IET Research Journals

Brief Paper

IET Journals
The Institution of Engineering and Technology

ISSN 1751-8644
doi: 0000000000
www.ietdl.org

Adaptive Trainable Nonlinear Reaction Diffusion for Rician Noise Removal

Huan Yang[1] *Hongwei Li*[2,3] *Yuping Duan*[*1]

[1] *Center for Applied Mathematics, Tianjin University, Tianjin, 300072, China*
[2] *School of Mathematical Sciences, Capital Normal University, Beijing, 100048, China*
[3] *Beijing Advanced Innovation Center for Imaging Technology, Capital Normal University, Beijing, 100048, China*
[*] *E-mail: yuping.duan@tju.edu.cn*

Abstract: Rician noise reduction is an essential issue in magnetic resonance imaging. Recently, learning-based methods have achieved great success in dealing with image restoration problems, which can provide fast inference and good performance. One limitation of these methods, however, is that the training procedure is usually noise-level dependent, i.e., the trained models are bound to a specific noise level and cannot automatically adapt to different noise levels. In this paper, we propose a variational model for Rician noise removal by integrating a noise adaption function into the field of experts image prior. Instead of directly solving the energy minimization problem, we unroll the gradient descent step of the energy functional for several iterations, the time-dependent parameters of which can be learned through a supervised training process. Our proposed methodology is robustness against noise level changing and noise distributions. Experimental results over T_1-, T_2- and PD-weighted MRI dataset demonstrate that our proposed model can achieve superior performance compared with other methods in terms of both the peak signal to noise ratio and the structural similarity index.

1 Introduction

Noise pollution is a common problem during the image acquisition and transmission process. It is well-known that signal measured in Magnetic Resonance Imaging (MRI) is complex data. In the single-coil MRI system, since both real and imaginary parts are corrupted by zero mean uncorrelated Gaussian noise with equal variance, the magnitude image follows a stationary Rician distribution. Unlike the additive Gaussian noise, Rician noise is signal-dependent, which makes it more challenging to be removed. Let Ω be an open bounded subset of $\mathbb{R}^n$, $f:\Omega\to\mathbb{R}$ be a given image defined on the domain Ω corrupted by Rician noise, i.e.,

$$f=\sqrt{(u+\eta_1)^2+{\eta_2}^2},$$

where η_1,η_2 represent two zero-mean uncorrelated Gaussian distributions with standard deviation $\sigma>0$, i.e., $\eta_1,\eta_2\sim\mathcal{N}(0,\sigma^2)$. Our goal is to recover a clean image u from a noisy observation f. Thus, the probability density function (PDF) of f follows the Rician distribution for the single-coil MRI data such as

$$P(f|u,\sigma)=\frac{f}{\sigma^2}\exp\Big(-\frac{u^2+f^2}{2\sigma^2}\Big)I_0\Big(\frac{fu}{\sigma^2}\Big),\tag{1}$$

where I_0 is a modified Bessel function of the first kind with order-zero. As demonstrated in [3], Rician distribution is quite suitable for modeling the MRI data. It has been observed that, in high Signal-to-noise ratio (SNR) regions, the Rician distribution can be well approximated by the Gaussian distribution, while in very low SNR regions (i.e., SNR ≈ 0), it reduces to Rayleigh distribution. In general, however, the Rician distribution is neither additive nor multiplicative, which makes Rician noise suppression a challenging task.

The *Maximum a-Posteriori (MAP)* estimation gives the most likely value of u for the given f such that

$$u^*=\arg\max_u \mathrm{P}(u|f).$$

Based on Bayes's theorem, we have

$$\begin{aligned}\max_u \mathrm{P}(u|f)&\Leftrightarrow \max_u \mathrm{P}(f|u)\mathrm{P}(u)\\&\Leftrightarrow \min_u\Big\{-\log(\mathrm{P}(f|u))-\log(\mathrm{P}(u))\Big\}.\end{aligned}\tag{2}$$

We further characterize the image prior according to the Gibbs distribution of image prior model as follows

$$\mathrm{P}(u)=\frac{1}{Z}\exp\big(-\mathcal{R}(u)\big),\tag{3}$$

where $\mathcal{R}(u)$ is a function of the image u, and Z is the so-called partition function to guarantee $\sum\limits_u P(u)=1$. By substituting (1) and (3) into (2) and ignoring the constants, we then have

$$\mathrm{P}(u|f)\propto\exp\Big(-\mathcal{R}(u)-\big(\frac{1}{2\sigma^2}\|u\|_2^2-\big\langle\log I_0(\frac{fu}{\sigma^2}),1\big\rangle\big)\Big),\tag{4}$$

where $\langle\cdot,\cdot\rangle$ denotes the Euclidean inner product. Thus, maximizing the posterior probability $\mathrm{P}(u|f)$ is equivalent to minimizing the following energy functional

$$\begin{aligned}&\min_u\ \ \mathcal{R}(u)+\lambda\mathcal{D}(u,f)\\&\text{with}\ \ \mathcal{D}(u,f)=\frac{1}{2\sigma^2}\|u\|_2^2-\big\langle\log I_0(\frac{fu}{\sigma^2}),1\big\rangle,\end{aligned}\tag{5}$$

where λ is a positive parameter to balance the contributions of the regularization term and data fidelity term.

Many methods have been proposed for Rician noise removal concerning the selection of the regularization term. Choosing the potential function as the magnitude of the discrete gradient at each pixel gives us the popular total variation (TV) prior, which has been widely used for Rician denoising in the literature. Basu *et al.* [1] proposed a Rician denoising model by introducing the Rician likelihood term into the anisotropic diffusion process of Perona-Malik. Later, Getreuer *et al.* [2] generated a variational model using the total variation regularizer and the negative log-likelihood fidelity term.

Liu *et al.* [3] developed a generalized total variation based denoising method and used a local variance estimator method to calculate the spatially adaptive regularization parameters. Li and Zeng [4] proposed a strictly convex TV-based model for Rician noise removal by adding an additional data-fidelity term into the non-convex model. Kang *et al.* [5] incorporated the convex fidelity term with a nonconvex hybrid TV regularization to recover piecewise-smooth images corrupted by Rician noises. Martin *et al.* [6] studied the properties of the TV-based Rician denoising model and employed the proximal point method for the minimization of the difference of convex functions, which demonstrated the effectiveness of the nonconvex data fidelity term over the convex approximation through numerical validation. Kang *et al.* [7] proposed a sparse representation based model associated with the non-convex data-fidelity and a non-convex TV regularizer. Chen *et al.* [8] presented a Rician denoising model based on sparse representation and dictionary learning. Liu *et al.* [9] developed a hybrid regularization model by regarding the MRI data as a combination of true intensity, bias field, and noise followed by Rician distribution. Although there is the mathematical guarantee for the solutions of the aforementioned methods, there are still two major drawbacks: 1) the regularization parameters need to be manually adjusted in these models, which are inefficient and ineffective in practice; 2) it is well-known that the TV regularization favors piecewise constant solution and has the staircase-effect, which leads to unnatural restoration results.

Other methods based on self-similarity and sparsity of images have also been implemented for Rician noise reduction. Coupé *et al.* [10] proposed a 3D optimized blockwise version of the nonlocal means filter for MRI denoising, which was further extended to spatially varying noise. Manjón *et al.* [27] presented new denoising methods based on a 3D moving-window discrete cosine transform hard thresholding and a 3D rotationally invariant version of the nonlocal means filter. The well-known denoising method BM3D has also been used for Rician distributed data through the forward and inverse variance-stabilizing transforms (VST) [11]. The nonlocal methods show superior performance in dealing with images full of texture and structured patterns. However, these methods lack spatial dependency due to the point-to-point estimation. Indeed, medical images are usually highly structured as their pixels exhibit strong spatial dependency.

Roth and Black [12] proposed the fields of experts (FoE) to model the prior probability of an image by using a set of linear filters and a potential function, which achieves great success in low-level image processing tasks via a supervised learning process. Specifically, the parameters of the FoE model can be learned from the training samples by either probabilistic sampling-based algorithm [12] or bilevel optimization method [13]. Interested readers can find more information on the FoE model in the survey paper [14] and the references therein. Despite the high denoising quality, the FoE models suffer from a time-consuming iterative process. Chen *et al.* [15, 16] proposed a trainable nonlinear reaction diffusion (TNRD) model based on the FoE prior, where all parameters including filters and influence functions were learned from training data through a loss-based approach. Instead of employing the iterative scheme adopted in the FoE models, the TNRD model took advantage of the unrolling strategy such that the learning parameters were embedded in a fixed number of iterations, which is highly computationally efficient and well suited for parallel computing on GPUs. Since then, the TNRD model has been successfully applied to different image denoising problems such as Poisson noise removal [17] and speckle reduction [18]. Later, Qiao *et al.* [19] proposed the trainable non-local reaction diffusion models by incorporating the nonlocal self-similarity prior. Feng *et al.* [20] built up a multi-scale pyramid image representation to devise a multi-scale nonlinear diffusion process. All these works demonstrate that discriminative learning methods can outperform traditional denoising methods and provide high computational efficiency.

1.1 Our contributions

In this paper, we propose a simple yet efficient TNRD model to deal with the Rician noise deduction problem with unknown noise levels. We introduce a novel noise adaption function into the FoE regularizer, which is derived based on the prior information on the distribution of the noise levels. The corresponding TNRD model can defeat the general TNRD model trained on the samples with varying noise levels, and approximate the restoration results of the TNRD model trained on the samples with specific noise level without retraining. More specifically, the contributions of our proposed noise adaptive trainable nonlinear reaction diffusion (A-TNRD) model are summarized as follows:

- We propose a novel FoE regularizer by integrating the noise adaption function and derive the trainable nonlinear reaction diffusion process for solving the FoE regularized variational model for Rician noise removal.
- We define the noise adaption function according to the assumption on the distribution of the noise variance. Both uniform distribution and normal distribution are chosen as examples to illustrate the performance of our A-TNRD model.
- The A-TNRD model can handle Racian noises of unknown noise levels and outperform several state-of-the-art methods on the public MRI dataset. Especially, our proposal achieves a significant improvement over the general TNRD model, which is trained on the same dataset with varying noise levels.
- Our proposal is further applied to other image denoising problems, i.e., the additive Gaussian noises and Laplace noises, and achieves consistent results as Rician denoising, which demonstrates the potentials of the adaptive FoE regularizer in real image restoration problems.

1.2 Organization of the paper

The rest of this paper is organized as follows. We review the classical FoE prior regularizer and TNRD process in Section 2. The proposed noise adapted FoE regularizer and A-TNRD model for Rician noise removal are described in Section 3. We present the implementation details of the A-TNRD model in Section 4. The comparison experiments on different MRI modalities and noise levels are conducted on both synthetic and real MRI data in Section 5. Finally, we conclude the paper and discuss possible future works in Section 6.

2 Preliminary knowledge

In this section, we review the FoE image prior in a variational formulation and the effective end to end solver for the FoE regularized model, i.e., the TNRD process.

2.1 The FoE image prior

The FoE prior is a filter based effective higher-order Markov random fields (MRF) image prior, which has been applied to many image processing problems. Unlike traditional methods based on manual adjustment of regularization parameters, the FoE prior model is a discriminative regularization term defined as

$$\mathcal{R}(u) = \sum_{i=1}^{N_k} \sum_{p=1}^{N} \rho_i\big((k_i * u)_p\big), \tag{6}$$

where N_k is the number of filters, N is the number of pixels in the image u, ρ_i is the penalty function, k_i is the linear filter, and $k_i * u$ denotes 2D convolution of the image u with the filter k_i. Both the potential functions ρ_i and filters $k_i, i = 1, \ldots, N_k$, are prelearned based on the training dataset, which can be realized by the loss-specific bilevel optimization scheme [23]. Such an FoE prior model has achieved great success in dealing with image restoration problems, such as image denoising, deblurring, inpainting, and

super-resolution, etc. By combining the FoE prior with suitable data fidelity term, we obtain the following minimization problem

$$\min_u \; E(u) = \sum_{i=1}^{N_k} \sum_{p=1}^{N} \rho_i\big((k_i * u)_p\big) + \lambda \mathcal{D}(u, f). \tag{7}$$

Although both the kernel functions and the influence functions in (7) can be learned from the training data, it still requires to solve a complex optimization problem in the testing stage, which comprises its practical usability.

2.2 The TNRD process

In order to improve the efficiency of the FoE prior model (7), Chen and Pock [15, 16] proposed to solve the following nonlinear diffusion equation

$$\begin{aligned} \frac{u_t - u_{t-1}}{\Delta t} &= -\nabla E(u_{t-1}) \\ &= \underbrace{-\sum_{i=1}^{N_k} \bar{k}_i^t * \phi_i(k_i^t * u_{t-1})}_{\text{difussion term}} \underbrace{- \psi^t(u_{t-1}, f)}_{\text{reaction term}}, \end{aligned} \tag{8}$$

where $\phi_i(\cdot) = \rho_i'(\cdot)$ is the influence function, $\bar{k}_i^t$ is obtained by rotating the filter k_i with 180 degrees, and Δt is the time step size. The reaction term is obtained from the gradient of the data fidelity term, i.e., $\psi(u, f) = \lambda \nabla_u \mathcal{D}(u, f)$. Instead of making use of the chosen filters and influence functions, one can use the unrolling strategy to generate a nonlinear diffusion process to learn the time-dependent parameters $\Theta^t = \{\lambda^t, k_i^t, \phi_i^t\}$. By truncating the diffusion process (8) after T stages, we obtain an iterative system with respect to the variable u as follow

$$\begin{aligned} &u_t = u_{t-1} - \Big(\sum_{i=1}^{N_k} \bar{k}_i^t * \phi_i^t(k_i^t * u_{t-1}) + \psi^t(u_{t-1}, f)\Big), \\ &\text{with } \; t = 1, 2 \dots T. \end{aligned} \tag{9}$$

In (9), the initial value of u is given as $u_0 = f$, and the time step size in equation (8) is set as $\Delta t = 1$, because the formula (8) on the right side can be freely scaled according to the actual needs.

The parameter set $\boldsymbol{\Theta} = \{\Theta^t\}_{t=0}^{t=T-1}$ can be learned through a supervised training process by minimizing a loss function measuring the distance between the estimated image and the ground truth. Similarly, the training framework of the reaction-diffusion model is realized by the bilevel optimization scheme as follows

$$\begin{cases} \min\limits_{\boldsymbol{\Theta}} \mathcal{L}(\boldsymbol{\Theta}) := \sum\limits_{s=1}^{N_s} \ell(u_T^s, u_{gt}^s) = \sum\limits_{s=1}^{N_s} \dfrac{1}{2} \|u_T^s - u_{gt}^s\|_2^2, \\ \text{s.t.} \begin{cases} u_0^s = f_0^s, \\ u_t^s = u_{t-1}^s - \Big(\sum_{i=1}^{N_k} \bar{k}_i^t * \phi_i^t(k_i^t * u_{t-1}^s) + \psi(u_{t-1}^s, f^s)\Big), \\ t = 1, 2 \dots T, \end{cases} \end{cases}$$

where N_s is the number of training samples, u_{gt}^s and f^s are the ground truth image and noisy image, respectively.

Followed the work of [16, 24], we parameterize the influence function as a weighted linear combination of Gaussian radial basis functions (RBFs) to approximate influence functions on each stage such as

$$\phi_i(z) = \sum_{j=1}^{M} w_{i,j} \exp\Big(-\frac{(z - \mu_j)^2}{2\gamma^2}\Big),$$

where M denotes the total number of the RBFs and $w_{i,j} = (w_{i,1}, w_{i,2} \cdots w_{i,M}) \in \mathbb{R}^M$ is the weights for the ith influence function, $i = 1, \dots, N_k$. As the RBFs with equidistant centers μ_j and unified scalar γ are adopted and fixed in the TNRD model, the weights $w_{i,j}$ are the only parameters that need to be learned for generating the influence functions.

As done in [16], we define the linear kernels k_i as a linear combination of Discrete Cosine Transform (DCT) basis kernels $\mathcal{B} \in \mathbb{R}^{R \times (R-1)}$ with zero mean to get rid of the possible scaling problem, i.e.,

$$k_i = \mathcal{B} \frac{c_i}{\|c_i\|_2},$$

where $k_i \in \mathbb{R}^R$, $R = r \times r$ is the filter size for $i = 1, \dots, N_k$. Thus, learning filters k_i are to estimate the parameters $c_i \in \mathbb{R}^{R-1}$. Consequently, we need to learn the parameters $\Theta^t = \{\lambda^t, c_i^t, w_{i,j}^t\}$ for $t = 1, \dots, T$, through he training process.

In this work, we use the joint training scheme to optimize the parameters of all stages simultaneously, which has shown to be more reliable than the greedy training [16]. Such bilevel optimization can be solved by the Limited-memory Broyden-Fletcher-Goldfarb-Shanno (L-BFGS) quasi-Newton's method [25]. It can be observed that the TNRD model is closely related to Convolutional Neural Networks (CNNs) based denoising methods. We simply conclude the relationship among the FoE model, TNRD model, and CNN model as follows:

1) If the parameters $\{\Theta^t\}_{t=0}^{t=T-1}$ are fixed across all the stages, the TNRD process deduces to the FoE prior regularized model for image restoration problem. The merit of the TNRD model is that the time-varying parameters can be simultaneously trained in a supervised way.
2) The TNRD process can be cast into recurrent neural networks, which uses the trained nonlinear activation function instead of fixed ones such as ReLU functions or sigmoid functions. Another difference is that the TNRD model contains a reaction term, which can describe the interrelationship between the observed data and the recovered data.

Although the TNRD model can achieve state-of-the-art performance for Rician denoising problems, there is one drawback that still needs to be addressed, i.e., the trained diffusion networks only perform well in the way they are trained [16]. For example, the trained model based on the noise level $\sigma = 20$ will break down for a test image with the noise level $\sigma = 40$. Although the users can train a TNRD model on the dataset with different noise levels, the performance of such a general TNRD model usually gives worse results than the engineering denoising methods like VST-BM3D [11].

3 The proposed model

In this section, we propose an adaptive FoE image prior to deal with Rician noise at different noise levels and develop the corresponding noise adaptive TNRD model.

3.1 The adaptive FoE prior and A-TNRD model

We formulate the FoE regularized model for Rician denoising by combining the FoE prior with the data fidelity term derived from the Rician distribution (5). Therefore, the following energy minimization problem is concerned with

$$\min_u \sum_{i=1}^{N_k} \sum_{p=1}^{N} \rho_i\big((k_i * u)_p\big) + \lambda\Big(\frac{1}{2\sigma^2}\|u\|_2^2 - \big\langle \log I_0(\frac{fu}{\sigma^2}), 1\big\rangle\Big). \tag{10}$$

Therefore, we can easily derive a trainable nonlinear diffusion process from the energy functional (10) for the Rician noise reduction

$$u_t = u_{t-1} - \sum_{i=1}^{N_k} \bar{k}_i^t * \phi_i^t(k_i^t * u_{t-1}) - \frac{\lambda^t}{\sigma^2}\Big(u_{t-1} - \frac{I_1(v_{t-1})}{I_0(v_{t-1})} f\Big), \tag{11}$$

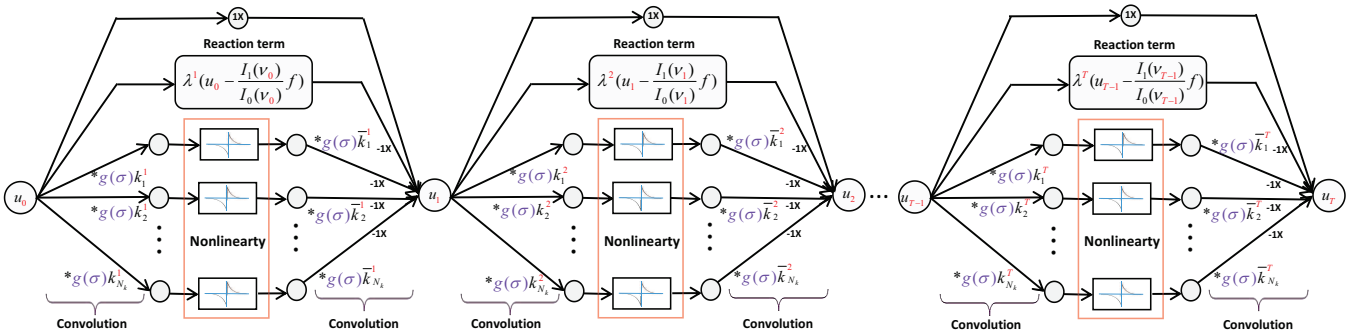


Fig. 1: The architecture of our A-TNRD model with the reaction term derived for Rician noises, each substructure of which corresponds to a diffusion step in (13).

where $v_{t-1} = \frac{fu_{t-1}}{\sigma^2}$, and the derivative of the modified Bessel functions are defined as $I_0'(v) = I_1(v)$, $I_1'(v) = I_0(v) - \frac{1}{v}I_1(v)$ with more details in [4].

Although the regularization parameter λ is affected by the noise level σ, the diffusion process (11) still loses its effectiveness when the noise level is different from the training process. Please refer to Table 2 in Section 5 for the results of applying TNRD_{30} trained on samples with noise level $\sigma = 30$ to images corrupted by Rician noise of $\sigma \in [15, 45]$, which coincide well with the above argument. In fact, the FoE prior term plays a more important role than the data fidelity term in the TNRD model. Thus, we propose a novel TNRD model with the noise level dependent FoE image prior. More specifically, we introduce a continuous function $g(\sigma)$ into the FoE prior such as

$$\mathcal{R}_\sigma(u) = \sum_{i=1}^{N_k}\sum_{p=1}^{N} \rho_i\Big(\big(g(\sigma)(k_i * u)\big)_p\Big), \tag{12}$$

where $g(\sigma)$ is called the noise adaption function used to depict the noise level. Based on the adaptive FoE image prior (12), we modify the TNRD model for Rician noise removal as follows

$$\begin{aligned} u_t = u_{t-1} - \Big(&\sum_{i=1}^{N_k} g(\sigma)\bar{k}_i^t * \phi_i^t\big(g(\sigma)(k_i^t * u_{t-1})\big) \\ &+ \lambda^t\big(u_{t-1} - \frac{I_1(v_{t-1})}{I_0(v_{t-1})}f\big)\Big). \end{aligned} \tag{13}$$

Note that the factor $1/\sigma^2$ in the reaction term is scaled by the noise adaptive function $g(\sigma)$. In this way, both the parameters of the filters and the influence functions will be affected by the noise adaption function $g(\sigma)$. We call the model (13) the adaptive trainable nonlinear reaction diffusion (A-TNRD). Such noise adapted FoE prior is not limited to the Rician denoising, which can be applied to other noise removal problems and image processing tasks.

Similarly, the diffusion model (13) can be interpreted as running the gradient descent of the energy function for several iterations, and the parameters of each step are optimized by training through the following process

$$\begin{cases} \min\limits_{\boldsymbol{\Theta}} \mathcal{L}(\boldsymbol{\Theta}) = \sum\limits_{s=1}^{N_s} \ell(u_T^s, u_{gt}^s) = \dfrac{1}{2}\sum\limits_{s=1}^{N_s}\|u_T^s - u_{gt}^s\|_2^2, \\ \text{s.t.} \begin{cases} u_0^s = f_0^s, \\ u_t^s = u_{t-1}^s - \Big(\sum\limits_{i=1}^{N_k} g(\sigma)\bar{k}_i^t * \phi_i^t\big(g(\sigma)(k_i^t * u_{t-1}^s)\big) \\ \qquad\qquad + \lambda^t\big(u_{t-1}^s - \frac{I_1(v_{t-1}^s)}{I_0(v_{t-1}^s)}f^s\big)\Big), \\ \text{where } t = 1 \ldots T. \end{cases} \end{cases} \tag{14}$$

The training process of our proposed model is shown in Fig. 1, which is a typical feed-forward network.

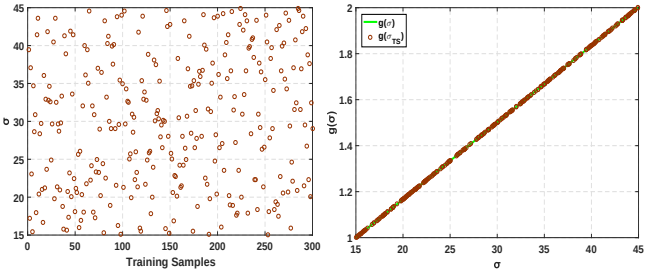
3.2 *The choices of noise adaption function* $g(\sigma)$

An important issue of our A-TNRD model is how to define the noise adaption function $g(\sigma)$, which can be derived based on the prior knowledge of the noise level. Two typical priors are considered in our implementations, i.e., the noise levels followed uniform distribution and normal distribution, respectively.

3.2.1 Uniform distribution: We suppose the noise levels of the training samples follow the uniform distribution by generating a set of σ satisfying the uniform distribution with $\sigma_{\min}$ and $\sigma_{\max}$, i.e., $\sigma \sim U(\sigma_{\min}, \sigma_{\max})$, which are then applied to degrade the clean images. Obviously, the cumulative distribution function can completely describe the probability of the random variable taking any given value. Therefore, we define the noise adaption function $g(\sigma)$ as a continuous function using the cumulative distribution function as follows

$$g(\sigma) = \frac{\sigma - \sigma_{\min}}{\sigma_{\max} - \sigma_{\min}} + 1\ , \tag{15}$$

where we plus one to transform the range of function values to $[1, 2]$ for better training of the TNRD model.



(a) σ of training samples (b) $g(\sigma)$

Fig. 2: Illustration of the distribution of σ in training samples and the function $g(\sigma)$ used in our $\text{A-TNRD}_{\text{UD}}$ model, where σ_{TS} denotes the noise level σ of training samples.

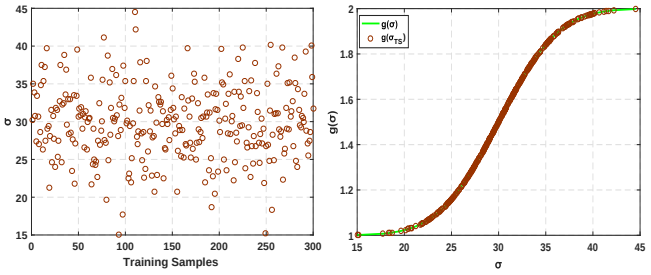
As shown in Fig. 2, we display the standard deviation used to generate the noisy images in the training stage with $\sigma \sim U(15, 45)$ and the corresponding $g(\sigma)$ used in our adaptive TNRD model. In the testing stage, for a given image, we adopt the Rician noise estimation method in [29] to evaluate the noise level σ, and then take the σ into the well-trained A-TNRD model to restore the clean image. With the uniform distribution prior, as long as the noise levels of the test images are in between $\sigma_{\min}$ and $\sigma_{\max}$, our $\text{A-TNRD}_{\text{UD}}$ model

can automatically adapt with different images degraded by Rician noises.

3.2.2 Normal distribution: We can also generate another training dataset with noise levels followed the normal distribution $\sigma \sim N(\mu, s^2)$. Similarly, we define $g(\sigma)$ based on the cumulative distribution function as follows

$$g(\sigma) = \frac{1}{s\sqrt{2\pi}} \int_{-\infty}^{\sigma} e^{-\frac{1}{2}(\frac{x-\mu}{s})^2} dx + 1 \,. \tag{16}$$

It is easy to check that, when we have $\sigma \sim N(30, 25)$ in the training stage, there are 68% training samples with noise level $\sigma \in [25, 35]$ and 95% of the training samples with noise levels $\sigma \in [20, 40]$. Thus, more samples are distributed near the mean value. We display both the standard deviation σ followed $\sigma \sim N(30, 25)$ used in the training stage and the corresponding function $g(\sigma)$ in Fig. 3. In principle, our A-TNRD$_{\rm GD}$ model using the normal distribution prior can handle noisy images of any noise level. However, due to the lack of training samples in tails, the qualities of the restoration results achieved by the A-TNRD$_{\rm GD}$ model may drop for noisy images with σ far from the mean of the Gaussian distribution.



(a) σ of training samples (b) $g(\sigma)$

Fig. 3: Illustration of the distribution of σ in training samples and the influence function $g(\sigma)$ of our A-TNRD$_{\rm GD}$ model, where $\sigma_{\rm TS}$ denotes the noise level σ of training samples.

3.3 Computing the gradients

We aim to minimize the upper-level objective function in (14) with respect to the parameters $\Theta^t = \{\lambda^t, c_i^t, w_{i,j}^t\}$ for $t = 1, \ldots, T$. As mentioned before, we use the L-BFGS quasi-Newton's method to learn these parametes, i.e.,

$$\Theta^{t+1} = \Theta^t - \delta_t \mathcal{P}^t \frac{\partial \ell(u_T, u_{gt})}{\partial \Theta^t}, \tag{17}$$

where δ_t is the step size and $\mathcal{P}^t$ is the approximation of the inverse of Hessian matrix obtained by the second derivatives of ℓ with respect to Θ^t. Because both δ_t and $\mathcal{P}^t$ can be calculated by the L-BFGS algorithm, we only have to compute the gradient of the loss function with respect to Θ^t explicitly, which can be obtained by the chain derivative rule as follows

$$\frac{\partial \ell(u_T, u_{gt})}{\partial \Theta^t} = \frac{\partial \ell(u_T, u_{gt})}{\partial u_t} \cdot \frac{\partial u_t}{\partial \Theta^t}. \tag{18}$$

It should be noted that we only consider the case with one training sample here for brevity, as the gradient of the overall loss function can be obtained by summing up the above gradient over all training samples. The standard back-prorogation technique is widely used in neural network based learning models [26], which can be implemented for our T-stage joint minimization problem. More especifically, there is

$$\frac{\partial \ell(u_T, u_{gt})}{\partial u_t} = \frac{\partial u_{t+1}}{\partial u_t} \cdot \frac{\partial u_{t+2}}{\partial u_{t+1}} \cdots \frac{\partial u_T}{\partial u_{T-1}} \cdot \frac{\partial \ell(u_T, u_{gt})}{\partial u_T},$$

the intermediate terms of which can be expressed as

$$\frac{\partial u_{t+1}}{\partial u_t} = (1 - \lambda^{t+1})\mathbf{I} + \lambda^{t+1}\Big(1 - \frac{1}{v_t} z_t - z_t^2\Big)\Big(\frac{f}{\sigma}\Big)^2 - \sum_{i=1}^{N_k} g(\sigma)(K_i^{t+1})^\top \mathbf{\Lambda}_i \Big(g(\sigma)(\bar{K}_i^{t+1})^\top\Big),$$

with $\mathbf{I}$ being the identity, $z_t = \frac{I_1(v_t)}{I_0(v_t)}$, $\mathbf{\Lambda}_i$ being a diagonal matrix defined as $\mathbf{\Lambda}_i = \mathrm{diag}(\phi_i^{t+1'}(x_1), \phi_i^{t+1'}(x_2) \ldots, \phi_i^{t+1'}(x_N))$, $x_p = g(\sigma)K_i^{t+1}(u_t)_p$, $p = 1, 2, \ldots, N$. Besides, $\bar{K}_i^{t+1}$ and $\bar{K}_i^{t+1}$ are the matrix forms of the filters k_i^{t+1} and $\bar{k}_i^{t+1}$, such that $K_i^{t+1}u \Leftrightarrow k_i^{t+1} * u$ and $\bar{K}_i^{t+1}u \Leftrightarrow \bar{k}_i^{t+1} * u$, respectively. Note that we do not need to explicitly construct the matrices K_i and $\bar{K}_i$. Instead, their operations on vectors can be computed by applying the convolution operations k_i and $\bar{k}_i$ with the symmetric boundary condition [16]. Specifically, we first pad the input image symmetrically and then feed it into the diffusion network. After the process, we discard the padding pixels to obtain the final output.

Meanwhile, the derivative of $\ell(u_T, u_{gt})$ w.r.t. u_T can be directly derived from the upper-level minimization problem of (14), which gives

$$\frac{\partial \ell(u_T, u_{gt})}{\partial u_T} = u_T - u_{gt}.$$

On the other hand, the second term on the right hand side of (18) can be derived from the diffusion equation (14). Here, we focus on illustrating the influence of $g(\sigma)$ in the derivation of $\frac{\partial u_t}{\partial \lambda^t}$, $\frac{\partial u_t}{\partial w_{i,j}^t}$ and $\frac{\partial u_t}{\partial c_i^t}$. More details of the derivation of these derivatives can be found in [16].

1) Computing $\frac{\partial u_t}{\partial \lambda^t}$: the derivatives of u_t with respect to λ^t can be straightforward computed according to (13) as follows

$$\frac{\partial u_t}{\partial \lambda^t} = -\left(u_{t-1} - \frac{I_1(v_{t-1})}{I_0(v_{t-1})} f\right)^\top.$$

2) Computing $\frac{\partial u_t}{\partial c_i^t}$: we have

$$\frac{\partial u_t}{\partial c_i^t} = \frac{\partial u_t}{\partial k_i^t} \cdot \frac{\partial k_i^t}{\partial c_i^t},$$

the derivatives on the right hand side of which can be obtained from

$$\frac{\partial k_i^t}{\partial c_i^t} = \frac{1}{\|c_i^t\|_2}\Big(I - \frac{c_i^t}{\|c_i^t\|_2} \cdot \frac{(c_i^t)^\top}{\|c_i^t\|_2}\Big) \cdot \mathcal{B}^\top,$$

and

$$\frac{\partial u_t}{\partial k_i^t} = -\big(g(\sigma)P_{inv}^\top V^\top + g(\sigma)U_{t-1}^\top \Lambda g(\sigma)(\bar{K}_i^t)^\top\big),$$

with U_{t-1} being constructed from u_{t-1} according to $k_i * u_{t-1} \Leftrightarrow U_{t-1}k_i$ and $P_{inv}^\top$ being a linear operation to invert the kernel k.

3) Computing $\frac{\partial u_t}{\partial w_{i,j}^t}$: we can obtain $\frac{\partial u_t}{\partial w_{i,j}^t}$ as

$$\frac{\partial u_t}{\partial w_{i,j}} = -G^\top g(\sigma)(\bar{K}_i^t)^\top$$

with $G = \exp\Big(-\frac{\big(g(\sigma)K_i^t u_{t-1} - \mu\big)^2}{2\gamma^2}\Big)$.

4 Implementation details

4.1 Dataset and evaluation details

We use the Hammersmith dataset for training the parameters and validating the denoising performance of the proposed model, which is a subset of the IXI dataset and can be downloaded from http://brain-development.org/ixi-dataset/. The Hammersmith dataset contains the T_1-, T_2- and PD-weighted MRI data, where different TNRD models are trained for each type of data. To be specific, we randomly selected two-dimensional T_1-weighted MRI images from the Hammersmith dataset and cropped a 150×150 pixel region from each image to generate the training dataset. The process is the same for the T_2- and PD-weighted MRI images.

To validate the effectiveness of the proposed model, we compared it with the well-established Rician denoising algorithms, the parameter settings of which are listed as follows

- LGTV: *Locally Generalized Total Variation model [3]*. We set the exponent parameter γ as $\gamma = 0.98$ for T_1- and PD-weighted MRI data, and $\gamma = 0.99$ for T_2-weighted MRI data for guaranteeing best restoration results. The maximum number of iteration is fixed as 1000 and the error tolerance is fixed as $1e^{-5}$ for all experiments.
- CTV: *Convex Total Variation based model [4]*. We select different values for the regularization parameter γ for different images and different noise levels, the range of which is $\gamma \in [0.01, 0.07]$. The step sizes in the primal-dual algorithm are defined as $\beta = 8/\gamma$, $\tau = 0.015/\gamma$. The maximum number of iteration as 1000 and the error parameter as $1e^{-5}$.
- ODCT: *Oracle-based Discrete Cosine Transform (DCT) filter [27]*. The three-dimensional moving-window DCT hard thresholding is adopted to deal with Rician noises. The only parameter of the ODCT method is the threshold τ in the prefiltering step, which is defined related to the noise level as $\tau = 2.7\sigma$ (σ denotes the noise level) as suggested in [27].
- PRINLM: *Prefiltered Rotationally Invariant NonLocal Means filter [27]*. The PRINLM method applies the three-dimensional rotationally nonlocal mean filter to the preprocessed data obtained by the ODCT model. To get a good compromise between the computational efficiency and denoising performance, the local neighborhood of $3 \times 3 \times 3$ voxels and the searching volume of $7 \times 7 \times 7$ voxels are used in the experiments.
- VST-BM3D: *Variance Stabilizing Transformation-BM3D [11]*. The VST-BM3D method converts the Rician noise into Gaussian noise via the variance-stabilization transformation and implements the BM3D algorithm to remove the noises, which is a parameter-free method.

We use two quantitative measurements to evaluate the denoising performance. One is the following Peak Signal-to-Noise Ratio (PSNR)

$$\mathrm{PSNR} = 20 \log_{10} \frac{255}{\mathrm{MES}},$$

where MES denotes the mean square error between the noise-free image and the denoised image. The other one is the Structural Similarity index (SSIM) [28] defined as

$$\mathrm{SSIM} = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c2)},$$

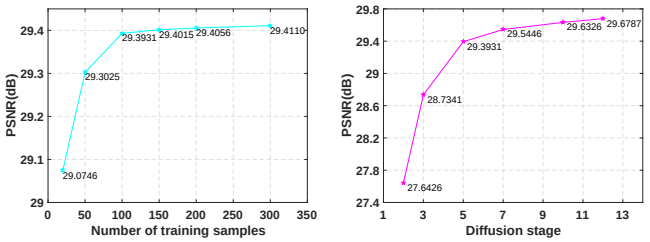
where μ_x and μ_y are the mean of the image x and y, c_1 and c_2 are two constants, σ_x and σ_y are the variances of image x and y, respectively, and σ_{xy} is the covariance of image x and y.

4.2 Model setup

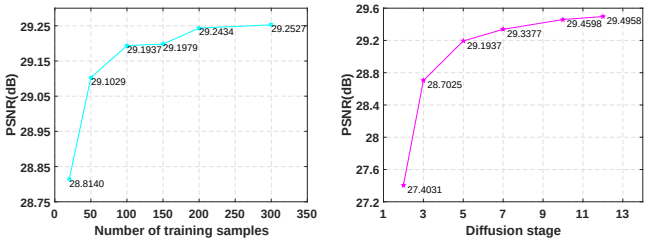
In this subsection, we discuss several issues that affect the performance of our A-TNRD model, including the size of training samples, the number of diffusion steps, and the size of filters. The A-TNRD$_{\mathrm{UN}}$ and A-TNRD$_{\mathrm{ND}}$ represent the A-TNRD model with the noise adaptive function (15) and (16), respectively. We denote the general TNRD model trained on samples with varying noise level $\sigma \in [15, 45]$ as TNRD$_{\mathrm{G}}$, and the TNRD model trained on specific noise level as TNRG$_{\mathrm{S}}$.

4.2.1 The choices of training samples and diffusion steps:

Both the number of training samples and the number of diffusion steps influence the performance of the TNRD model. We track the PSNR of the TNRD$_{\mathrm{S}}$ and our A-TNRD$_{\mathrm{UN}}$ model w.r.t. different training samples from $N_s = 20$ to $N_s = 300$, and different diffusion steps from $T = 2$ to $T = 12$, the results of which are presented in Fig. 4. Note that we evaluate the denoising performance on a testing dataset containing 40 T_1-weighted MRI images degenerated by Rician noise of $\sigma = 30$. Fig. 4 (a) shows the influence of the number of training samples and diffusion stages for the TNRD$_{30}$ model. One can find that 100 images are typically enough to provide satisfactory results. When the number of samples is increased to more than 100, the improvement of the PSNR value is insignificant. Therefore, we set the number of training samples to 100 for the TNRD$_{\mathrm{S}}$ model. And the figure on the right side in Fig. 4 (a) indicates that 10 diffusion stages are suitable for the TNRD$_{\mathrm{S}}$ model to balance the performance and computational time. Similar results of our A-TNRD$_{\mathrm{UN}}$ model are displayed in Fig. 4 (b), which show that 200 training samples and 10 diffusion stages are suitable to provide the best trade-off between efficiency and effectiveness. It is reasonable that more training samples are required for our A-TNRD model since the noises vary adaptively among the samples.



(a) TNRD$_{\mathrm{S}}$



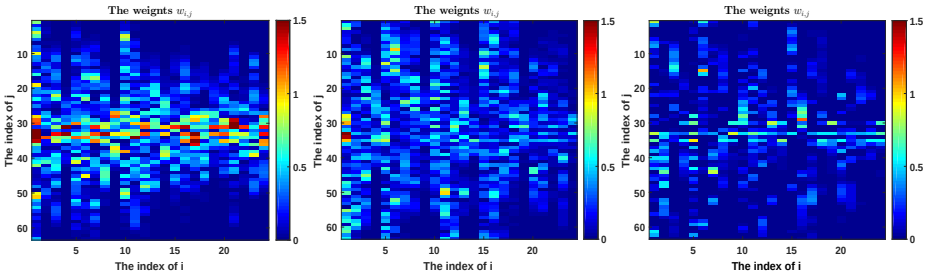
(b) A-TNRD$_{\mathrm{UN}}$

Fig. 4: The influences of the number of training samples and diffusion stages for both TNRD$_{\mathrm{S}}$ model and A-TNRD$_{\mathrm{UD}}$ model.

4.2.2 The choices of the size of filters and Gaussian RBFs:

Another issue is how to select the size of the filter and the number of Gaussian RBFs. In principle, we can exploit the filters of any size, but in practice, to balance the trade-off between computational efficiency and image quality, we use the filters of size 5×5 and set the number of the filters to 24. We adopt 63 Gaussian RBFs with equidistant centers at $[-310 : 10 : 310]$ and set the scaling parameter $\gamma = 10$, which are the same as [16]. Unless otherwise specified, our training models use the same RBFs, filter size, and the number of filters.

Here, we evaluate the differences in between the weights of the influence functions obtained by the TNRD$_{\mathrm{G}}$, A-TNRD$_{\mathrm{UD}}$, and A-TNRD$_{\mathrm{ND}}$ model. The weights $w_{i,j}$ of the RBFs for generating the influence function ϕ_i of the last stage for the three models are displayed in Fig. 5, where the same stage, filter size, and Gaussian RBFs are used in the three TNRD models. As shown, the distributions of the weights obtained by our A-TNRD models are more dispersed



(a) TNRD_{G} model (b) $\text{A-TNRD}_{\text{UD}}$ model (c) $\text{A-TNRD}_{\text{ND}}$ model

Fig. 5: The learned weights $w_{i,j}, i = 1, 2 \cdots, 24\,, j = 1, 2 \cdots, 63$ of the TNRD_{G}, $\text{A-TNRD}_{\text{UD}}$ and $\text{A-TNRD}_{\text{ND}}$ model on the last stage.

than the one obtained on the TNRD_{G} model, especially the weights trained followed the uniform distribution prior. The comparison illustrates that our A-TNRD models use more RBFs to approximate the influence functions to deal with the images corrupted by mixed noise levels, which can somehow explain why our model works better than the TNRD_{G} model.

In summary, the size of parameters $\boldsymbol{\Theta} = \left\{\{\lambda^t, c_i^t, w_{i,j}^t\}\right\}_{t=0}^{t=T-1}$ in our TNRD model are c_i^t (24×24), $w_{i,j}^t$ (24×63) and one λ^t in each stage, which means that our models have 2089 parameters on each stage and 20890 parameters totally. On the other hand, the parameter sets of the CNN models [21, 22] are more than 20 times and 120 times larger compared to our model.

5 Experimental Results

5.1 TNRD models on Rician denoising

In this subsection, we evaluate the advantages of the Rician data fidelity and the adaptive FoE regularizer in dealing with noisy images degenerated by different levels of Rician noise.

5.1.1 Comparison with VST-TNRD model: We compare the TNRD_{S} model with the VST-TNRD model, which converts the Rician noise to Gaussian noise by the variance stabilizing transformation and uses the TNRD model trained for Gaussian noise to remove noise. Note that the VST-TNRD models are trained on noisy images corrupted by a specific level of Gaussian noise as done in [16]. We list both the PSNR and SSIM in Table 1, which are the average values obtained on 40 T_1-weighted MRI images corrupted by Rician noise with noise levels 20, 30 and 40, respectively. As can be observed, the TNRD model with reaction force derived by Rician data fidelity always provides better restoration results than the VST-TNRD model.

Table 1 The average PSNR(dB) and SSIM results on 40 T_1-weighted images from IXI-Hammersmith dataset corrupted by different levels Rician noise.

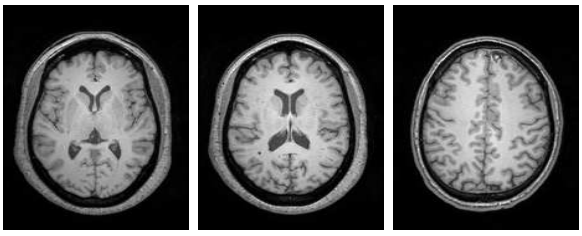
Methods	20		30		40	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
VST-TNRD	31.5612	0.8846	29.2389	0.8486	27.4808	0.8069
TNRD_{S}	**31.8682**	**0.9073**	**29.6326**	**0.8654**	**27.9441**	**0.8321**

5.1.2 The comparison between TNRD models for Rician denoising: To demonstrate the advantage of our adaptive FoE prior in processing images with unknown noise levels, we train four TNRD models for general Rician noise removal problem, i.e., the TNRD_{30} model trained on samples with noise level $\sigma = 30$, the general TNRD_{G} model and the other two A-TNRD models trained on mix noise levels of $\sigma \in [15, 45]$. All models are evaluated on a testing dataset containing 40 T_1-weighted MRI images, which are corrupted by Rician noises with noise level $\sigma = 15$, 18, 21, 24, 27, 30, 33, 36, 39, 42 and 45, respectively.

As shown in Table 2, although the data fidelity term of Rician noise is dependent on the noise level σ, the TNRD_{30} model fails to handle with noise changing. The TNRD_{G} model can achieve better restoration results than TNRD_{30} except for the noise levels $\sigma = 30$, 27, and 24, which illustrate that the general model is more suitable to deal with images of unknown noise levels. Moreover, we observe that our adaptive TNRD model outperforms the TNRD_{G} model such that TNRD_{UD} gives a 2 dB and a 1 dB higher PSNR than the TNRD_{G} model for $\sigma = 15$ and $\sigma = 45$, respectively. The $\text{A-TNRD}_{\text{ND}}$ also outperforms TNRD_{G} with clear superiority. The results demonstrate that our model can learn more efficient priors from the training samples with varying noise levels. Besides, we also observe that the performance of the $\text{A-TNRD}_{\text{ND}}$ model is better than the $\text{A-TNRD}_{\text{UD}}$ model when the noise level is around $\sigma = 30$, and drops its advantages as the noise level becomes far from $\sigma = 30$, i.e., $\sigma < 25$ and $\sigma > 35$. Thus, $g(\sigma)$ with the uniform distribution assumption is more suitable for removing noises with σ varying in a large interval. In the following experiments, we simply use the $\text{A-TNRD}_{\text{UD}}$ model for evaluation unless otherwise specified.

5.2 Comparison with the-state-of-the-art methods

We further evaluate our A-TNRD model by comparing it with other well-known Rician denoising methods on T_1-, T_2- and PD-weighted MRI data.



(a) $\text{T}_1.1$ (b) $\text{T}_1.2$ (c) $\text{T}_1.3$

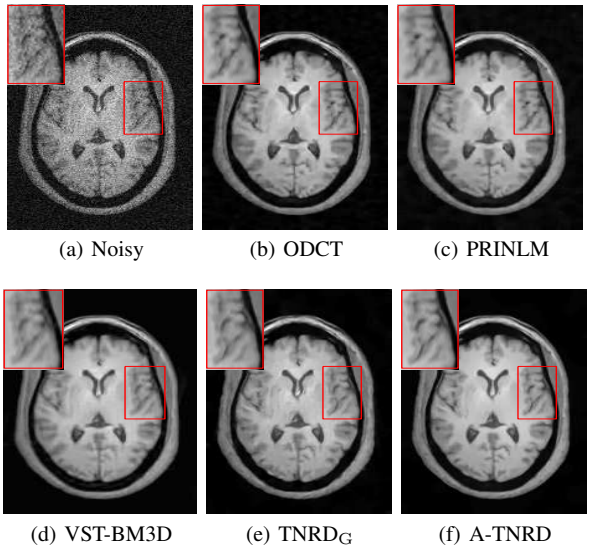
Fig. 6: Three images of the clean T_1-weighted MRI data.

5.2.1 Results on T_1-weighted dataset: In the first place, we apply our proposed adaptive TNRD model as well as other denoising methods on three T_1-weighted brain images as illustrated in Fig. 6 and a three-dimensional T_1-weighted data from the IXI-Hammersmith dataset. Both the PSNR and SSIM are summarized in Table 3, where the best two results are highlighted in red and blue colors, respectively. As shown, the TNRD_{S} model presents the most robust performance compared to all other methods in terms of PSNR and SSIM, followed by our proposed A-TNRD model. Pay attention that three TNRD_{S} models need to be trained to achieve such good restoration results, while our A-TNRD model only needs to be trained once and can deal with images corrupted by different

Table 2 The average PSNR(dB) and SSIM results obtained by the TNRD models on 40 T_1-weighted images from IXI-Hammersmith dataset corrupted by different levels Rician noise, where the best two results are highlighted in red and blue color, respectively.

	Methods	15	18	21	24	27	30	33	36	39	42	45
PSNR	TNRD_{30}	28.5308	30.3736	30.2329	30.1785	30.0334	29.6326	27.9078	25.8152	24.0499	22.5548	21.2520
	TNRD_{G}	30.2859	31.4313	30.4320	30.0732	29.7608	29.2604	28.7732	28.2263	27.3877	26.1288	24.7001
	$\text{A-TNRD}_{\text{UD}}$	32.6792	31.9520	31.2680	30.6270	30.0217	29.4598	28.9274	28.4126	27.8898	27.3128	26.5108
	$\text{A-TNRD}_{\text{ND}}$	32.2579	31.7876	31.2217	30.6263	30.0715	29.5277	28.9638	28.3813	27.6836	26.6988	25.4509
SSIM	TNRD_{30}	0.6608	0.7752	0.8670	0.8640	0.8614	0.8654	0.6524	0.5918	0.5589	0.5335	0.5100
	TNRD_{G}	0.8693	0.8782	0.7425	0.7713	0.8579	0.8532	0.8410	0.8155	0.6948	0.6017	0.5547
	$\text{A-TNRD}_{\text{UD}}$	0.9062	0.9032	0.8956	0.8856	0.8736	0.8599	0.8426	0.8252	0.8091	0.7972	0.7786
	$\text{A-TNRD}_{\text{ND}}$	0.9037	0.9003	0.8950	0.8815	0.8753	0.8668	0.8539	0.8395	0.8198	0.7833	0.7372

noise levels. Actually, it is more reasonable to compare our A-TNRD model with the TNRD_{G} model, which is trained on the same dataset as ours. It can be seen that our A-TNRD model produces much better restoration results than the TNRD_{G} model in terms of PSNR and SSIM. Thus, the A-TNRD model is a better choice for Rician noise removal problems with different noise levels.



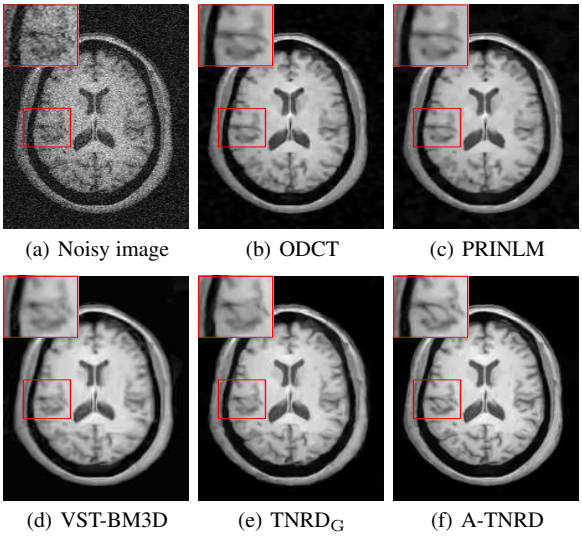
(a) Noisy (b) ODCT (c) PRINLM

(d) VST-BM3D (e) TNRD_{G} (f) A-TNRD

Fig. 7: The noisy image and recovery results obtained by ODCT, PRINLM, VST-BM3D, TNRD_{G} and A-TNRD model on the image $T_1.1$ corrupted by Rician noise of $\sigma = 20$.

Fig. 7 presents the recovery results of image $T_1.1$ with Rician noise $\sigma = 20$, the magnified part of which can illustrate the superiority of learnable methods in recovering edges and structures of the brain image. We show the recovery results of image $T_1.2$ with Rician noise $\sigma = 30$ in Fig. 8. Similarly, our A-TNRD model gives the restoration result with the best visual quality, which provides more details than other methods. The recovered images and the difference images of image $T_1.3$ with Rician noise $\sigma = 40$ are presented in Fig. 9 and Fig. 10, respectively. The difference images demonstrate that our A-TNRD model gives the recovery result with a clean background. Although the performance of the PRINLM and ODCT method seems acceptable in the brain regions, the results of the background reveal that their results are not as homogeneous as ours.

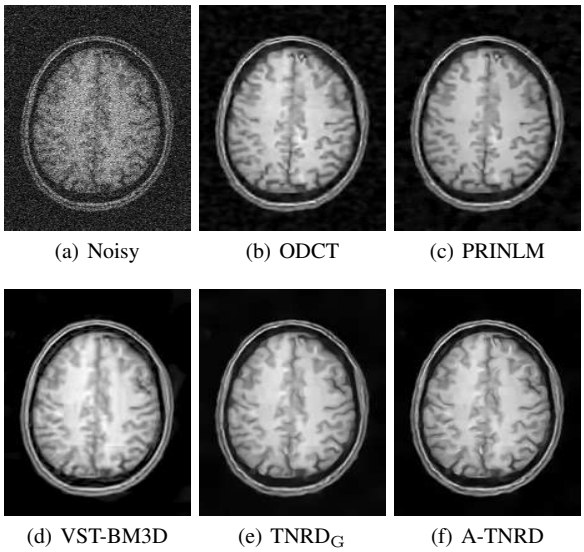
Although our model is a typical 2D approach, we apply the trained model on a 3D T_1-weighted volume data using the slice by slice strategy and compare the results with other comparison algorithms. Table 4 tabulates the PSNR and SSIM obtained by different methods, where our model gives the best restoration results with an average 0.5 dB higher PSNR than the VST-BM3D model. Besides, we make a run time comparison of the comparative algorithms in Table 5. We see that our A-TNRD model is faster than the VST-BM3D and variational methods. Currently, our TNRD model consumes more computational time than the ODCT and PRINLM



(a) Noisy image (b) ODCT (c) PRINLM

(d) VST-BM3D (e) TNRD_{G} (f) A-TNRD

Fig. 8: The noisy image and recovery results obtained by ODCT, PRINLM, VST-BM3D, TNRD_{G} and A-TNRD model on the image $T_1.2$ corrupted by Rician noise of $\sigma = 30$.



(a) Noisy (b) ODCT (c) PRINLM

(d) VST-BM3D (e) TNRD_{G} (f) A-TNRD

Fig. 9: The noisy image and recovery results obtained by ODCT, PRINLM, VST-BM3D, TNRD_{G} and A-TNRD model on the image $T_1.3$ corrupted by Rician noise of $\sigma = 40$.

model, the efficiency of which can be improved by exploring the parallel computing power of GPU devices.

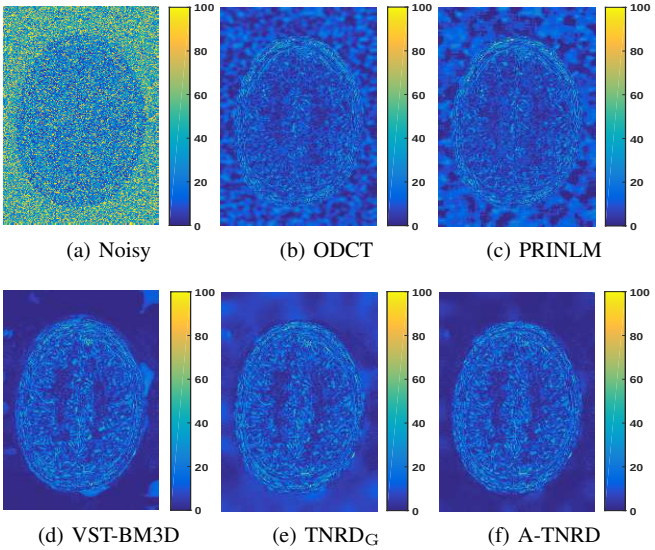
For the T_1-weighted MR images, the regularization parameter γ in the CTV model are set as $\gamma = 0.03, 0.02, 0.018$ for image

Table 3 Denoising results comparison for T_1-weighted MR images corrupted by different levels of Rician noise in terms of PSNR and SSIM, where the best two results are highlighted in red and blue color, respectively.

Images	Methods	20		30		40	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
T_1.1	Original	20.6852	0.4268	17.0899	0.3116	14.5588	0.2317
	LGTV	26.1562	0.6247	24.4429	0.5640	23.5159	0.5330
	CTV	28.1364	0.8173	26.1322	0.7228	25.0478	0.7063
	ODCT	31.1060	0.8079	28.4823	0.7229	26.9545	0.6748
	PRINLM	30.9743	0.8033	28.6256	0.7343	27.3902	0.7287
	VST-BM3D	30.7771	0.8543	28.7359	0.8164	27.3165	0.7785
	$TNRD_S$	31.7257	0.8986	29.5784	0.8550	27.9690	0.8196
	$TNRD_G$	31.1352	0.8314	29.2210	0.8359	26.9769	0.6654
	A-TNRD	31.3988	0.8890	29.5062	0.8530	27.8478	0.8062
T_1.2	Original	20.6460	0.4499	17.0479	0.3415	14.5131	0.2625
	LGTV	25.0421	0.6389	24.0134	0.6042	22.4302	0.5206
	CTV	28.9826	0.8409	26.8260	0.8211	25.1489	0.7768
	ODCT	30.8346	0.7996	28.4587	0.7153	26.9709	0.6697
	PRINLM	30.6762	0.7832	28.4568	0.7187	27.1044	0.7121
	VST-BM3D	30.3998	0.8614	28.3968	0.8276	26.9499	0.7996
	$TNRD_S$	31.3309	0.9072	29.2015	0.8684	27.6095	0.8384
	$TNRD_G$	30.4597	0.7886	28.7612	0.8458	26.6239	0.6718
	A-TNRD	30.9694	0.8973	29.0962	0.8634	27.5031	0.8343
T_1.3	Original	20.5262	0.4297	16.79550	0.3289	14.4316	0.2532
	LGTV	25.9284	0.5818	23.6895	0.5249	21.4891	0.4620
	CTV	27.4578	0.8208	25.9555	0.7833	24.6723	0.7309
	ODCT	31.2044	0.7937	28.6281	0.6976	26.2431	0.6066
	PRINLM	31.0309	0.7801	28.7065	0.7409	26.0620	0.6184
	VST-BM3D	31.1352	0.8314	29.0794	0.7780	27.4743	0.7220
	$TNRD_S$	31.6203	0.9175	29.4363	0.8781	27.8003	0.8488
	$TNRD_G$	30.4221	0.7536	29.0005	0.8765	26.8018	0.6326
	A-TNRD	31.3307	0.9092	29.3327	0.8727	27.6195	0.8258

Table 5 Time consumption (in second) of different algorithms on T_1-weighted MRI experiment.

Methods	LGTV	CTV	ODCT	PRINLM	VST-BM3D	$TNRD_S$	$TNRD_G$	A-TNRD
Single image	4.79	2.20	-	-	0.65	0.39	0.40	0.40
3D volume	1235.54	849.87	43.71	55.48	147.52	113.96	115.72	117.60



(a) Noisy (b) ODCT (c) PRINLM

(d) VST-BM3D (e) $TNRD_G$ (f) A-TNRD

Fig. 10: The absolute difference images between the ground-truth and the restoration of ODCT, PRINLM, VST-BM3D, $TNRD_G$, A-TNRD in Fig. 9.

T_1.1 of noise level $\sigma = 20, 30, 40$; $\gamma = 0.03, 0.025, 0.022$ for image T_1.2 of $\sigma = 20, 30, 40$; $\gamma = 0.03, 0.025, 0.02$ for image T_1.3 of $\sigma = 20, 30, 40$ and $\gamma = 0.03, 0.025, 0.02$ for the 3D T_1-weighted volume of $\sigma = 20, 20, 40$, respectively.

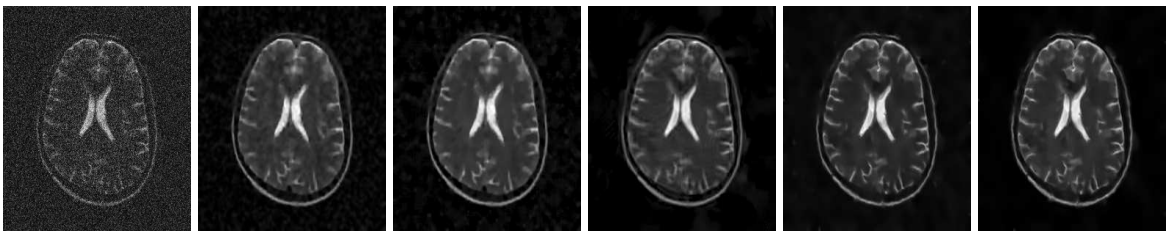
5.2.2 Results on T_2-weighted dataset: Similarly, we train the A-TNRD model on the T_2-weighted MR images corrupted by Rician noises $\sigma \in [15, 45]$ and apply it to a three-dimensional T_2-weighted MRI data corrupted by Rician noise of different noise levels. The

Table 4 The PSRN (dB) and SSIM of different methods on a three-dimensional T_1-weighted MRI data with different levels of Rician noises, where the best two results are highlighted in red and blue color, respectively.

Methods	20		30		40	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
Original	20.3358	0.3203	16.7770	0.2226	14.2561	0.1590
LGTV	26.6416	0.4988	24.7620	0.4352	23.3374	0.3679
CTV	29.2136	0.7473	28.6148	0.7172	25.8081	0.6410
ODCT	31.4025	0.7067	28.8157	0.5863	26.5359	0.4830
PRINLM	30.8705	0.6777	28.6967	0.6207	26.1385	0.4889
VST-BM3D	31.1352	0.8314	29.0794	0.7780	27.4743	0.7220
$TNRD_S$	31.8978	0.8501	29.7162	0.7923	28.1165	0.7521
$TNRD_G$	29.8986	0.6515	29.2570	0.7741	26.9589	0.4975
A-TNRD	31.5864	0.8370	29.7008	0.7887	27.9949	0.7267

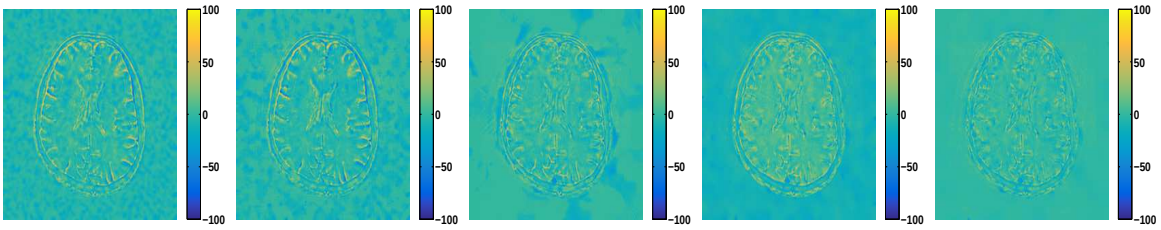
quantitative comparison results are presented in Table 6, where both average PSNR and SSIM of our proposal are better than other methods except for $TNRD_S$. Especially, our A-TNRD model shows a significant advantage over the $TNRD_G$ model, which demonstrates the noise adaption function can significantly improve the denoising performance. Fig. 11 and Fig. 12 provide one slice of restoration images corrupted by Rician noise with $\sigma = 40$ and the difference images between the ground truth and the restored images, respectively. Because Rician noise is signal-dependent, there exists structure information in the difference images for all the methods. As shown, fewer signals are left in the difference image of our A-TNRD model, which can demonstrate the effectiveness of the proposed model in recovering fine structures and details. The comparison between our model and $TNRD_G$ model also certificates the superiority of the noise adaption function in dealing with Rician noises of the unknown noise level.

For the three-dimensional T_2-weighted MR volume, we set the regularization parameter γ in the CTV model as $\gamma = 0.07$, 0.04, 0.03, 0.025, 0.035, 0.035, 0.03 for noise level $\sigma = 15$, 20, 25, 30, 35, 40, 45, respectively.



(a) Noisy image (b) ODCT (c) PRINLM (d) VST-BM3D (e) TNRD_{G} (f) A-TNRD

Fig. 11: The noisy image and recovery results obtained by ODCT, PRINLM, VST-BM3D, TNRD_{G} and A-TNRD on one slice of the T_2-weighted MRI data corrupted by Rician noise of $\sigma = 40$.

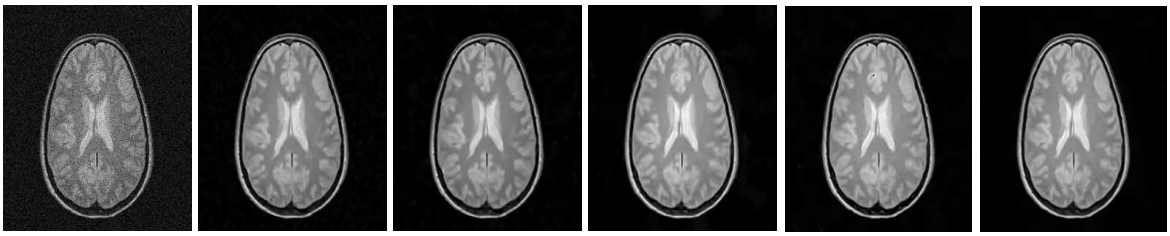


(a) ODCT (b) PRINLM (c) VST-BM3D (d) TNRD_{G} (e) A-TNRD

Fig. 12: The difference images between the ground truth and the restoration of ODCT, PRINLM, VST-BM3D, TNRD_{G}, A-TNRD in Fig. 11.

Table 6 PSRN(dB) and SSIM of different methods on a three-dimensional T_2-weighted MRI data with Rician noise of different levels, where the best two results are highlighted in red and blue color, respectively.

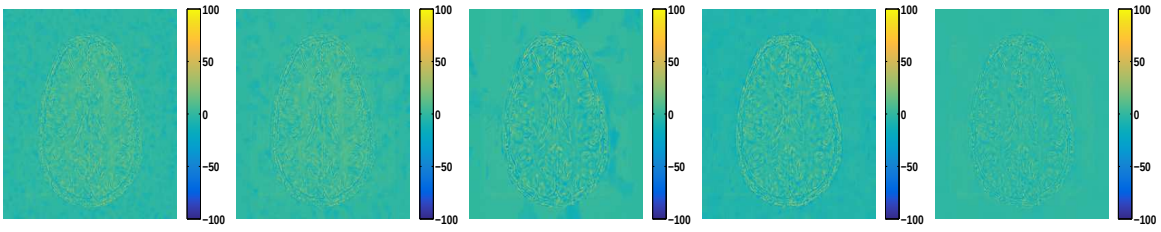
	Methods	15	20	25	30	35	40	45
PSNR	Original	22.8260	20.2510	18.2619	16.6404	15.2702	14.0825	13.0333
	LGTV	28.4066	27.8411	27.3937	26.4809	25.6296	24.8370	24.1310
	CTV	32.0357	31.0110	29.7173	28.6148	27.2634	26.0315	25.7385
	ODCT	34.0581	32.2916	30.4283	29.1546	27.8695	27.2088	26.4645
	PRINLM	33.6890	31.9685	30.0645	28.6351	27.3965	27.1046	26.7606
	VST-BM3D	34.1138	32.4017	31.0543	29.9390	28.9766	28.0942	27.2909
	TNRD_{S}	35.0459	33.5466	32.4142	31.4641	30.5942	29.7400	29.0090
	TNRD_{G}	30.6964	31.0235	31.7659	30.9948	29.1405	26.0617	23.3561
	A-TNRD	34.5974	33.4041	32.3610	31.4374	30.5593	29.6531	28.5642
SSIM	Original	0.3159	0.2510	0.2051	0.1763	0.1436	0.1220	0.1043
	LGTV	0.6965	0.6576	0.6041	0.5652	0.5116	0.4795	0.4662
	CTV	0.8353	0.7992	0.7573	0.7172	0.6789	0.6551	0.6005
	ODCT	0.8059	0.7483	0.6354	0.5871	0.5265	0.5076	0.4760
	PRINLM	0.7935	0.7553	0.6409	0.5775	0.5227	0.5493	0.5568
	VST-BM3D	0.8088	0.7615	0.7231	0.6926	0.6669	0.6422	0.6193
	TNRD_{S}	0.8649	0.8388	0.8184	0.7982	0.7614	0.7341	0.7067
	TNRD_{G}	0.6501	0.6012	0.7560	0.7767	0.5818	0.4374	0.4096
	A-TNRD	0.8528	0.8379	0.8180	0.7945	0.7613	0.7245	0.6807



(a) Noisy image (b) ODCT (c) PRINLM (d) VST-BM3D (e) TNRD_{G} (f) A-TNRD

Fig. 13: The noisy image and recovery results obtained by ODCT, PRINLM, VST-BM3D, TNRD_{G} and A-TNRD on one slice of the PD-weighted MRI data corrupted by Rician noise of $\sigma = 20$.

5.2.3 Results on PD-weighted dataset: Last but not least, we also train the TNRD models for the PD-weighted images degraded by Rician noise, where training samples are with noise levels $\sigma \in [15, 45]$ for our A-TNRD and TNRD_{G} model. Table 7 shows both average PSNR and SSIM obtained by the comparative algorithms on a three-dimensional PD-weighted MRI volume from the IXI-Hammersmith dataset. Thanks to the three-dimensional patch similarities, the ODCT and PRINLM give better restoration results

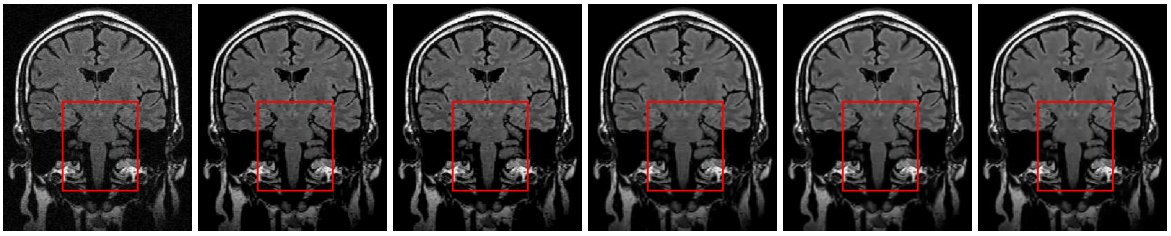


(a) ODCT (b) PRINLM (c) VST-BM3D (d) TNRD_{G} (e) A-TNRD

Fig. 14: The difference images between the ground truth and the restoration of ODCT, PRINLM, VST-BM3D, TNRD_{G}, A-TNRD in Fig. 13.

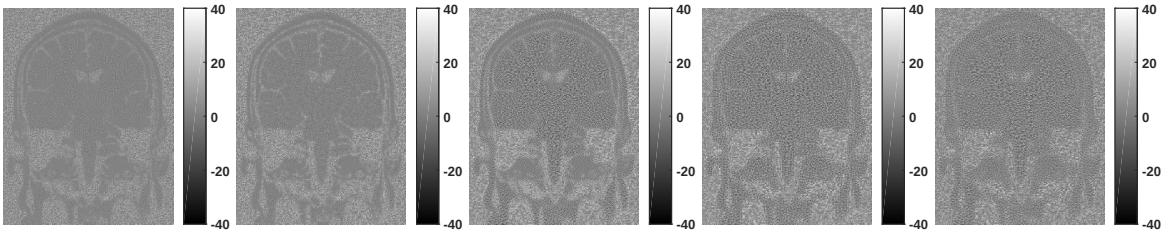
Table 7 PSRN(dB) and SSIM of different methods on a three-dimensional PD-weighted MRI data with Rician noise of different levels, where the best two results are highlighted in red and blue color, respectively.

	Methods	15	20	25	30	35	40	45
PSNR	Original	22.6617	20.1030	18.1257	16.5145	15.1554	13.9805	12.9459
	LGTV	27.2522	26.5239	25.2996	24.7906	24.4195	24.2255	23.2747
	CTV	32.0496	30.5474	29.7112	28.9564	27.7776	27.4726	26.6218
	ODCT	35.5010	33.5483	31.9526	30.8450	29.5191	28.5394	27.6130
	PRINLM	35.3556	33.6749	32.2363	31.0563	30.1177	29.1720	28.5084
	VST-BM3D	34.5948	32.9271	31.6307	30.5704	29.6633	28.8753	28.1693
	TNRD_{S}	35.8634	34.4205	33.2431	32.2455	31.4418	30.6946	30.0366
	TNRD_{G}	31.6452	32.4333	31.7947	31.7997	31.0813	29.0094	25.6817
	A-TNRD	35.1981	34.0922	33.0898	32.2014	31.3931	30.6087	29.7758
SSIM	Original	0.2964	0.2396	0.1989	0.1678	0.1432	0.1231	0.1065
	LGTV	0.6292	0.5545	0.5274	0.4884	0.4496	0.4081	0.3532
	CTV	0.8059	0.7813	0.7109	0.7062	0.6748	0.6282	0.5955
	ODCT	0.8359	0.7488	0.6786	0.6481	0.5752	0.5361	0.4960
	PRINLM	0.8503	0.8043	0.7562	0.7168	0.6833	0.6550	0.6335
	VST-BM3D	0.8150	0.7773	0.7489	0.7253	0.7066	0.6903	0.6767
	TNRD_{S}	0.8892	0.8745	0.8534	0.8328	0.8184	0.7931	0.7710
	TNRD_{G}	0.8130	0.7936	0.7187	0.8178	0.7987	0.5454	0.4029
	A-TNRD	0.8729	0.8642	0.8502	0.8319	0.8139	0.7895	0.7659



(a) Noisy (b) ODCT (c) PRINLM (d) VST-BM3D (e) TNRD_{G} (f) A-TNRD

Fig. 15: The noisy image and recovery results of ODCT, PRINLM, VST-BM3D, TNRD_{G} and A-TNRD of a real noisy MR image with the estimated noise level $\sigma = 10.0497$.

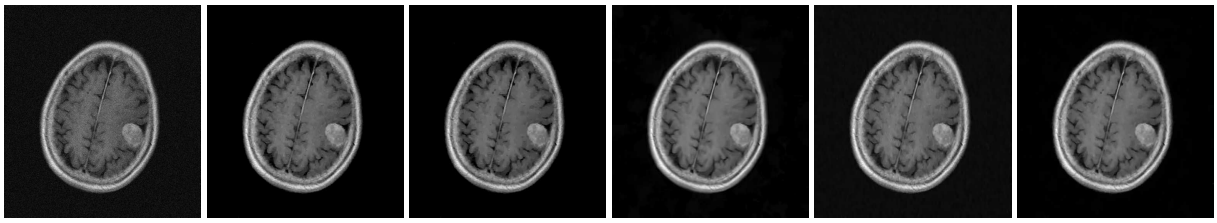


(a) ODCT (b) PRINLM (c) VST-BM3D (d) TNRD_{G} (e) A-TNRD

Fig. 16: The residual images between the restoration of ODCT, PRINLM, VST-BM3D, TNRD_{G}, A-TNRD and the noisy image in Fig. 15.

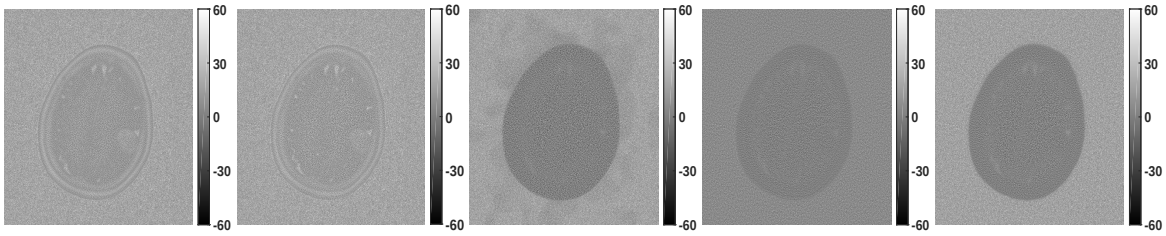
than our A-TNRD for noise level $\sigma = 15$. However, the restoration results of the ODCT and PRINLM method lack spatial dependency due to the point-to-point estimation. Therefore, the SSIM of ODCT and PRINLM are still inferior to our A-TNRD model. We observe that our A-TNRD model provides the overall best restoration results, the PSNR values of which triumph over both ODCT and PRINLM for all other noise levels. Fig. 13 and Fig. 14 illustrate the visual results of an image corrupted by Rician noise of $\sigma = 20$. We can see that better visual results are achieved by our A-TNRD model with fine details in the brain region and fewer outliers in the background.

For the three-dimensional PD-weighted MR volume, we set the regularization parameter γ in the CTV model as $\gamma = 0.07$, 0.05,



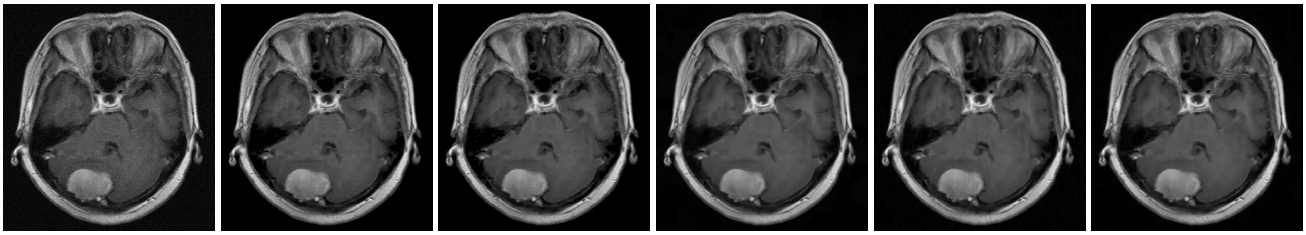
(a) Noisy (b) ODCT (c) PRINLM (d) VST-BM3D (e) $\mathrm{TNRD_G}$ (f) A-TNRD

Fig. 17: The noisy image and recovery results of ODCT, PRINLM, VST-BM3D, $\mathrm{TNRD_G}$ and A-TNRD of a tumor lesion MR image with the estimated noise level $\sigma = 13.7254$.



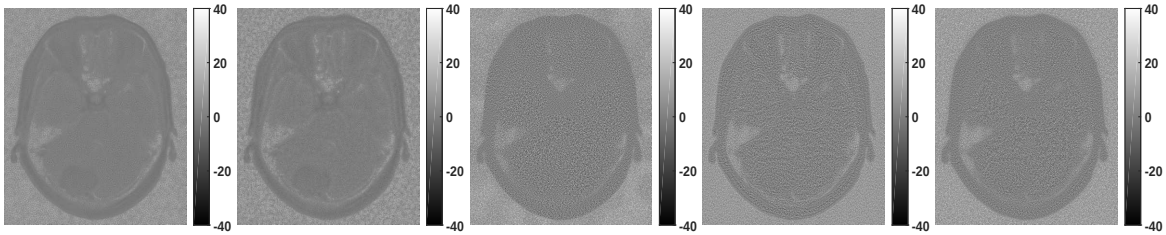
(a) ODCT (b) PRINLM (c) VST-BM3D (d) $\mathrm{TNRD_G}$ (e) A-TNRD

Fig. 18: The residual images between the restoration of ODCT, PRINLM, VST-BM3D, $\mathrm{TNRD_G}$, A-TNRD and the noisy image in Fig. 17.



(a) Noisy (b) ODCT (c) PRINLM (d) VST-BM3D (e) $\mathrm{TNRD_G}$ (f) A-TNRD

Fig. 19: The noisy image and recovery results of ODCT, PRINLM, VST-BM3D, $\mathrm{TNRD_G}$ and A-TNRD of a tumor lesion MR image with the estimated noise level $\sigma = 8.3647$.



(a) ODCT (b) PRINLM (c) VST-BM3D (d) $\mathrm{TNRD_G}$ (e) A-TNRD

Fig. 20: The residual images between the restoration of ODCT, PRINLM, VST-BM3D, $\mathrm{TNRD_G}$, A-TNRD and the noisy image in Fig. 19.

0.045, 0.04, 0.035, 0.03, 0.025 for noise level $\sigma = 15$, 20, 25, 30, 35, 40, 45, respectively.

5.3 Denoising performance on real data

To show the effectiveness of our proposal, we applied the A-TNRD model on real-world noisy MRI data in this subsection. Because the noise levels are unknown, we adopt the method proposed in [29] to estimate the noise levels of the noisy images shown in Fig. 15(a), Fig. 17(a) and Fig. 19(a), which are $\sigma = 10.4097$, 13.7254 and 8.3647, respectively. Thus, we train both A-TNRD and $\mathrm{TNRD_G}$ models on samples with noise levels $\sigma \in [1, 15]$. The restored images and residual images obtained by the ODCT, PRINLM, VST-BM3D, $\mathrm{TNRD_G}$, and A-TNRD model are shown in Fig. 15 and Fig. 16, respectively. As can be seen, our A-TNRD model can achieve better denoising results compared to other algorithms such that less information is exhibited in the residual images, especially the region inside the rectangle box.

Another two examples of tumor MRI images are shown in Fig. 17 - 20, which demonstrate that all the denoising methods can remove the Rician noises quite well. The comparisons between the residual images illustrate that the VST-BM3D model is not very

Table 8 The mean PSRN (dB) and SSIM of T_1-weighted MR images with different levels of Gaussian noise.

	Methods	15	20	25	30	35	40	45
PSNR	Original	24.5732	22.0745	20.1363	18.5526	17.2137	16.0539	15.0308
	TNRD_G	31.9491	31.2710	30.4546	29.5375	28.7648	27.9784	27.4424
	A-TNRD	**33.1149**	**31.8675**	**30.8015**	**29.8940**	**29.1012**	**28.3488**	**27.7587**
SSIM	Original	0.5330	0.4467	0.3819	0.3306	0.2887	0.2539	0.2246
	TNRD_G	0.9128	0.9003	0.8886	0.8561	0.8511	0.8215	0.8076
	A-TNRD	**0.9276**	**0.9113**	**0.8950**	**0.8781**	**0.8606**	**0.8413**	**0.8122**

Table 9 The mean PSRN(dB) and SSIM of T_1-weighted MR images with different levels of Laplace noise.

	Methods	15	20	25	30	35	40	45
PSNR	Original	24.6331	22.1343	20.1961	18.6125	17.2736	16.1137	15.0907
	TNRD_G	31.4705	30.6995	29.9500	29.1998	28.4557	27.6986	26.8468
	A-TNRD	**32.9158**	**31.5213**	**30.6983**	**29.7306**	**29.0374**	**28.3357**	**27.1439**
SSIM	Original	0.5405	0.4551	0.3908	0.3397	0.2980	0.2631	0.2337
	TNRD_G	0.8960	0.8834	0.8678	0.8508	0.8326	0.8101	0.7817
	A-TNRD	**0.9213**	**0.9076**	**0.8948**	**0.8802**	**0.8643**	**0.8331**	**0.7886**

good at processing homogeneous background, while the ODCT and PRINLM methods remove certain structural information of the images. Both TNRD models work well on the two MR images with tumors and our A-TNRD model outperforms the TNRD_G model since less information is left in the residual images.

5.4 Extension to other denoising problems

In this subsection, we apply our proposal to the T_1-weighted MRI images corrupted by Gaussian noises and Laplace noises to further investigate the effectiveness of our noise adaptive FoE regularization technique.

5.4.1 Gaussian noises removal: In order to deal with additive Gaussian noises, we minimize the energy functional combined with the noise adaptive FoE regularizer and the L^2 data fidelity term, which gives

$$\min_u \sum_{i=1}^{N_k}\sum_{p=1}^{N} \rho_i\Big(\big(g(\sigma)(k_i * u)\big)_p\Big) + \frac{\lambda}{2}\|u-f\|_2^2. \tag{19}$$

The diffusion process can be derived from (19) as follow

$$u_t = u_{t-1} - \sum_{i=1}^{N_k} g(\sigma)\bar{k}_i^t * \phi_i^t\big(g(\sigma)(k_i^t * u_{t-1})\big) - \lambda^t(u_{t-1}-f).$$

We train both TNRD_G and A-TNRD model for additive Gaussian noises removal on a training dataset with 100 image samples corrupted by the noise levels $\sigma \in [15, 45]$. The number of unrolling stages is fixed as 10 for both TNRD models. We evaluate both models on a testing dataset with 40 T_1-weighted images, the mean PSNR and SSIM of which are listed in Table 8. It can be seen that our A-TNRD model always achieves better restoration results than the TNRD_G model in terms of both PSNR and SSIM. More specifically, when noise level is small, e.g., $\sigma = 15$, our model gains a 1 dB higher PSNR than the TNRD_G model.

5.4.2 Laplace noises removal: According to the Bayesian statistics, we adopt the L^1 data term for dealing with the additive Laplace noises [30]. Therefore, we consider the following minimization problem for Laplace noise removal

$$\min_u \sum_{i=1}^{N_k}\sum_{p=1}^{N} \rho_i\Big(\big(g(\sigma)(k_i * u)\big)_p\Big) + \lambda\|u-f\|_1.$$

Similarly, the diffusion process can be obtained with the following form

$$u_t = u_{t-1} - \sum_{i=1}^{N_k} g(\sigma)\bar{k}_i^t * \phi_i^t\big(g(\sigma)(k_i^t * u_{t-1})\big) - \lambda^t\Big(\frac{u_{t-1}-f}{\sqrt{u_{t-1}-f^2+\varepsilon^2}}\Big),$$

where we define $\|u-f\|_1 = \sqrt{(u-f)^2+\varepsilon^2}$ with $\varepsilon = 0.01$ to avoid the singularity.

Numerically, we train both TNRD_G and A-TNRD model on a training dataset with 100 samples corrupted by the additive Laplace noises with the noise level $\sigma \in [15, 45]$. We list the mean value of PSNR and SSIM of a testing dataset composing of 40 T_1-weighted MR images in Table 9. As can be seen, the results of our model are consistently better than the TNRD_G model, especially for noise level $\sigma = 15$, which are also in accord with Rician and Gaussian denoising problems.

6 Discussion and Conclusion

Although we provided the PSNR and SSIM of the TNRD_S model trained on each specific noise level as the reference in Table 3, 4, 6, and 7, it is indeed unfair to compare our model with TNRD_S. The TNRD_S was trained on 100 samples with the same noise level, while our A-TNRD model was trained on 200 samples with varying noise levels $\sigma \in [15, 45]$. Obviously, the number of samples averaged to each noise level of our model is much smaller than the specific model. The unfairness in the training stage results in the differences of the PSNR and SSIM between the A-TNRD model and TNRD_S. The performance of our A-TNRD model can be further improved by increasing the size of training samples and the number of stages. In the following, we train another two A-TNRD models, i.e., A-TNRD^{300} and A-TNRD^{400}, which were trained on 300 training samples with 15 training stages and 400 training samples with 20 training stages, respectively. The mean values of the PSNR and SSIM on a test dataset with 40 T_1-weighted MRI images are displayed in Table 10. We can find out that the performance of the A-TNRD model has been improved compared to the A-TNRD^{200} model. Besides, both PSNR and SSIM of A-TNRD^{200} are inferior to TNRD_S for all noise levels, while almost all PSNR and SSIM of A-TNRD^{400} are higher than TNRD_S. Thus, we can improve the performance of our A-TNRD model by increasing the number of training samples and training stages in practice.

Table 10 The average PSNR (dB) and SSIM results on 40 T_1-weighted images corrupted by different levels of Rician noise.

Methods	20		30		40	
	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
TNRD_S	**31.8682**	0.9073	29.6326	0.8654	27.9441	0.8321
A-TNRD^{200}	31.4914	0.8985	29.4598	0.8599	27.7094	0.8048
A-TNRD^{300}	31.6570	0.9037	29.6862	0.8697	28.0468	0.8225
A-TNRD^{400}	31.7204	**0.9081**	**29.7620**	**0.8724**	**28.1388**	**0.8385**

In the end, we conclude the paper as follows. We developed a noise adapted field of experts image prior, which can learn effective prior information from training data utilizing a noise adaption function. The proposed FoE regularized model was solved by the trainable nonlinear diffusion process by learning all the parameters from the training dataset through a loss-based approach. The main advantage of the proposed TNRD model is that it can deal with Rician denoising problems with unknown noise levels. Compared to the general TNRD model trained with the same noisy samples, our model can achieve significantly better restoration results on both two- and three-dimensional data of different modality types including T_1-, T_2- and PD-weighted MRI data. Besides, the proposed model also outperforms other established Rician denoising methods such as VST-BM3D and PRINLM. Moreover, apart from Rician noises, consistent results were obtained by the proposed A-TNRD model for images corrupted by additive Gaussian noises and Laplace noises.

Our future works include improving the effectiveness and efficiency of our A-TNRD model on three-dimensional data by using the three-dimensional FoE prior and GPU programming. Another direction is to work on developing efficient learning-based models for blind denoising problems, where images are corrupted by noises of mixed distributions and diverse intensities.

7 Acknowledgements

The authors would like to thank the anonymous referees for their valuable comments and suggestions, which have helped very much to improve the presentation of this paper. The work was supported by NSFC 11701418, Major Science and Technology Project of Tianjin 18ZXRHSY00160 and Recruitment Program of Global Young Expert. The second author Hongwei Li was supported by National Natural Science Foundation of China (NSFC) (61971292) and key research project of the Academy for Multidisciplinary Studies, Capital Normal University. And he is also grateful to Beijing Higher Institution Engineering Research Center of Testing and Imaging as well as Beijing Advanced Innovation Center for Imaging Technology for funding his research.

8 References

1 Basu S., Fletcher T., Whitaker R. 'Rician noise removal in diffusion tensor mri', *Proc. Int. Conf. Medical Image Computing and Computer-Assisted Intervention*, Oct 2006, pp. 117–125
2 Getreuer P., Tong M., Vese L.A.: 'A variational model for the restoration of mr images corrupted by blur and rician noise', *Proc. International Symposium on Visual Computing*, 2011, pp. 686–698
3 Liu R.W., Shi L., Huang W., et al.: 'Generalized total variation-based mri rician denoising model with spatially adaptive regularization parameters.', *Magn. Reson. Imaging*, 2014, **32**, (6), pp. 702–720
4 Chen L., Zeng T.: 'A convex variational model for restoring blurred images with large rician noise', *J. Math. Imaging Vis.*, 2015, **53**, (1), pp. 92–111
5 Kang M., Kang M., Jung M.: 'Nonconvex higher-order regularization based rician noise removal with spatially adaptive parameters', *J. Vis. Commun. Image Represent.*, 2015, **32**, pp. 180–193
6 Martín A., Schiavi E., León S.S.D.: 'On 1-laplacian elliptic equations modeling magnetic resonance image rician denoising', *J. Math. Imaging Vis.*, 2017, **57**, (2), pp. 202–224
7 Kang M., Jung M., Kang M.: 'Rician denoising and deblurring using sparse representation prior and nonconvex total variation', *J. Vis. Commun. Image Represent.*, 2018, **54**, pp. 80–99
8 Chen W., You J., Che, B., et al.: 'A sparse representation and dictionary learning based algorithm for image restoration in the presence of rician noise', *Neurocomputing*, 2018, **286**, pp. 130–140
9 Liu L., Yang H., Fan J., et al.: 'Rician noise and intensity nonuniformity correction (nnc) model for mri data', *Biomed. Signal Process. Control*, 2019, **49**, pp. 506–519
10 Coupé, 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), pp. 425–441
11 Foi A.: 'Noise estimation and removal in mr imaging: the variance-stabilization approach', *Proc. IEEE International Symposium on Biomedical Imaging: from Nano to Macro*, Mar 2011, pp. 1809-1814
12 Roth S., Black M.J.: 'Fields of experts', *Int. J. Comput. Vis.*, 2009, **82**, (2), pp. 205-229
13 Tappen M.F.: 'Utilizing variational optimization to learn markov random fields', *Proc. IEEE Conference on Computer Vision and Pattern Recognition* , Jun 2007. pp. 1–8
14 Chen Y., Ranftl R., Pock T.: 'Insights into analysis operator learning: from patch-based sparse models to higher order mrfs', *IEEE Trans. Image Process.*, 2014, **23**, (3), pp. 1060–1072
15 Chen Y., Yu W., Pock T.: 'On learning optimized reaction diffusion processes for effective image restoration', *Proc. IEEE conference on computer vision and pattern recognition*, Jun 2015. pp. 5261–5269
16 Chen Y., Pock T.: 'Trainable nonlinear reaction diffusion: a flexible framework for fast and effective image restoration', *IEEE Trans. Pattern Anal. Mach. Intell.*, 2017, **39**, (6), pp. 1256–1272
17 Feng W., Qiao P., Chen Y.: 'Fast and accurate poisson denoising with trainable nonlinear diffusion', *IEEE T. Cybern.*, 2017, **48**, (6), pp. 1708–1719
18 Feng W., Chen Y.: 'Speckle reduction with trained nonlinear diffusion filtering', *J. Math. Imaging Vis.*, 2017, **58**, (1), pp. 162–178
19 Qiao P., Dou Y., Feng W., et al.: 'Learning non-local image diffusion for image denoising'. *Proc. ACM International Conference on Multimedia*, Oct 2017. pp. 1847–1855
20 Feng W., Qiao P., Xi X., et al: 'Image denoising via multiscale nonlinear diffusion models', *SIAM J. Imaging Sci.*, 2017, **10**, (3), pp. 1234–1257
21 Jiang D., Dou W., Vosters L., et al.: 'Denoising of 3d magnetic resonance images with multi-channel residual learning of convolutional neural network', *Jpn. J. Radiol.*, 2018, **36**, (9), pp. 566–574
22 Ran M., Hu J., Chen Y., et al.: 'Denoising of 3d magnetic resonance images using a residual encoder-decoder wasserstein generative adversarial network', *Med. Image Anal.*, 2019, **36**, pp. 165–180
23 Chen Y., Pock T., Ranftl R., et al.: 'Revisiting loss-specific training of filter-based mrfs for image restoration'. *Proc. German Conference on Pattern Recognition*, Sep 2013, pp. 271–281
24 Schmidt, U., Roth, S.: 'Shrinkage fields for effective image restoration', *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, Jun 2014, pp. 2774–2781
25 Liu D.C., Nocedal,J.: 'On the limited memory bfgs method for large scale optimization', *Math. Program.*, 1989, **45**, (1), pp. 503–528
26 LeCun Y., Bottou L., Bengio Y., et al.: 'Gradient-based learning applied to document recognition', *Proc. IEEE*, 1998, **86**, (11), pp. 2278–2324
27 Manjón J.V., Coupé P., Buades A., et al.: 'New methods for mri denoising based on sparseness and self-similarity', *Med. Image Anal.*, 2012, **16**, (1), pp. 18–27
28 Wang Z., Bovik A.C., Sheikh H.R., et al.: 'Image quality assessment: from error visibility to structural similarity', *IEEE Trans. Image Process.*, 2004, **13**, (4), pp. 600–612
29 Coupé P., Manjón J.V., Gedamu E., et al.: 'Robust rician noise estimation for mr images', *Med. Image Anal.*, 2010, **14**, (4), pp. 483–493
30 Calatroni L., Carlos D.L.R.J., Schönlieb C.B.: 'Infimal convolution of data discrepancies for mixed noise removal', *SIAM J. Imaging Sci.*, 2016, **10**, (3), pp. 1196–1233