. Image Process.*, 2007, 16, (8), pp. 2080–2095
- [7] Zuo, W., Zhang, L., Song, C., et al.: 'Gradient histogram estimation and preservation for texture enhanced image denoising', *IEEE Trans. Image Process.*, 2014, 23, (6), pp. 2459–2472
- [8] Dong, W., Zhang, L., Shi, G., et al.: 'Nonlocally centralized sparse representation for image restoration', *IEEE Trans. Image Process.*, 2013, 22, (4), pp. 1620–1630
- [9] Beck, A., Teboulle, M.: 'Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems', *IEEE Trans. Image Process.*, 2009, 18, (11), pp. 2419–2434
- [10] Zhu, M., Chan, T.: 'An efficient primal-dual hybrid gradient algorithm for total variation image restoration', UCLA CAM Report, 2008
- [11] Chan, T.F., Chen, K.: 'An optimization-based multilevel algorithm for total variation image denoising', *Multiscale. Model. Simul.*, 2006, 5, (2), pp. 615–645
- [12] Frohn, C., Henn, S., Witsch, K.: 'Nonlinear multigrid methods for total variation image denoising', *Comput. Vis. Sci.*, 2004, 7, (3–4), pp. 199–206
- [13] Gu, S., Zhang, L., Zuo, W., et al.: 'Weighted nuclear norm minimization with application to image denoising'. *Proc. IEEE Conf. Computer Vision Pattern Recognition*, 2014, pp. 2862–2869

- [14] Lefkimmiatis, S.: ‘Universal denoising networks: a novel CNN architecture for image denoising’. *Proc. IEEE Conf. Computer Vision Pattern Recognition*, 2018, pp. 3204–3213
- [15] Zhang, K., Zuo, W., Zhang, L.: ‘FFDNet: toward a fast and flexible solution for CNN based image denoising’, *IEEE Trans. Image Process.*, 2018, pp. 4608–4622
- [16] Lefkimmiatis, S.: ‘Non-local color image denoising with convolutional neural networks’. *Proc. IEEE Conf. Computer Vision Pattern Recognition*, 2017, pp. 3587–3596
- [17] Zhang, K., Zuo, W., Gu, S., et al.: ‘Learning deep CNN denoiser prior for image restoration’. *Proc. IEEE Conf. Computer Vision Pattern Recognition*, 2017, pp. 3587–3596
- [18] Du, B., Xiong, W., Wu, J., et al.: ‘Stacked convolutional denoising auto-encoders for feature representation’, *IEEE Trans Cybern.*, 2017, 47, (4), pp. 1017–1027
- [19] Wu, D., Kim, K., Fakhri, G.E., et al.: ‘A cascaded convolutional neural network for X-ray low-dose CT image denoising’, arXiv preprint arXiv:1705.04267, 2017
- [20] Bako, S., Vogels, T., McWilliams, B., et al.: ‘Kernel-predicting convolutional networks for denoising Monte Carlo renderings’, *ACM Trans. Graph*, 2017, 36, (4), pp. 1–14