Original contribution

Denoising of MR images with Rician noise using a wider neural network and noise range division

Xuexiao You[a,b], Ning Cao[a,*], Hao Lu[a], Minghe Mao[a], Wei Wang[a,c]

[a] School of Computer and Information, Hohai University, Nanjing 210098, China
[b] School of Mathematics and Statistics, Hubei Normal University, Huangshi 435002, China
[c] Key Laboratory of Clinical and Medical Engineering, School of Biomedical Engineering and Informatics, Nanjing Medical University, Nanjing 211166, China

ARTICLE INFO

ABSTRACT

Magnetic resonance (MR) images denoising is important in medical image analysis. Denoising methods based on deep learning have shown great promise and outperform all of the other conventional methods. However, deep-learning methods are limited by the number of training samples. In this article, using a small sample size, we applied a wider denoising neural network to MR images with Rician noise and trained several denoising models. The first model is specific to a certain noise, while the other applies to a wide range of noise levels. We considered the noise range as one interval, two sub-intervals, three sub-intervals, or even more sub-intervals to train the corresponding models. Experimental results demonstrate that for MR images, the proposed deep-learning models are efficient in terms of peak-signal-to-noise ratio, structure-similarity-index metrics and normalized mutual information. In addition, for blind noise, the effect of the three sub-intervals is better than that of the other sub-intervals.

1. Introduction

Magnetic resonance (MR) imaging is a strong medical imaging modality that provides highly detailed images of human tissues and organs, such as the brain [1]. MR images, however, may inevitably be captured along with arbitrary noise during the procurement process, including eddy-current distortions, physiological motion, and in-stabilities of the MR imaging scanning hardware. The noise affects both the precision of clinical diagnosis and facilitation of automatic com-puterized evaluation, including segmentation, classification, and re-gistration. Consequently, removal or lowering of noise is essential for the comprehension and evaluation of MR images. Noise in magnitude MR images can be modeled by a Rician distribution when acquired with single coil.

Numerous denoising techniques are available to filter noise in MR images. For example, Samsonov and Johnson [2] proposed a noise-adaptive nonlinear diffusion strategy to denoise MR images. Krissian and Aja-Fernandez [3] devised a noise-driven anisotropic diffusion fil-tering for MR images that was created from the linear minimum mean-square-error(LMMSE) estimator. In [4], by using wavelet packets, Wood and Johnson studied noise suppression in low signal-to-noise (SNR) MR images. Buades et al. [5] developed a nonlocal means (NLM) filter that utilized the excessive information contained in natural images to eliminate noise. The NLM filter has denoised MR images and proposed many filters based on the NLM filter, for example, a filter based on 3D optimized blockwise version (abbreviated to OBNLM) [6], a filter using an optimal random sampling pattern(SNLM), and an en-hanced NLM filter with pre-processing(PENLM) [8]. Dabov et al. [9] suggested a block-matching three-dimensional (BM3D) filter that groups nonlocal similar patches by collaborative filtering in a trans-formed domain. The BM3D filter has been expended to volumetric data and is designated BM4D for the denoising of 3D MR images [10]. The author of [11] later proposed the prefiltered rotationally invariant NLM (PRI-NLM) technique to exploit the sparseness and self-similarity of 3D MR images. The PRI-NLM method combined the 3D discrete cosine transform (DCT) [12] and an altered NLM filter that was created from a rotationally invariant similarity measure. Zhang et al. [13] leveraged the use and improvement of higher-order singular value decomposition (HOSVD) to denoise MR volume data. Kong et al. [14] utilized the nonlocal similarity property and concurrently characterized primary 3D patches of MR images. They further developed the modified nonlocal tensor-SVD (MNL-tSVD) method, which was supple and quickly altered to a two-stage denoising technique. While robust denoising perfor-mance is usually achievable, the majority of the aforementioned con-ventional approaches involve detailed optimization procedures for images and are therefore time-consuming [15]. In addition, such

---

approaches typically depend upon the optimization of a non-convex cost function and needs manual parameter selection, indicating that they typically require a process for approximating the noise-level parameter.

Not long ago, because of the quick creation of deep neural networks (NNs), numerous novel kinds of NNs have been used in image denoising, e.g., the stacked sparse auto-encoder [16,17], multi-layer perceptron [18], and convolutional denoising autoencoders [19], which have all shown good performance.

More recently, Zhang et al. [20] introduced a deep denoising conventional NN (DnCNN) for Gaussian denoising using residual learning, and achieved promising performance. Following [20], Jifara et al. [21] designed a denoising CNN in a deep framework for medical image denoising. The two deep nets have 17 and 15 layers, respectively. Liu and Fang [22] demonstrated that wider CNNs with increased filter number and size for natural image denoising led to remarkable Gaussian denoising results. Therefore, motivated by the above observations, in this paper, we present a wider denoising NN, i.e., a WDNN, by increasing the filter number for denoising MR images with Rician noise, which is the usual kind of noise that exists in high-magnitude MR images. The WDNN is an extension of a DnCNN with 10 layers.

Moreover, the loss layer, as the efficient driver for the network's learning ability, is based on the MSE for all of the approaches above. Most of the NNs for image-denoising tasks have even been completed by training deep NNs with MSE loss functions; of course, a more general form is the l_2 norm, which can simply generate blur artifacts in denoised images [23]. Zhao et al. [24] showed that, even when the MSE was the appropriate loss, rotating the training loss function with a pertinent loss, including l_1, could lead to the discovery of a better solution for the MSE. Therefore, in this study, the WDNN is trained using l_1 as the loss function.

In the work reported in this paper, a discrete model for a specific Rician noise level, i.e., comprising the aforementioned WDNNs, was trained. In addition, for Rician denoising with an unknown noise level (i.e., blind Rician denoising), a single model, i.e., WDNN − 1, was trained. Moreover, the blind Rician noise range was divided equally or nearly equally into two, three or even four parts to train the corresponding networks, i.e., WDNN − 2, WDNN − 3 and WDNN − 4 separately. In almost all of the literature, a single model has been trained whether it is blind Gaussian noise or blind Rician noise.

The rest of this paper is set up as follows. In Section 2, we describe the method used, and in Section 3 detail the experiments and present the results with relevant discussions. Finally, we conclude the paper in Section 4.

2. Method

In this section, we introduce the investigation of the wider network in two aspects: the number of convolution layer and the number of filters. Moreover, we demonstrate the network architecture for MR image denoising and the loss function employed.

2.1. Network architecture

As can be seen in Fig. 1, the proposed structure is comprised, sequentially, of one convolutional layer with a rectified linear unit (ReLU) [25], 8 convolutional layers with batch normalization (BN) [26] and ReLU, and a one-layer convolutional layer.
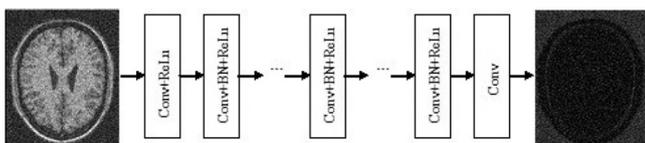

Fig. 1. Architecture of network for MR images denoising.

For the first layer, 192 filters of size $3 \times 3 \times c$ ($c = 1$ for grayscale images and $c = 3$ for color images) are utilized to produce 192 feature maps from a noisy image. ReLUs are then utilized to achieve non-linearity. Here, 3×3 is the convolution height by width. In our application, we deal with MR volume slice by slice.

The next layers, minus the final layer, are of the identical type and have 192 filters of size $3 \times 3 \times 192$ to generate 192 feature maps. BN is added between convolutions as well as a ReLU. For the final layer, one filter of $3 \times 3 \times 192$ convolution size is utilized to rebuild the output. Two key features of the model, BN and residual learning formulation, are incorporated to accelerate training and improve denoising performance.

2.2. Loss function

Letting z be the observed image and x the latent clean image, the network is situated to predict the noise v; in other words, the residual learning formulation is used to train a residual mapping $F(z) \approx v$, and then we have $x = z - F(z)$.

Generally, the loss function in the most common methods adopts an averaged MSE among the coveted residual images and those approximated from noise input:

$$L_{MSE}(\theta) = \frac{1}{2N} \sum_{i=1}^{N} \|R(x_i, \theta) - (z_i - x_i)\|_2^2, \tag{1}$$

where $\{(z_i, x_i)\}_{i=1}^{N}$ represents N noisy-clean training-image (patch) pairs, and $R(\cdot)$ is the estimated residual image determined by the network.

Using MSE loss, however, can produce blur artifacts, so the exact network must be trained utilizing l_1:

$$L_{l_1}(\theta) = \frac{1}{2N} \sum_{i=1}^{N} |R(x_i, \theta) - (z_i - x_i)|, \tag{2}$$

The two losses weigh errors differently: L_{l_1} does not over-penalize larger errors, and thus, consequently, various convergence properties may manifest [24].

3. Experiments

To start with, we investigate the width of the proposed network through experiments, and secondly we present the detailed setups for the proposed network. Finally, the denoising performance of the proposed methods was evaluated.

3.1. Experimental settings

Simulated data were taken from the Brain Web Database [27], which is widely used in the literature. In the following experiments, two images were used: T1-weighted (T1w) MR image and T2w MR image. Each of these samples contained $181 \times 217 \times 181$ voxels with $1 - mm^3$ voxel resolution.

For Rician denoising with known or unknown noise levels, we dealt with the MR volume slice by slice and used 40 images, 30 of which were adopted for training the NN and 10 for estimating the detailed techniques. Note that all of these images have a spatial resolution of 181×217 pixels. To simulate a realistic image-acquisition procedure, we corrupted every image with different levels of Rician noise (5 % - 21% maximum intensity in 4% steps). Data augmentation (clipping and rotation) were utilized in the training set. In evaluating, we utilized the entire image as input without cropping.

We refer to the proposed WDNN model for Rician denoising with particular noise levels as WDNNs.

To train the WDNN model for blind Rician denoising, we established the range of the noise levels to [0%, 24%]. We refer to our individual model for blind Rician denoising task as WDNN − 1. The interval

$[0\%, 24\%]$ was divided into two equal parts, $[0\%, 12\%]$ and $[12\%, 24\%]$, or three equal parts, $[0\%, 8\%]$, $[8\%, 16\%]$, and $[16\%, 24\%]$, or four equal parts, $[0\%, 6\%]$, $[6\%, 12\%]$, $[12\%, 18\%]$, and $[18\%, 24\%]$. We call these three trained models for blind Rician denoising tasks $WDNN-2$, $WDNN-3$, and $WDNN-4$, respectively.

In [22], it was empirically found that using small-size training samples can assist in eluding boundary artifacts since, instead of utilizing large-size training patches, cutting them to small patches can allow a CNN to "see" additional boundary information. Therefore, the training patch size was set to 21×21 because of the network depth and filter size.

The Adam algorithm [28] was used to optimize the network parameters, with $\alpha = 0.01$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\varepsilon = 1e - 8$. The minibatch size was set to 128 and the models were trained for 50 epochs. The learning rate decreased exponentially from $1e - 2$ to $1e - 4$ for the 50 epochs. The initial weights of the network were set by the technique suggested in [29], which has been revealed to be better than random initialization when utilizing a non-linear ReLU as the activation function. All of the experiments were implemented in the MatLab (version R2017b, MathWorks, USA) environment with the MatConvNet package [30].

3.2. Quantitative and qualitative evaluations

We compared the proposed network's quantitative and qualitative performance with other well-established MR denoising methods, including OBNLM [6], PRI-MLM3D [11], and MNL-tSVD [14] and. The implementations were all from the publicly available codes provided by the authors. Several quantitative measures, PSNR (peak SNR ratio), SSIM(structural similarity index), and NMI (Normalized mutual information) were used. The PSNR and SSIM are defined as follows:

$$PSNR = 20\log_{10}\frac{255}{RMSE}, \tag{3}$$

where RMSE is the root mean-square error between denoised data and noise-free data,

$$SSIM = \frac{(2\mu_x\mu_{\widehat{x}} + c_1)(2\sigma_{x\widehat{x}} + c_2)}{(\mu_x^2 + \mu_{\widehat{x}}^2 + c_1)(\sigma_x^2 + \sigma_{\widehat{x}}^2 + c_2)}, \tag{4}$$

where μ_x and $\mu_{\widehat{x}}$ are the means of the data x and $\widehat{x}$, respectively, c_1 and c_2 are constants, σ_x and $\sigma_{\widehat{x}}$ are the variances, and $\sigma_{x\widehat{x}}$ is the covariance of x and $\widehat{x}$.

The NMI is an expression that measures the similarity between two images. The larger its value, the higher the similarity between the two images. The NMI can be expressed with joint entropy $H(P, Q)$ and entropy $\{H(P), H(Q)\}$ of the two images P and Q as

$$NMI(P, Q) = \frac{H(P) + H(Q)}{H(P, Q)}, \tag{5}$$

3.2.1. The width of the proposed network

For simplicity, CNN with 40 layers and 64 feature maps was referred to as 40L-64F, CNN with 10 layers and 64 feature maps was referred to 10L-64F, CNN with 10 layers and 128, 192 and 256 feature maps were referred to 10L-128F, 10L-192F, 10L-256F, respectively. In order to find the suitable parameters for the optimal performance of the proposed network, we compare the denoising effections of several networks in the same experimental setting. We used PSNR as the criterion for determining the best number of feature maps. All comparing experiments are performed on T1w images and the results are shown in Table 1 and Fig. 2.

As can be seen in Table 1 and Fig. 2, the performance of 10 layers with 192 feature maps is better than other networks. The performance of the 10 layers with 64 feature maps is even better than that of the 40 layers with 64. The performance of the 192 feature maps is very close to 256 at the same number of layers. So, "Deeper is better" is not

Table 1
PSNR value comparisons on T1w image of different number of layers and different number of filters for each layers.

|	5%	9%	13%	17%	21%
10L-64F	37.4259	33.7264	31.2521	29.4957	28.0088
10L-128F	37.5372	33.8356	31.3604	29.6167	28.1039
10L-192F	37.5686	33.8544	31.3949	29.6445	28.2347
10L-256F	37.5673	33.8330	31.4062	29.6216	28.2016
40L-64F	36.9170	33.0229	31.0300	29.2685	28.2200
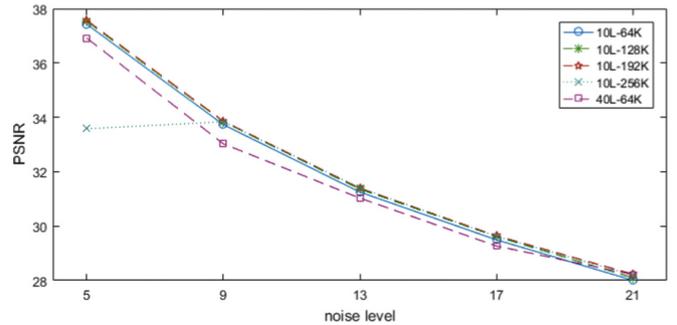
Fig. 2. PSNR comparison of 10L-64F, 10L-128F,10L-192F,10L-256F, and 40L-64F for T1w images under noise levels varying from 5% to 21% in 4% steps.

necessarily correct under certain conditions, and the performance of the wide and shallow networks may be better than the thin and deep ones. The reason for this may be that the performance of the network is not only determined by the network structure, but also other factors, such as optimization algorithms.

Furthermore, Table 1 and Fig. 2 both show that for the noise of the removed image, the performance of the number of the feature maps, 128, 192 and 256, is very close. However, the training time of the 256 feature maps is twice that of 128 and also much longer than that of 192. Therefore, in this paper, 192 is used as the number of feature maps.

3.2.2. Comparison with PSNR, SSIM and NMI

The average PSNRs and SSIMs for various strategies and various levels of noise (5%,9%, 13%, 17%, 21%) of Rician noise on T1w images are shown in Figs.3 and 4, respectively. The original records are shown in Table 2 and Table 3 for better discrimination. Tables 2 and 3 list the average values of PSNRs and SSIMs in denoising T1w images utilizing the eight methods under investigation separately. It can be seen that the trained models: WDNNs, $WDNN-1$, $WDNN-2$, $WDNN-3$ and
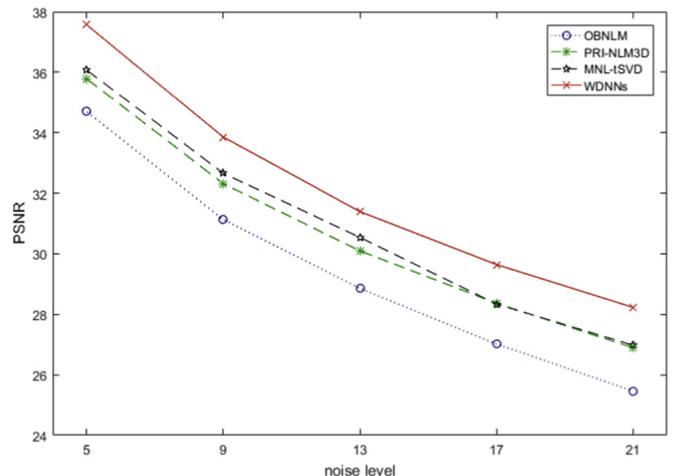
Fig. 3. PSNR comparisons of OBNLM, PRI-NLM3D, MNL-tSVD, and WDNNs for T1w images under noise levels varying from 5% to 21% in 4% steps.
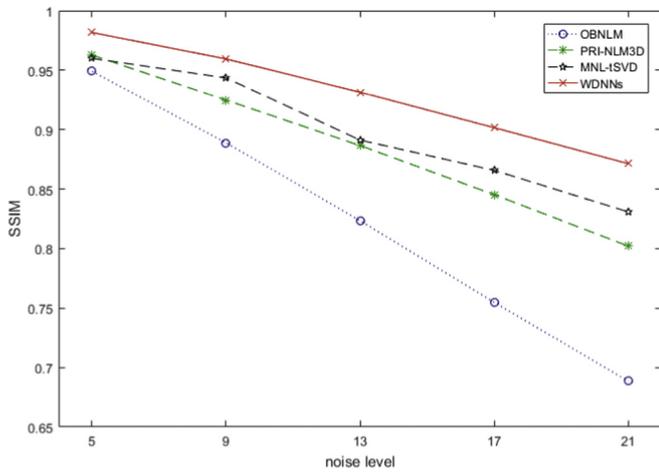
Fig. 4. SSIM comparisons of OBNLM, PRI-NLM3D, MNL-tSVD, and WDNNs for T1w images under noise levels varying from 5% to 21% in 4% steps.

Table 2
PSNR value comparisons of different algorithms on T1w image from Brain web database.

Method	5%	9%	13%	17%	21%
OBNLM	34.6954	31.1267	28.8605	27.0189	25.4648
PRI-NLM3D	35.7826	32.3126	30.0983	28.3665	26.9012
MNL-tSVD	36.0726	32.6568	30.5432	28.3413	26.9827
WDNNs	37.5686	33.8544	31.3949	29.6445	28.2347
WDNN-1	36.6766	33.2647	30.9552	29.2124	27.7628
WDNN-2	37.2457	33.6024	30.9019	29.2566	27.9313
WDNN-3	37.4021	33.4701	31.1860	29.3042	27.9756
WDNN-4	37.3721	33.3825	31.0580	29.3263	27.9996

Table 3
SSIM value comparisons of different algorithms on T1w image from Brain Web Database.

Method	5%	9%	13%	17%	21%
OBNLM	0.9491	0.8889	0.8235	0.7549	0.6885
PRI-NLM3D	0.9628	0.9248	0.8866	0.8453	0.8022
MNL-tSVD	0.9600	0.9435	0.8912	0.8657	0.8309
WDNNs	0.9817	0.9595	0.9315	0.9016	0.8714
WDNN-1	0.9770	0.9519	0.9124	0.8781	0.8535
WDNN-2	0.9799	0.9555	0.9227	0.8947	0.8596
WDNN-3	0.9809	0.9549	0.9275	0.8692	0.8364
WDNN-4	0.9801	0.9544	0.9258	0.8973	0.8680

WDNN − 4, can achieve better PSNR and SSIM results than the competing methods. Especially, in Figs. 3 and 4, we can see that WDNNs trained with a certain noise level significantly outperform other methods such as OBNLM, PRI-MLM3D and MNL-tSVD in terms of PSNR and SSIM. The five models outperform the latest MNL-tSVD and improve the range of PSNR from 0.5 to 1.5 dB and SSIM from 0.01 to 0.04. These experimental results once again demonstrate the advantages of deep learning.

In order to show that our approach and its calibration are not specific to T1w images, T2w images were used. Figs. 5 and 6 show the denoised results of the four methods OBNLM, PRI-MLM3D, MNL-tSVD and WDNNs for T2w noisy images. Tables 4 and 5 present the original records of the experimental results. From Figs. 5 and 6, the PSNR and SSIM of the WDNNs model significantly superior to OBNLM, PRI-MLM3D, MNL-tSVD. The experimental results show that WDNNs is also effective for removing the noise of the T2w image.

In addition, WDNN − 1, WDNN − 2, WDNN − 3 and WDNN − 4 are four different types of networks for blind Rician denoising. From Figs. 7 and 8, it can be seen that the WDNN − 2, WDNN − 3, and WDNN − 4
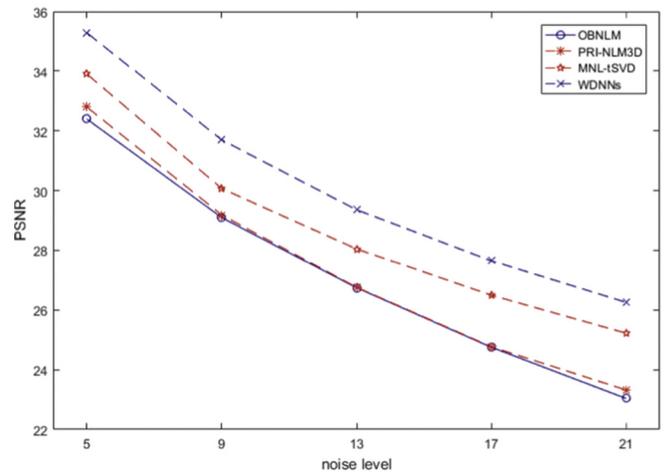

Fig. 5. PSNR comparisons of OBNLM, PRI-NLM3D, MNL-tSVD, WDNNs for T2w images under noise levels varying from 5% to 21% in 4% steps.
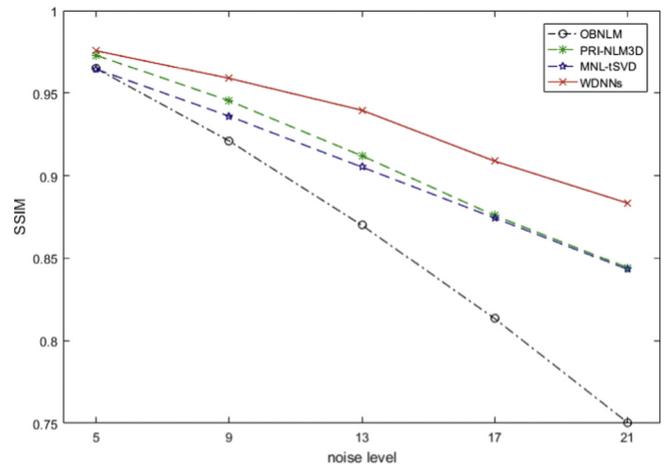

Fig. 6. SSIM comparisons of OBNLM, PRI-NLM3D, MNL-tSVD, WDNNs for T2w images under noise levels varying from 5% to 21% in 4% steps.

Table 4
PSNR value comparisons of different algorithms on T2w image from Brain Web Database.

Method	5%	9%	13%	17%	21%
OBNLM	32.3956	29.0966	26.7444	24.7522	23.0364
PRI-NLM3D	32.7947	29.1833	26.7579	24.7685	23.3204
MNL-tSVD	33.9005	30.0683	28.0468	26.4945	25.2211
WDNNs	35.2875	31.6958	29.3591	27.6484	26.2534

Table 5
SSIM value comparisons of different algorithms on T2w image from Brain Web Database.

Method	5%	9%	13%	17%	21%
OBNLM	0.9648	0.9210	0.8700	0.8134	0.7505
PRI-NLM3D	0.9729	0.9453	0.9120	0.8759	0.8442
MNL-tSVD	0.9644	0.9360	0.9052	0.8744	0.8432
WDNNs	0.9757	0.9589	0.9396	0.9088	0.8834

models, all three of them outperform the single model WDNN − 1. However, the performance of these three models have slightly worse than WDNNs, and the reason is obvious.

From Figs. 7 and 8, we can also see that the effects of image denoising in the sub-interval of the complete interval are better than that
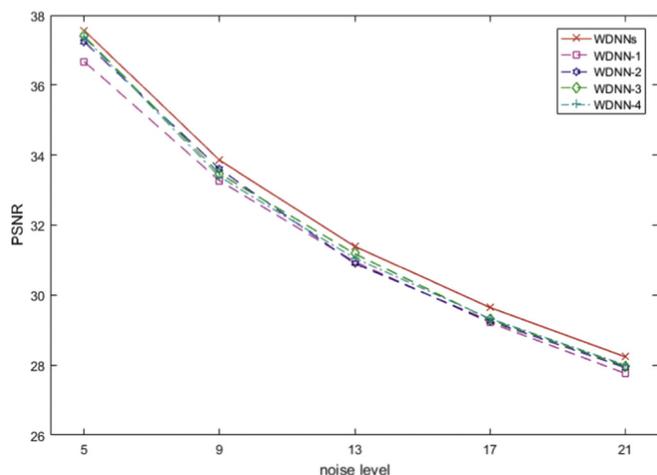
Fig. 7. PSNR comparisons of WDNNs, WDNN − 1, WDNN − 2, and WDNN − 3 for T1w images under noise levels varying from 5% to 21% in 4% steps.
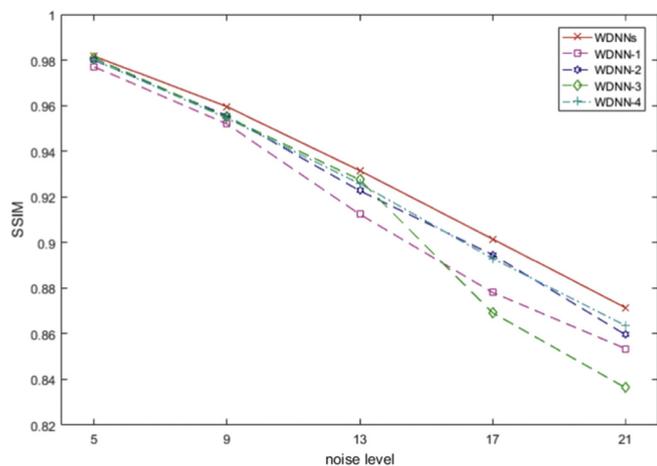

Fig. 8. SSIM comparisons of WDNNs, WDNN − 1, WDNN − 2, and WDNN − 3 for T1w images under noise levels varying from 5% to 21% in 4% steps.

in the whole interval. The reason is that the interval is smaller, and the training model can be more focused. However, according to the experimental results, the interval is not divided as thin as possible. Moreover, it seems that the effect of dividing the interval into three parts equally or approximately is better.

Fig. 9 provides visual comparisons of the WDNNs, WDNN − 1, WDNN − 2, and WDNN − 3 denoising results for T1w data with a 17% noise level. It can be seen in the figure that the reconstructed image is virtually identical to the initial one. After visual inspection, it can be deduced that the outcome of the deep-learning technique is more improved than the others in fine-structure retention and edge and texture reconstruction.

Table 6 shows a comparison of the experimental results in NMI value on T1w image. NMI represents the correlation between the clean image and the denoised image. It should be noted that the WDNNs method yielded better results.

4. Conclusion

In this paper, an effective deep and wide CNN was used to deal with Rician-noise-perturbed MR images. This network is a feed-forward neural network that integrates residual learning and BN to accelerate the training procedure and boost denoising presentation. Our experimental outcomes on a little training dataset show that the performance of the proposed denoising models, WDNN − 1, WDNN − 2, WDNN − 3,
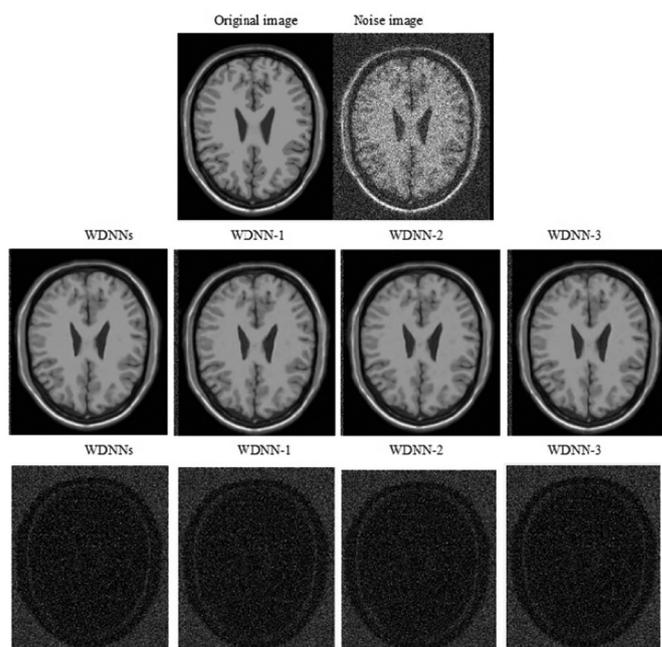

Fig. 9. Denoised results of different algorithms on synthesized T1w brain image. Top row: noise-free T1w image and image with Rician noise level of 17%. Middle row: denoised images with different algorithms. Bottom row: corresponding error images.

Table 6
NMI value comparisons of different algorithms on T1w image from Brain Web Database.

Method	5%	9%	13%	17%	21%
OBNLM	1.2715	1.2232	1.1944	1.1706	1.1532
PRI-NLM3D	1.3002	1.2483	1.2176	1.1947	1.1796
MNL-tSVD	1.2809	1.2302	1.2019	1.1899	1.1632
WDNNs	1.7893	1.7016	1.6319	1.5751	1.5061

and WDNN − 4 is better than that of the existing denoising techniques currently used in T1w MR image denoising. The excellent denoising performance of deep-learning methods is due to two main characteristics. The first is that CNNs with very deep architecture have increased capacity and flexibility for exploiting image characteristics, and the second is that smaller intervals have a better denoising effect than whole interval. We plan to investigate the combination of traditional MR image denoising methods and deep-learning methods in the future research.

References

[1] Mohan J, Krishnaveni V, Guo Y. A survey on the magnetic resonance image denoising methods. Biomed Signal Process Control 2014;9(1):56–69.
[2] Samsonov AA, Johnson CR. Noise-adaptive nonlinear diffusion filtering of MR images with spatially varying noise levels. Magn Reson Med 2004;52(4):798–806.
[3] Krissian K, Aja-Fernàndez S. Noise-driven anisotropic diffusion filtering of MRI. IEEE Trans Image Process 2009;18(10):2265–74.
[4] Wood JC, Johnson KM. Wavelet-packet denoising of magnetic resonance images: importance of Rician statistics at low SNR. Magn Reson Med 1999;41(3):631–5.
[5] Buades A, Coll B, Morel JM. A non-local algorithm for image denoising. Proc IEEE Comput Soc Conf Comput Vis Pattern Recognit 2005;2:60–5.
[6] Coup P, Yger P, Prima S, Hellier P, Kervrann C, Barillot C. An optimized blockwise nonlocal means denoising filter for 3-D magnetic resonance images. IEEE Trans Med Imaging 2008;27(4):425–41.
[8] Liu H, Yang CH, Pan N, Song EM, Green R. Denoising 3D MR images by the enhanced non-local means filter for Rician noise. Magn Reson Imaging 2010;28(10):1485–96.
[9] Dabov K, Foi A, Katkovnik V, Egiazarian K. Image denoising by sparse 3-D transform-domain collaborative filtering. IEEE Trans Image Process 2007;16(8):2080–95.

[10] Maggioni M, Katkovnik V, Egiazarian K, Foi A. Nonlocal transform-domain filter for volumetric data denoising and reconstruction. IEEE Trans Image Process 2013;22(1):119–33.

[11] Manjn JV, Coup P, Buades A, Collins DL, Robles M. New methods for MRI denoising based on sparseness and self-similarity. Med Image Anal 2012;16(1):18–27.

[12] Wu YG, Tai SC, Lin CW. An adaptive 3-D discrete cosine transform coder for medical image compression. IEEE Trans Inf Technol Biomed 2000;4(3):259–63.

[13] Zhang XY, Xu ZB, Jia N, Yang W, Feng QJ, Chen WF, et al. Denoising of 3D magnetic resonance images by using higher-order singular value decomposition. Med Image Anal 2015;19(1):75–86.

[14] Kong ZM, Han L, Liu XL, Yang XW. A new 4-D nonlocal transform-domain filter for 3-D magnetic resonance images denoising. IEEE Trans Med Imaging 2018;37(4):941–54.

[15] Gu SH, Zhang L, Zuo WM, Feng XC. Weighted nuclear norm minimization with application to image denoising. Proc IEEE Conf Comput Vis Pattern Recognit 2014;2014:2862–9.

[16] Vincent P, Larochelle H, Lajoie I, Bengio Y, Manzagol PA. Stacked denoising autoencoders: learning useful representations in a deep network with a local denoising criterion. J Mach Learn Res 2010;11(12):3371–408.

[17] Dolz J, Betrouni N, Quidet M, Kharroubi D, Leroy HA, Reyns N, et al. Stacking denoising auto-encoders in a deep network to segment the brainstem on MRI in brain cancer patients: a clinical study. Comput Med Imaging Graph 2016;52:8–18.

[18] Kwon K, Kim D, Park H. A parallel MR imaging method using multilayer perceptron. Med Phys 2017;44(12):6209–24.

[19] Gondara L. Medical image denoising using convolutional denoising autoencoders Available from: https://arxiv.org/abs/1608.04667; 2016.

[20] Zhang K, Zuo W, Chen Y, Meng D, Zhang L. Beyond a Gaussian denoiser: residual learning of deep CNN for image denoising. IEEE Trans Image Process 2017;26(7):3142–55.

[21] Jifara W, Jiang F, Rho S, Cheng M, Liu S. Medical image denoising using convolutional neural network: a residual learning approach. J Supercomput 2017;75(2):704–18.

[22] Liu P, Fang R. Learning pixel-distribution prior with wider convolution for image denoising Available from: https://arxiv.org/pdf/1707.09135.pdf; 2017.

[23] Wang Z, Bovik AC. Mean squared error: love it or leave it? A new look at signal fidelity measures. IEEE Signal Process Mag 2009;26(1):98–117.

[24] Zhao H, Gallo O, Frosio I, Kautz J. Loss functions for image restoration with neural networks. IEEE Trans Comput Imag 2017;3(1):47–57.

[25] Krizhevsky A, Sutskever I, Hinton G. Imagenet classification with deep convolutional neural networks. Proc Adv Neural Inf Process Syst 2012;2012:1097–105.

[26] Ioffe S, Szegedy C. Batch normalization: Accelerating deep network training by reducing internal covariate shift Available from: https://arxiv.org/abs/1502.03167; 2015.

[27] http://www.bic.mni.mcgill.ca/brainweb/.

[28] Kingma D, Ba J. Adam: A method for stochastic optimization Available from: https://arxiv.org/abs/1412.6980; 2014.

[29] He K, Zhang X, Ren S, Sun J. Delving deep into rectifiers: Surpassing human-level performance on ImageNet classification Available from: https://arxiv.org/abs/1502.01852; 2015.

[30] Vedaldi A, Matconvnet Lenc K. Convolutional neural networks for MATLAB. Proc 23rd ACM Int Conf Multimedia 2012;2012:689–92.