IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 26, 2020, accepted April 20, 2020, date of publication April 27, 2020, date of current version May 19, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2990463

Medical Image Denoising Based on Biquadratic Polynomial With Minimum Error Constraints and Low-Rank Approximation

LINLIN JI[1], QIANG GUO[1,2], (Member, IEEE), AND MINGLI ZHANG[3], (Member, IEEE)
[1]Shandong Provincial Key Laboratory of Digital Media Technology, Shandong University of Finance and Economics, Jinan 250014, China
[2]School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan 250014, China
[3]Montreal Neurological Institute, McGill University, Montréal, QC H3A 2B4, Canada

Corresponding author: Qiang Guo (guoqiang@sdufe.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61873145 and Grant 61902220, in part by the Natural Science Foundation of Shandong Province under Grant ZR2017JL029, and in part by the Science and Technology Innovation Program for Distributed Young Talents of Shandong Province Higher Education Institutions under Grant 2019KJN045.

ABSTRACT To improve the visual quality of noisy medical images acquired by low radiation dose imaging, medical image denoising is highly desirable for clinical disease diagnosis. In this paper, a geometric regularization method is proposed for medical image denoising. To the best of our knowledge, this is the first work aiming at reconstructing surface by minimizing the gradient error and approximation error of the surface to suppress the noise in medical images. The proposed denoising method consists of two stages: one is to output a basic estimate and the other is for the residual noise reduction. Specifically, the method first exploits a biquadratic polynomial surface to generate an initial estimate of the noise-free image. The surface is constructed by dividing its coefficients into two groups. With the reconstruction error constraint, one group is used to minimize the gradient of the surface, and the other is to minimize the approximation accuracy of the surface. Then the residual noise in the initial result is further reduced by using the singular value thresholding mechanism, which exploits the self-similarity of medical images and the intrinsic low-rank property. Unlike the traditional truncated singular value thresholding scheme, the proposed singular value thresholding is derived by optimizing an objective function with a constraint. Experimental results on a real clinical data set demonstrate the effectiveness of the proposed denoising method, especially in detail-preserving. Compared with several widely used denoising methods, our method can achieve a better performance in terms of both quantitative metrics and subjective visual quality.

INDEX TERMS Error minimization, geometric regularization, low-rank approximation, medical image denoising, polynomial.

I. INTRODUCTION

Medical imaging has been widely applied in clinical disease diagnosis with the advent of digital imaging technologies. However, due to the requirement of low exposure to radiation and some existing technique limitations, all the acquired images with low radiation dose are more or less contaminated by noise [1]. The noise decreases the visual quality of medical images and leads to a negative effect on the accuracy of clinical diagnosis. Generally, noise will increase with the amount of radiation being decreased. Therefore, as an important preprocessing step to improve the quality of medical images, denoising is highly desirable for proper medical image analysis.

Formally, the acquired noisy image $X \in R^{m\times n}$ can be modeled as follows

$$X = Y + N, \tag{1}$$

where Y denotes the noise-free image and N is the additive noise with the standard deviation τ. The aim of image denoising is to restore the noise-free image Y from its noisy version X as accurately as possible. In the past decades, the problem of noise reduction has been extensively studied and numerous denoising algorithms have been proposed in the literature. Most of existing algorithms can be formulated as

The associate editor coordinating the review of this manuscript and approving it for publication was Krishna Kant Singh.

the following minimization problem

$$\arg\min_{Y} \|X - Y\|_2^2 + \lambda R(Y), \tag{2}$$

where the first term $\|X - Y\|_2^2$ is the data-fidelity term that constrains Y to be consistent with X as accurately as possible, the second term $R(Y)$ is a regularization term that constrains Y subject to some a prior, and λ is a balance parameter. Earlier efforts mainly focused on least square based priors, such as total variation (TV) prior [2], [3], local autoregressive prior [4], and others. By penalizing local discontinuity or dissimilarity, these regularization priors can lead to smooth denoised images. As a result, some subtle details are partly lost, while they are crucial information for clinical disease diagnosis. To overcome the limitation of these local priors, Buades *et al.* [5] proposed the well-known non-local mean (NLM) prior, which assumes each pixel can be represented by a weighted averaging of its non-local similar pixels in the image. In fact, NLM is a nonlocal generalization of local autoregressive prior, which is modeled as

$$R(y_i) = \frac{\sum_{j\in\Omega_i} w_{i,j} y_j}{\sum_{j\in\Omega_i} w_{i,j}}, \tag{3}$$

where y_i is the ith pixel of an image Y, $w_{i,j} = \exp\left(\frac{-\|Py_i - Py_j\|_2^2}{\sigma^2}\right)$ is the weight, σ is a similarity control parameter, and P and Ω denote an patch extraction operator and a search window centered at the i the pixel, respectively. To improve NLM's shape-adaptivity, a variant is developed in [6]. In essence, both local and non-local priors attempt to model the spatial relationship of pixels [7], and apply this relationship to estimate each pixel.

Unlike the spatial domain denoising methods described above, noise reduction can also be conducted in the transform domain. A basic assumption is that an image can be sparsely represented by a set of basis functions, such as wavelets [8], curvelets [9], and learned representation dictionary [1], [10]. The power of transform based methods stems from the sparsity of transformation coefficients, i.e., sparse representation prior, which makes noise be more easily distinguished. A widely used sparse representation prior is defined as

$$R(\alpha) = \|\alpha\|_1, \quad s.t., \ D\alpha = Y, \tag{4}$$

where α is a sparse representation vector, and D denotes a representation dictionary that can be constructed by using the fixed wavelet/curvelet basis or adaptively learned by a greedy algorithm from the noisy images or given noise-free images [11]. One disadvantage of sparse representation prior based methods is high computational complexity, while they have more effective denoising performance than traditional spatial methods.

Recent works have shown that low-rank priors are powerful models to reduce noise [12]–[14], in which an image is represented by a low-rank matrix. A simple and effective representation model for the low-rank prior is written as

$$rank(Y) = r, \quad s.t. \ r \ll \min\{m, n\}, \tag{5}$$

where $rank(\cdot)$ represents a rank function and r is the rank of matrix Y. When the rank r is unknown, finding the lowest rank matrix with rank constraint is NP-hard. In [15], the authors model the low-rank prior by using the nuclear norm that is defined by the sum of singular values of a matrix. In theory, the nuclear norm is the best convex approximation of the rank function. Under the low-rank assumption, the problem Eq. (2) can be translated as

$$\arg\min_{Y} \|X - Y\|_2^2 + \lambda\|Y\|_*, \tag{6}$$

where $\|Y\|_*$ is the nuclear norm of Y. This model has been widely used to deal with various image restoration problems. Many variants, such as weighted nuclear norm and low-rank and sparse combination priors, have also been proposed to further improve its performance [16]–[18]. The minimization problem (6) is tractable by iterative singular value thresholding algorithm. However, the iterative algorithm is computationally expensive. In order to address this issue, the fixed rank strategy can be adopted to avoid the iterative process and leads to lower running time. The goal of the fixed rank strategy is to solve the following problem

$$\arg\min_{Y} \|X - Y\|_2^2 \quad s.t. \ rank(Y) = r. \tag{7}$$

This problem has a close-form solution when the rank r is given [19]. Several feasible rank estimate schemes have been developed to determine the value of r [20], [21], which result in an efficient algorithm for low-rank image denoising.

More recently, there exists a growing interest in using deep learning to handle medical image denoising problem [22]–[24]. The reason is that deep learning based denoising methods have shown a great promise. However, this type of denoising methods is limited by large training test data and high training time complexity. Therefore, in this paper, we still focus on the traditional prior-based denoising method. Inspired by the combined image denoisers [25], we present a two-stage image denoising method. Specifically, the method first applies a biquadratic polynomial surface reconstruction algorithm to derive an initial denoised image, and then the residual noise in the initial result is further reduced by a singular value shrinkage strategy.

II. OVERVIEW OF THE PROPOSED METHOD

In general, the noise in medical images is relatively small, which means that most of the important features in the images are preserved. Therefore, it is possible to reduce the noise in medical images by surface fitting techniques. The biquadratic polynomial surface representation of image patches is an effective tool for dealing with image restoration problems [26], [27]. The key issue is how to define the constraint conditions that are used to construct a fitted surface with more textural details and little noise. In addition, medical images always focus on a specific tissue. They exhibit a high self-similarity, which results in the low-rank representation of image patches. Thus it is very suitable to apply singular value decomposition theory to reduce noise.

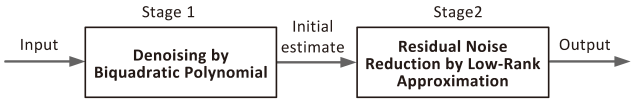


FIGURE 1. Flowchart of the proposed denoising method.

The proposed denoising method consists of two stages that depend on different regularization priors. The first stage exploits a biquadratic polynomial surface to derive an initial estimate of the noise-free image, in which the surface is constructed by dividing its coefficients into two groups. With the reconstruction error constraint, one group is used to minimize the gradient of the surface, and the other is to minimize the approximation accuracy of the surface. Based on the self-similarity of medical images and the intrinsic low-rank property, the second stage of the method is to optimize the singular value to further reduce the residual noise in the initial estimate. Figure 1 illustrates the denoising process of the proposed method.

III. DENOISING USING BIQUADRATIC POLYNOMIAL WITH MINIMUM ERROR CONSTRAINT

By using the gradient error as a regularization term, the minimization problem (2) can be formulated as

$$\arg\min_{Y} \|X - Y\|_2^2 + \lambda|\nabla Y|, \tag{8}$$

where $|\nabla Y|$ denotes the gradient term that enforces the image Y to be smooth. Numerical optimization algorithms have been developed to solve the Eq. (8). Unlike the traditional aforementioned methods, we propose a new method to solve it by decomposing it into two simple linear sub-problems. More specifically, the data-fidelity is approximated by a biquadratic polynomial, and the gradient regularization term is treated as a constraint term to mine the pixels with low noise. Therefore, the Eq. (8) can be represented as a constrained least square problem.

A. BIQUADRATIC POLYNOMIAL CONSTRUCTION

For simplicity of presentation, we assume X be an $n \times n$ noisy image, i.e., $x_{i,j}$ $(i, j = 1, 2, \cdots, n)$. $y_{i,j}$ is the corresponding noise-free pixel of $x_{i,j}$. In fact, from the perspective of geometry, $x_{i,j}$ is viewed as the function value of a point (i, j) in the ouv plane. Namely, the corresponding 3-D point of $x_{i,j}$ denoted by (u, v, x) is $(i, j, x_{i,j})$. With the geometrical symmetry, $y_{i,j}$ can be approximated by $x_{i,j}$ and its neighbors (here we use twenty-five pixels), i.e., $x_{i+l,j+k}$ $(l, k = -2, -1, 0, 1, 2)$. Figure 2 illustrates the neighbor relationship of the pixels used to calculate $y_{i,j}$. In this paper, we assume that each 5×5 patch centered on $y_{i,j}$ can be approximated by a biquadratic polynomial surface. Upon this assumption, image denoising problem is alternative to a biquadratic polynomial surface reconstruction. In other words, we exploit twenty-five pixels $x_{i+l,j+k}$ $(l, k = -2, -1, 0, 1, 2)$ to construct a biquadratic polynomial surface $f_{i,j}(u, v)$, and then uses $f_{i,j}(u, v)$ to estimate the noisy-free pixel $y_{i,j}$. Formally, the biquadratic polynomial is defined as

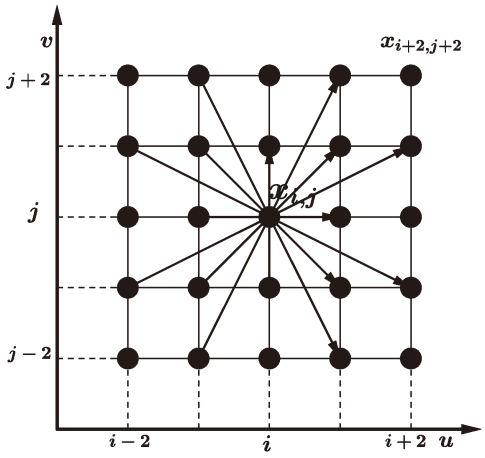


FIGURE 2. Neighbor pixels that are used to approximate the noisy-free pixel $y_{i,j}$.

$$f_{i,j}(u, v) = a_0 s^2 t^2 + a_1 s^2 t + a_2 s t^2 + a_3 s^2 + a_4 s t + a_5 t^2 + a_6 s + a_7 t + a_8. \tag{9}$$

where $s = u - i,\ t = v - j$.

In theory, we can use nine points to construct a unique surface. Thus it is possible to construct a high precision surface $f_{i,j}(u, v)$ using the known twenty-five pixels. A key issue is how to find the available pixels with low noise from these twenty-five pixels. Obviously, the significance of pixels in surface construction is inversely proportional to the noise level of these pixels. We adopt the following method to calculate the polynomial coefficients in Eq. (9).

B. COEFFICIENT CALCULATION

Ideally, the pixels $y_{i+l,j+k}$ $(l, k = -2, -1, 0, 1, 2)$ located in each image patch should be points in the surface $f_{i,j}(u, v)$. This implies that we can define a gradient term in Eq. (8) using the gradient information of the surface $f_{i,j}(u, v)$. By minimizing the gradient of $f_{i,j}(u, v)$, the parameters used in Eq. (9) will be adaptively determined. Specifically, we first calculate parameters a_1, a_2, a_6 and a_7. For ease of presentation, let $f_{i+s,j+t} = f_{i,j}(i + s, j + t)$. We have the following eight differences that are defined by eight directions (arrows in Fig. 2):

$$\begin{aligned}
e_1 &= (f_{i+1,j} - f_{i-1,j})/2 = a_6, \\
e_2 &= (f_{i+1,j+1} - f_{i-1,j-1})/2 \\
&= a_1 + a_2 + a_6 + a_7, \\
e_3 &= (f_{i,j+1} - f_{i,j-1})/2 = a_7, \\
e_4 &= (f_{i+1,j-1} - f_{i-1,j+1})/2 \\
&= -a_1 + a_2 + a_6 - a_7, \\
e_5 &= (f_{i+1,j+2} - f_{i-1,j-2})/2 \\
&= 2a_1 + 4a_2 + a_6 + 2a_7,
\end{aligned}$$

$$\begin{aligned} e_6 &= (f_{i+2,j+1} - f_{i-2,j-1})/2 \\ &= 4a_1 + 2a_2 + 2a_6 + a_7, \\ e_7 &= (f_{i+2,j-1} - f_{i-2,j+1})/2 \\ &= -4a_1 + 2a_2 + 2a_6 - a_7, \\ e_8 &= (f_{i+1,j-2} - f_{i-1,j+2})/2 \\ &= -2a_1 + 4a_2 + a_6 - 2a_7, \end{aligned} \tag{10}$$

where

$$\begin{aligned} e_1 &= (x_{i+1,j} - x_{i-1,j})/2, \\ e_2 &= (x_{i+1,j+1} - x_{i-1,j-1})/2, \\ e_3 &= (x_{i,j+1} - x_{i,j-1})/2, \\ e_4 &= (x_{i+1,j-1} - x_{i-1,j+1})/2, \\ e_5 &= (x_{i+1,j+2} - x_{i-1,j-2})/2, \\ e_6 &= (x_{i+2,j+1} - x_{i-2,j-1})/2, \\ e_7 &= (x_{i+2,j-1} - x_{i-2,j+1})/2, \\ e_8 &= (x_{i+1,j-2} - x_{i-1,j+2})/2. \end{aligned} \tag{11}$$

Based on the weighted least square theory, we define an objective function as follows

$$\begin{aligned} &G(a_1, a_2, a_6, a_7) \\ &\quad = w_1(e_1 - a_6)^2 + w_2(e_2 - a_1 - a_2 - a_6 - a_7)^2 \\ &\qquad + w_3(e_3 - a_7)^2 + w_4(e_4 + a_1 - a_2 - a_6 + a_7)^2 \\ &\qquad + w_5(e_5 - 2a_1 - 4a_2 - a_6 - 2a_7)^2 + w_6(e_6 \\ &\qquad - 4a_1 - 2a_2 - 2a_6 - a_7)^2 + w_7(e_7 + 4a_1 - 2a_2 \\ &\qquad - 2a_6 + a_7)^2 + w_8(e_8 + 2a_1 - 4a_2 - a_6 + 2a_7)^2, \end{aligned} \tag{12}$$

where w_i, $(i = 1, 2, \cdots, 8)$ are the weights defined as

$$\begin{aligned} w_1 &= w_3 = 1/(1 + \alpha), \\ w_2 &= w_4 = 1/(1 + 4\alpha), \\ w_5 &= w_6 = w_7 = w_8 = 1/(1 + 8\alpha). \end{aligned} \tag{13}$$

$2 \leq \alpha \leq 3$ is a control parameter that adjust the influence of eight differences defined in Eq. (10) on the objective function $G(a_1, a_2, a_6, a_7)$. By setting the derivatives of $G(a_1, a_2, a_6, a_7)$ to zero, i.e.,

$$\frac{\partial G(a_1, a_2, a_6, a_7)}{a_i} = 0, \quad i = 1, 2, 6, 7, \tag{14}$$

the optimal values of parameters a_i, $(i = 1, 2, 6, 7)$ can be obtained.

If only $x_{i,j}$ of twenty-five pixels $x_{i+l,j+k}$, $(l, k = -2, -1, 0, 1, 2)$ has noise, the optimal values of parameters a_i, $(i = 1, 2, 6, 7)$ derived from Eq. (14) are accurate. Otherwise, we need to find the pixels with low noise that have significant influences on the determination a_i, $(i = 1, 2, 6, 7)$ of these parameters. It is necessary to point out that the weights defined in Eq. (13) do not reflect the influence of noise level in each pixel. In the following, we discuss how to define new weights w_i, $(i = 1, 2, \cdots, 8)$, of which the influences on parameters are inversely proportional to noise level. Since the noise causes inaccurate estimation values of parameters a_i, $(i = 1, 2, 6, 7)$, we can define their estimation errors as

$$\begin{aligned} \omega_1 &= (e_1 - a_6)^2, \\ \omega_2 &= (e_2 - a_1 - a_2 - a_6 - a_7)^2, \\ \omega_3 &= (e_3 - a_7)^2, \\ \omega_4 &= (e_4 + a_1 - a_2 - a_6 + a_7)^2, \\ \omega_5 &= (e_5 - 2a_1 - 4a_2 - a_6 - 2a_7)^2, \\ \omega_6 &= (e_6 - 4a_1 - 2a_2 - 2a_6 - a_7)^2, \\ \omega_7 &= (e_7 + 4a_1 - 2a_2 - 2a_6 + a_7)^2, \\ \omega_8 &= (e_8 + 2a_1 - 4a_2 - a_6 + 2a_7)^2. \end{aligned} \tag{15}$$

For the first difference e_1 defined in Eq. (10), if the pixels $x_{i+1,j}$ and $x_{i-1,j}$ are accurate, the error ω_1 should be very small. Otherwise, ω_1 should be with a large value. It is easy to know that w_1 is inversely proportional to ω_1. Likewise, w_i, $(i = 2, 3, \cdots, 8)$ are also inversely proportional to ω_i, $(i = 2, 3, \cdots, 8)$. Therefore, we update the weights w_i as follows.

$$\begin{aligned} w_i &= 1/\big((1 + \alpha)(\omega_i + \delta)\big), \quad i = 1, 3; \\ w_i &= 1/\big((1 + 4\alpha)(\omega_i + \delta)\big), \quad i = 2, 4; \\ w_i &= 1/\big((1 + 8\alpha)(\omega_i + \delta)\big), \quad i = 5, 6, 7, 8, \end{aligned} \tag{16}$$

where δ is a very small constant for avoiding division by zero. By combining Eqs. (12) and (16), we can obtain the updated parameters a_i, $(i = 1, 2, 6, 7)$. These parameters can be used to update the errors ω_i in Eq. (15), which further leads to new weights w_i by Eq. (16). This is an iterative updating process. In this paper, we empirically found that it is sufficient to set the iteration number to three. By the weights determination algorithm described above, we can indirectly mine the pixels with low noise.

To determine other unknown parameters a_0, a_3, a_4, a_5 and a_8, we reformulate Eq. (9) as follows

$$f_{i,j}(u, v) = a_0 s^2 t^2 + a_3 s^2 + a_4 st + a_5 t^2 + a_8 + d(s, t), \tag{17}$$

where $d(s, t) = a_1 s^2 t + a_2 st^2 + a_6 s + a_7 t$. For each 5×5 image patch that consists of twenty-five pixels, the proposed method applies a biquadratic polynomial surface $f_{i,j}(u, v)$ to approximate it in the sense of least squares. Let

$$\begin{aligned} &H(a_0, a_3, a_4, a_5, a_8) \\ &\quad = \sum_{s=-2}^{2} \sum_{t=-2}^{2} w_{s,t} \big(a_0 s^2 t^2 + a_3 s^2 \\ &\qquad + a_4 st + a_5 t^2 + a_8 + d(s, t) - x_{i+s,j+t}\big)^2, \end{aligned} \tag{18}$$

where $w_{s,t} = 1/\big(1 + \beta(s^2 + t^2)\big)$ are the weights, β is a control coefficient. Minimizing this objective function $H(a_0, a_3, a_4, a_5, a_8)$ can obtain the values of parameters a_0, a_3, a_4, a_5, a_8. However, $H(a_0, a_3, a_4, a_5, a_8)$ only depends on the distances, which does not reflect the effect

of noise level in each pixel. Similar to Eqs. (15) and (16), we redefine the weight $w_{s,t}$ as

$$w_{s,t} = 1/\big((1+\beta(s^2+t^2))(\omega_{s,t}+\delta)\big), \quad s,t=-2,-1,0,1,2, \tag{19}$$

where $\omega_{s,t} = \big(f_{i,j}(s,t) - x_{i+s,j+t}\big)^2 = \big(a_0 s^2t^2 + a_3 s^2 + a_4 st + a_5 t^2 + a_8 + d(s,t) - x_{i+s,j+t}\big)^2$. Note that $f_{i,j}(s,t)$ denotes the value of the point $(i+s, j+t)$ of a surface $f_{i,j}(u,v)$. $\omega_{s,t}$ essentially is the error between $f_{i,j}(s,t)$ and the pixel $x_{i+s,j+t}$. As discussed above, if $x_{i+s,j+t}$ has a small amount of noise, the error $\omega_{s,t}$ is also small and the weight $w_{s,t}$ is large. By combining Eqs. (18) and (19), we can obtain the parameters a_i, $(i = 0, 3, 4, 5, 8)$. Similarly, we can use these parameters to further update the weight $w_{s,t}$. After running this iterative updating process three times, we can achieve the final values of the parameters a_i, $(i = 0, 3, 4, 5, 8)$. From Eq. (9), we know that a_8 is the estimate of $y_{i,j}$, namely, $\hat{y}_{i,j} = a_8$. The whole calculation procedure can be summarized as.

S1. Construct a biquadratic polynomial by Eq. (9) for a noisy pixel $x_{i,j}$;
S2. Calculate its eight differences using Eq. (10);
S3. Calculate the parameters a_i, $(i = 1, 2, 6, 7)$ by performing the following steps:
  S3-1. Define an objective function $G(a_1, a_2, a_6, a_7)$ according to Eq. (12);
  S3-2. Obtain the values of parameters a_i, $(i = 1, 2, 6, 7)$ by solving the Eq. (14);
  S3-3. Calculate the estimation errors ω_i, $(i = 1, 2, \cdots, 8)$ by Eq. (15);
  S3-4. Apply these errors to update the weights w_i, $(i = 1, 2, \cdots, 8)$ by Eq. (16);
  S3-5. Repeat steps S3-1 $\sim$ S3-4 three times to output the parameters a_i, $(i = 1, 2, 6, 7)$.
S4. Calculate the parameters a_i, $(i = 0, 3, 4, 5, 8)$ by performing the following steps:
  S4-1. Define an objective function $H(a_0, a_3, a_4, a_5, a_8)$ according to Eq. (18);
  S4-2. Obtain the values of parameters a_i, $(i = 0, 3, 4, 5, 8)$ by minimizing this objective function;
  S4-3. Calculate the weights w_i, $(i = 1, 2, \cdots, 8)$ by Eq. (19);
  S4-4. Repeat steps S4-1 $\sim$ S4-3 three times to output the parameters a_i, $(i = 0, 3, 4, 5, 8)$.
S5. Repeat the steps S1. $\sim$ S.4 described above for each pixel to output an initial denoised image $\widehat{Y}_0$.

IV. RESIDUAL NOISE REDUCTION USING SVD-BASED LOW-RANK APPROXIMATION

The biquadratic polynomial based denoising method is a local adaptive smooth technique. It can remove most of noise in a noisy image. But there are still a small amount of residual noise in the denoised result. To handle this problem, an alternative way is to exploit the non-local self-similarity of medical images and its intrinsic low-rank property to further reduce the residual noise. For each noisy image patch, we can search its similar patches and reshape them to form a similar patch matrix (denoted as $P \in R^{n\times n}$ for ease of discussion). Then this patch matrix is factorized by singular value decomposition (SVD) and approximated by truncating the singular values and corresponding singular vectors. This process can be simply formulated as

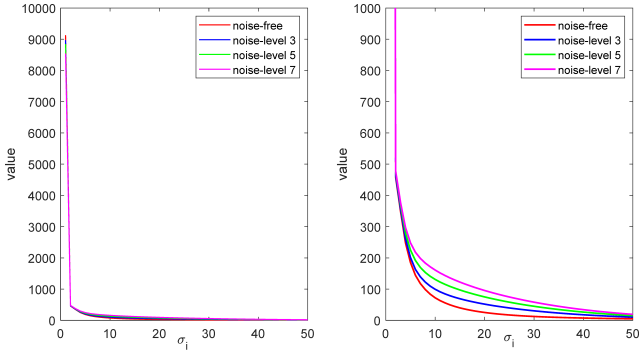


FIGURE 3. Distributions of singular values of images. From left to right: original figure and zoomed one.

$$\widehat{P}_{tsvd} = \sum_{i=1}^{r} \sigma_i u_i v_i^T, \tag{20}$$

where σ_i is the singular values of P, u_i and v_i denote the corresponding left singular vector and right singular vector, and the rank r of P is a key parameter that needs to be preseted. In [20], an estimate of the rank r is derived by using the Eckart-Young-Mirsky theorem. The self-similarity of medical images leads to the similar patch matrix P be low-rank. Owing to the energy compact property of SVD, most of singular values of P are approximately zero. Therefore, only a few largest singular values preserve most of important information. In essence, this truncated SVD can be considered as an adaptive group sparse representation method, in which singular values σ_i are representation coefficients and the products of left singular vector and right singular vector, i.e., $u_i v_i^T$, are basis functions. The truncation operator is equivalent to the well-known hard thresholding operator [28].

Even though this residual noise reduction method based on truncated SVD is simple and efficient, the truncated operation leads to some important details loss, especially for medical images. The main reason is that the patch matrix P is not an exact low-rank matrix. Figure 3 illustrates the distribution of singular values of a patch matrix P. It can be observed that the singular values $\sigma_i (i = 10, \cdots, n)$ are very small but not equal to zeros. These small singular values contain the noises and a part of important detail information. Therefore, to address this issue, we apply the classical soft shrinkage strategy to estimate these small singular values instead of setting them to zeros. Upon this shrinkage strategy, the noise-free estimate of a patch matrix P can be formulated as

$$\widehat{P} = \sum_{i=1}^{n} \mathcal{T}(\sigma_i) u_i v_i^T, \tag{21}$$

where the shrinkage operator $\mathcal{T}(\cdot)$ with the thresholds $\mu_i \geq 0$ $(i = r+1, \cdots, n)$ is defined by

$$\mathcal{T}(\sigma_i) = \begin{cases} \sigma_i, & i \leq r, \\ \sigma_i - \mu_i, & i > r. \end{cases} \tag{22}$$

Note that we empirically set $r = 9$ due to the high self-similarity of medical images. Although this strategy is very simply, we found that it is effective for denoising the test images used in experiments. Next, we discuss how to determine the thresholds μ_i. According to the Eckart-Young-Mirsky theorem, the error E between the patch matrix P and its estimate $\widehat{P}$ derived by truncated SVD is

$$E = \sum_{i=r+1}^{n} \sigma_i u_i v_i^T. \tag{23}$$

It is easy to know that the sum S of all elements of E can be represented by

$$S = c_{r+1}\mu_{r+1} + c_{r+2}\mu_{r+2} + \cdots + c_n\mu_n. \tag{24}$$

Ideally, the error should be the white gaussian noise, which means the sum of all elements of E is equal to zero. Without loss generality, let $c_{r+1} \neq 0$. Thus, Eq. (24) is written as

$$\mu_{r+1} = d_{r+2}\mu_{r+2} + d_{r+3}\mu_{r+3} + \cdots + d_n\mu_n, \tag{25}$$

where $d_i = -c_i/c_{r+1}$ $(i = r+2, \cdots, n)$. The method aims at minimizing the following constrained objective function

$$\begin{aligned} &\sum_{i=r+1}^{n} (\sigma_i - \mu_i)^2, \\ &s.t. \mu_{r+1} = d_{r+2}\mu_{r+2} + d_{r+3}\mu_{r+3} + \cdots + d_n\mu_n. \end{aligned} \tag{26}$$

Due to the fact that the singular values meet the condition $\sigma_i \geq \sigma_{i+1}$, the thresholds μ_i should progressively decrease. Upon this condition, the objective function (26) is changed as

$$\begin{aligned} &G(\mu_{r+2}, \mu_{r+3}, \cdots, \mu_n) \\ &= \left(\sum_{i=r+2}^{n} d_i\mu_i - \sigma_{r+1} \right)^2 \\ &\quad + \sum_{i=r+2}^{n} (\mu_i - \sigma_i)^2 + \sigma_i \left(\sum_{i=r+2}^{n} d_i\mu_i \right)^2 + \sum_{i=r+2}^{n} \sigma_i\mu_i^2, \end{aligned} \tag{27}$$

If d_i $(i = r+2, \cdots, n)$ are known, because of the convexity of $G(\mu_{r+2}, \mu_{r+3}, \cdots, \mu_n)$, the threshold μ_i can be calculated by

$$\frac{\partial G(\mu_{r+2}, \mu_{r+3}, \cdots, \mu_n)}{\mu_i} = 0. \tag{28}$$

In our experiments, we empirically set $d_i = \tau^2 - \frac{\sigma_i}{\sigma_{r+1}}$.

Due to the overlapping regions among neighbor patches, multiple estimates of a pixel can be achieved. Therefore, after obtaining the estimate of each patch matrix P, we need to aggregate all the denoised patches to output the final denoised image. The widely-used aggregation method is the weighted averaging. Although many complicated strategies have been presented to adaptively select the weights [6], here we adopt the uniform weights in the averaging scheme due to the computational simplicity, which can be expressed as

$$\hat{y}_{i,j} = \frac{1}{N_{i,j}} \sum_{k=1}^{N_{i,j}} \hat{y}_{i,j}^k, \tag{29}$$

where $y_{i,j}^k$ is the kth estimate of $y_{i,j}$ and $N_{i,j}$ is the number of its estimates. Based on the above discussions, the residual noise can be reduced by the following steps:

S1. Divide the initial denoised image $\widehat{Y}_0$ into a lot of overlapping patches, and reshaping each patch into a vector;
S2. For each patch, search its similar patches to construct a patch matrix P;
S3. Apply SVD on each patch matrix P to obtain the singular values σ_i and corresponding left and right singular vectors;
S4. Shrink the singular values using the thresholding function Eq. (22);
S5. Calculate the denoised version of patch matrix P by Eq. (21);
S6. Aggregate all denoised patches to output the final denoised image $\widehat{Y}$ by Eq. (29).

V. EXPERIMENTAL RESULTS

A. DATASETS

In our experiments, we use a real clinical data set that includes one anonymous patient's 30 brain images. The noisy images with different noise levels ($\sigma = 3, 5, 7$) are produced by adjusting the radiation dose. The information of these medical images are provided in Tables 1-3, and several images are shown in Figures 2-4.

B. EVALUATION CRITERIA

To provide a quantitative performance of the proposed method, three commonly-used objective metrics, peak signal-to-noise ratio (PSNR) [29], structural similarity (SSIM) index [30], and feature similarity (FSIM) index [31], are used to evaluate. The mathematical representations of these metrics are as follows.

- The PSNR is defined as

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right), \tag{30}$$

where $MSE = \frac{1}{mn}\sum_{i=1}^{m}\sum_{j=1}^{n}\left(\widehat{Y}_{i,j} - Y_{i,j}\right)^2$. PSNR represents the ratio between the maximum value of an image and its distort that affects the quality. The higher the PSNR, the better the quality of the denoised image.

- The form of SSIM index is

$$SSIM = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}, \tag{31}$$

where μ_x, σ_x and σ_{xy} denote the mean, standard deviation and cross-correlation of a signal, respectively, and

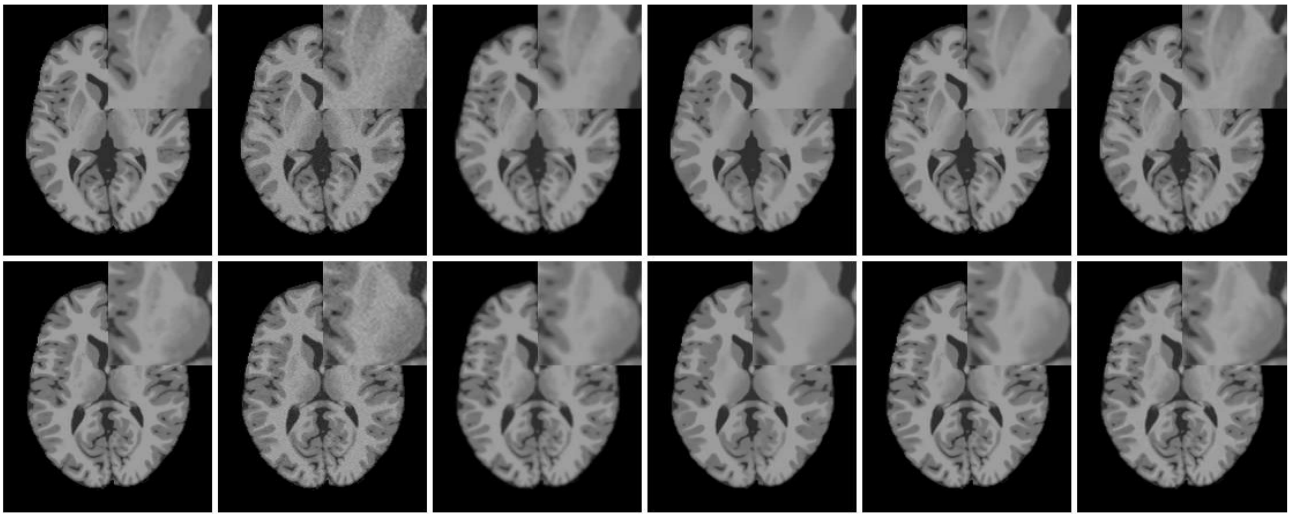


FIGURE 4. Visual comparisons on test *brain_02* and *brain_09* (noise level 3). From left column to right: Ground truth; Noisy; Gaussian; NLM; Hybrid; Proposed.

TABLE 1. PSNR/SSIM/FSIM results of different methods on test images with noise level 3.

Images	Noisy	Guassian Filtering	NLM Filtering	Hybrid Thresholding	Proposed
brain_01	37.51/0.944/0.936	38.77/0.978/0.975	39.26/0.978/0.975	42.16/0.985/0.984	39.53/0.984/0.984
brain_02	38.26/0.944/0.936	38.80/0.978/0.975	39.21/0.978/0.976	42.20/0.985/0.984	41.03/0.984/0.984
brain_03	36.57/0.947/0.938	38.63/0.978/0.974	39.20/0.978/0.977	42.11.0.985/0.985	38.10/0.984/0.984
brain_04	37.41/0.947/0.938	38.47/0.978/0.974	39.15/0.979/0.978	41.25/0.985/0.985	39.46/0.984/0.985
brain_05	37.30/0.948/0.941	38.56/0.978/0.973	39.10/0.979/0.978	42.25/0.985/0.985	39.16/0.984/0.985
brain_06	37.73/0.947/0.939	38.55/0.978/0.974	38.88/0.978/0.977	42.21/0.985/0.985	39.89/0.984/0.984
brain_07	38.08/0.947/0.939	38.65/0.978/0.974	38.93/0.978/0.977	42.17/0.985/0.985	40.71/0.984/0.984
brain_08	38.06/0.946/0.939	38.83/0.978/0.974	39.01/0.978/0.977	42.16/0.985/0.985	40.73/0.984/0.984
brain_09	38.32/0.945/0.937	39.00/0.979/0.975	39.10/0.977/0.980	42.19/0.985/0.985	41.53/0.984/0.984
brain_10	36.31/0.946/0.939	38.83/0.979/0.975	39.21/0.978/0.977	41.91/0.985/0.985	39.82/0.984/0.984
brain_11	36.67/0.945/0.938	38.75/0.979/0.975	39.14/0.978/0.978	41.75/0.985/0.985	39.38/0.984/0.985
brain_12	38.31/0.945/0.937	38.81/0.979/0.976	39.25/0.979/0.979	41.70/0.985/0.985	41.26/0.984/0.984
brain_13	38.33/0.944/0.932	39.02/0.979/0.976	39.48/0.980/0.981	41.31/0.985/0.985	41.58/0.985/0.984
brain_14	36.04/0.946/0.932	38.91/0.979/0.977	39.65/0.981/0.982	42.63/0.986/0.986	39.33/0.985/0.984
brain_15	36.35/0.945/0.926	39.00/0.980/0.978	39.76/0.982/0.983	41.06/0.985/0.985	39.93/0.985/0.983
brain_16	37.78/0.944/0.927	38.94/0.980/0.978	40.06/0.983/0.983	42.21/0.986/0.985	40.44/0.985/0.984
brain_17	37.99/0.943/0.918	39.03/0.980/0.979	40.37/0.985/0.986	41.59/0.985/0.986	40.96/0.985/0.980
brain_18	37.17/0.943/0.925	39.26/0.981/0.980	40.37/0.984/0.984	42.90/0.986/0.986	39.27/0.985/0.983
brain_19	36.64/0.945/0.923	39.39/0.982/0.980	40.42/0.985/0.986	41.59/0.984/0.985	38.26/0.986/0.982
brain_20	36.02/0.947/0.923	39.61/0.982/0.980	40.50/0.985/0.985	40.56/0.984/0.985	39.35/0.986/0.980
brain_21	35.15/0.947/0.933	39.87/0.982/0.980	40.42/0.985/0.984	43.08/0.986/0.986	39.18/0.986/0.983
brain_22	38.86/0.947/0.933	40.09/0.983/0.980	40.33/0.985/0.983	43.08/0.987/0.985	42.41/0.987/0.984
brain_23	38.97/0.949/0.923	40.13/0.983/0.982	40.49/0.985/0.986	35.89/0.979/0.980	42.23/0.987/0.976
brain_24	38.34/0.949/0.923	40.02/0.983/0.982	40.64/0.985/0.986	38.09/0.982/0.983	40.87/0.987/0.975
brain_25	36.85/0.949/0.937	40.16/0.983/0.981	40.68/0.985/0.985	42.54/0.985/0.985	38.39/0.986/0.982
brain_26	36.48/0.950/0.926	40.39/0.983/0.983	40.55/0.985/0.986	35.08/0.977/0.979	38.83/0.987/0.979
brain_27	37.68/0.949/0.928	40.46/0.984/0.983	40.46/0.985/0.986	37.13/0.980/0.982	39.52/0.986/0.976
brain_28	38.92/0.950/0.941	40.52/0.984/0.983	40.48/0.985/0.984	42.79/0.985/0.985	41.82/0.987/0.981
brain_29	35.67/0.952/0.926	40.63/0.984/0.984	40.56/0.985/0.987	35.34/0.978/0.978	39.71/0.987/0.983
brain_30	38.81/0.951/0.926	40.77/0.984/0.985	40.68/0.985/0.987	35.63/0.978/0.978	41.56/0.987/0.967
Average	37.42/0.947/0.932	39.36/0.981/0.978	39.84/0.982/0.982	40.89/0.984/0.984	40.13/0.985/0.981

C_1 and C_2 are small constants to characterize the saturation effects of the visual system at low luminance and contrast regions. Comparison with PSNR, SSIM is closer to the quality perception of the human visual system.

- The FSIM index is formed as

$$FSIM = \frac{\sum_{x\in\Omega} S_L(x) \cdot PC_m(x)}{\sum_{x\in\Omega} PC_m(x)}, \tag{32}$$

where $PC_m(\cdot)$ and Ω denote the phase congruency value and the whole image domain, respectively, and $S_L(\cdot)$ is the similarity metric that combines the phase congruency similarity and the gradient similarity.

C. DENOISING RESULTS

In order to evaluate the performance of noise reduction of our method, we compare the proposed method with two classical

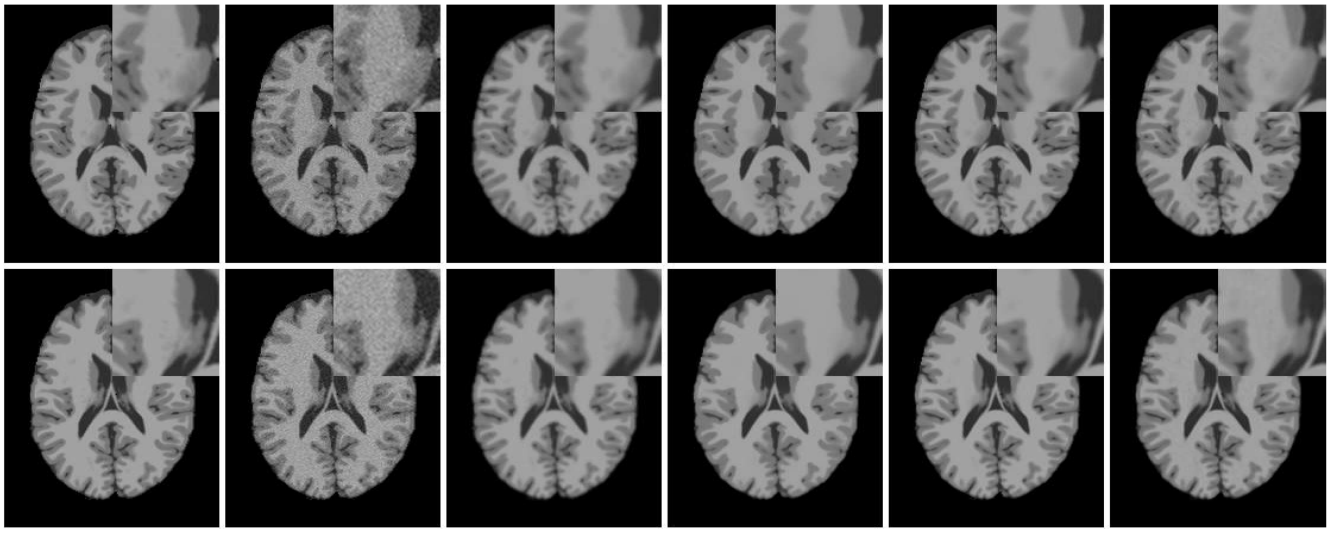


FIGURE 5. Visual comparisons on test *brain_13* and *brain_16* (noise level 5). From left column to right: Ground truth; Noisy; Gaussian; NLM; Hybrid; Proposed.

TABLE 2. PSNR/SSIM/FSIM results of different methods on test images with noise level 5.

Images	Noisy	Guassian Filtering	NLM Filtering	Hybrid Thresholding	Proposed
brain_01	33.01/0.879/0.874	38.17/0.975/0.971	38.71/0.976/0.973	40.40/0.980/0.979	38.84/0.981/0.980
brain_02	32.67/0.885/0.878	38.20/0.975/0.971	38.64/0.976/0.974	40.34/0.981/0.979	40.09/0.981/0.980
brain_03	30.46/0.888/0.881	38.03/0.975/0.970	38.63/0.976/0.975	40.31/0.981/0.980	37.53/0.981/0.980
brain_04	33.47/0.887/0.879	37.87/0.975/0.970	38.58/0.977/0.976	38.50/0.979/0.978	38.78/0.982/0.981
brain_05	32.72/0.887/0.881	37.95/0.975/0.970	38.53/0.977/0.976	40.45/0.981/0.980	38.46/0.982/0.981
brain_06	33.88/0.884/0.877	37.95/0.975/0.970	38.30/0.976/0.975	40.38/0.981/0.980	39.07/0.981/0.980
brain_07	32.91/0.889/0.884	38.04/0.975/0.971	38.36/0.975/0.975	40.39/0.981/0.980	39.84/0.981/0.980
brain_08	33.49/0.886/0.880	38.23/0.975/0.971	38.44/0.976/0.975	40.42/0.981/0.980	39.76/0.981/0.980
brain_09	32.33/0.886/0.880	38.39/0.976/0.972	38.55/0.975/0.975	40.46/0.981/0.980	40.53/0.981/0.981
brain_10	33.60/0.882/0.876	38.23/0.976/0.972	38.68/0.976/0.975	39.40/0.980/0.979	39.28/0.981/0.981
brain_11	31.70/0.887/0.882	38.15/0.975/0.972	38.61/0.976/0.976	39.29/0.979/0.978	38.97/0.981/0.981
brain_12	33.35/0.884/0.876	38.21/0.976/0.972	38.71/0.977/0.977	39.04/0.978/0.977	40.34/0.982/0.982
brain_13	32.91/0.883/0.872	38.42/0.976/0.973	38.94/0.978/0.979	39.82/0.980/0.980	40.66/0.982/0.982
brain_14	31.92/0.885/0.871	38.30/0.976/0.974	39.09/0.980/0.980	40.80/0.982/0.982	38.84/0.982/0.981
brain_15	31.89/0.884/0.865	38.38/0.977/0.975	39.21/0.980/0.981	39.06/0.978/0.977	39.42/0.982/0.981
brain_16	33.65/0.882/0.865	38.32/0.977/0.975	39.51/0.982/0.982	40.32/0.982/0.982	39.63/0.983/0.981
brain_17	32.20/0.884/0.857	38.41/0.978/0.977	39.81/0.983/0.985	39.80/0.980/0.980	40.14/0.983/0.978
brain_18	33.15/0.883/0.864	38.65/0.978/0.977	39.81/0.983/0.983	41.11/0.983/0.982	38.62/0.983/0.981
brain_19	33.35/0.881/0.859	38.78/0.979/0.978	39.83/0.984/0.985	40.68/0.980/0.981	37.73/0.984/0.980
brain_20	32.79/0.885/0.862	39.01/0.980/0.978	39.91/0.984/0.984	39.83/0.980/0.980	38.91/0.984/0.978
brain_21	32.75/0.888/0.876	39.27/0.980/0.977	39.83/0.984/0.983	41.31/0.983/0.982	38.84/0.983/0.980
brain_22	32.63/0.892/0.881	39.49/0.980/0.977	39.73/0.983/0.982	41.28/0.983/0.981	41.32/0.984/0.981
brain_23	33.81/0.892/0.869	39.52/0.980/0.979	39.88/0.984/0.984	35.77/0.977/0.978	41.13/0.984/0.974
brain_24	33.75/0.892/0.870	39.41/0.980/0.979	40.04/0.984/0.985	37.68/0.978/0.978	40.04/0.985/0.974
brain_25	32.13/0.894/0.888	39.55/0.981/0.979	40.11/0.984/0.984	41.38/0.981/0.981	37.85/0.984/0.980
brain_26	32.24/0.893/0.878	39.78/0.981/0.981	39.98/0.984/0.985	35.04/0.976/0.978	38.37/0.984/0.972
brain_27	31.76/0.894/0.882	39.86/0.981/0.981	39.90/0.983/0.984	36.89/0.978/0.979	38.87/0.984/0.972
brain_28	33.82/0.897/0.897	39.92/0.981/0.980	39.89/0.983/0.983	41.11/0.982/0.981	40.85/0.984/0.979
brain_29	32.34/0.897/0.884	40.02/0.982/0.982	39.97/0.984/0.985	35.28/0.977/0.977	39.31/0.984/0.966
brain_30	33.50/0.897/0.886	40.16/0.982/0.983	40.10/0.984/0.986	35.54/0.978/0.978	40.63/0.985/0.965
Average	32.78/0.888/0.876	38.76/0.978/0.975	39.28/0.9801/0.980	39.40/0.981/0.979	39.42/0.983/0.978

noise reduction methods: Gaussian filtering [32], non-local mean (NLM) filtering [5], an hybrid singular value thresholding [33]. All methods are coded in Matlab programming language. In our experiments, we perform these three methods on 30 test images with different noise levels ($\sigma = 3, 5, 7$), and evaluate the results in terms of both objective quantitative metrics and subjective visual quality.

To compare the denoising performance objectively, Tables 1-3 list the quantitative results of four denosing methods in terms of PSNR, SSIM and FSIM. It is seen that the proposed method outperforms Gaussian filtering and NLM filtering, which achieves the best quantitative performance, especially in term of PSNR. The proposed method is capable of obtaining gains in PSNR of up to

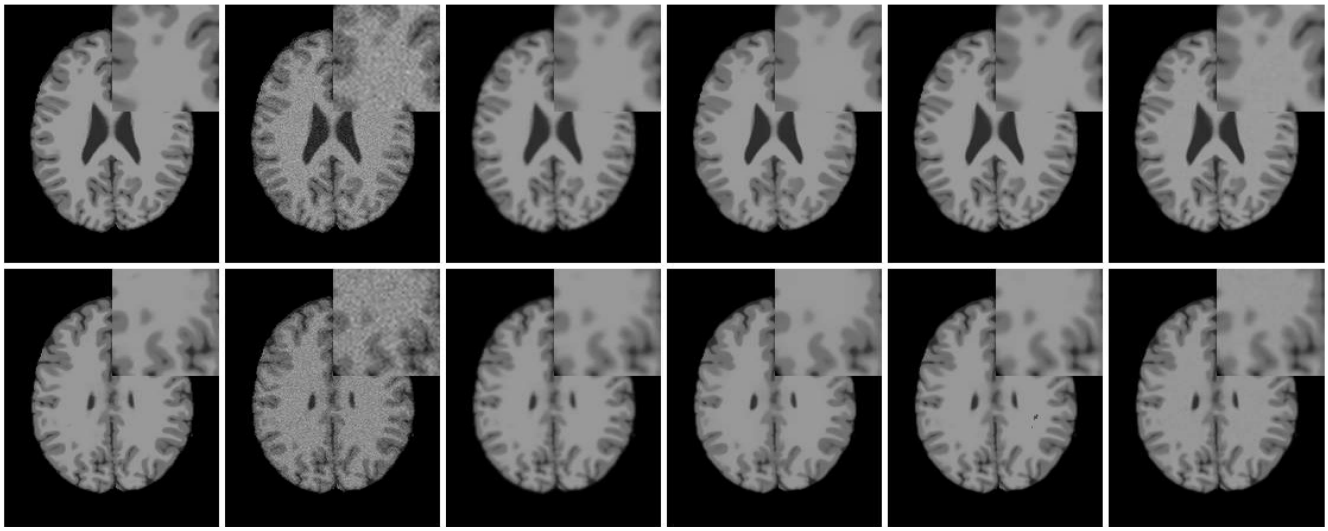


FIGURE 6. Visual comparisons on test *brain_22* and *brain_28* (noise level 7). From left column to right: Ground truth; Noisy; Gaussian; NLM; Hybrid; Proposed.

TABLE 3. PSNR/SSIM/FSIM results of different methods on test images with noise level 7.

Images	Noisy	Guassian Filtering	NLM Filtering	Hybrid Thresholding	Proposed
brain_01	29.15/0.820/0.823	37.45/0.971/0.967	38.18/0.974/0.971	38.08/0.976/0.974	38.22/0.978/0.977
brain_02	27.75/0.828/0.830	37.46/0.971/0.967	38.09/0.974/0.972	38.99/0.976/0.974	39.27/0.978/0.977
brain_03	29.68/0.828/0.826	37.28/0.970/0.966	38.08/0.974/0.973	38.98/0.976/0.974	37.18/0.978/0.977
brain_04	27.97/0.834/0.833	37.12/0.970/0.966	38.03/0.975/0.974	37.10/0.975/0.973	38.10/0.979/0.978
brain_05	28.79/0.832/0.833	37.20/0.970/0.966	37.99/0.975/0.973	39.09/0.977/0.975	37.90/0.978/0.977
brain_06	29.76/0.830/0.832	37.22/0.970/0.966	37.75/0.974/0.973	38.98/0.976/0.975	38.57/0.978/0.977
brain_07	29.68/0.828/0.830	37.30/0.970/0.966	37.82/0.973/0.973	38.96/0.976/0.974	38.96/0.978/0.977
brain_08	29.50/0.830/0.832	37.47/0.971/0.967	37.91/0.973/0.973	39.00/0.976/0.974	38.88/0.978/0.977
brain_09	30.17/0.825/0.825	37.63/0.971/0.968	38.02/0.973/0.972	39.11/0.976/0.975	39.58/0.978/0.977
brain_10	29.02/0.825/0.828	37.47/0.971/0.968	38.18/0.974/0.973	38.47/0.976/0.974	38.74/0.978/0.977
brain_11	29.48/0.828/0.831	37.41/0.971/0.968	38.09/0.974/0.974	38.38/0.976/0.974	38.14/0.978/0.978
brain_12	28.93/0.830/0.828	37.47/0.971/0.969	38.20/0.975/0.975	38.55/0.976/0.975	39.40/0.979/0.979
brain_13	29.75/0.827/0.823	37.67/0.972/0.970	38.43/0.976/0.978	38.46/0.976/0.975	39.77/0.980/0.979
brain_14	30.10/0.823/0.816	37.55/0.972/0.971	38.56/0.978/0.979	39.38/0.978/0.977	38.21/0.979/0.978
brain_15	29.94/0.823/0.812	37.61/0.972/0.972	38.69/0.978/0.980	38.41/0.976/0.975	38.94/0.980/0.978
brain_16	30.34/0.824/0.814	37.55/0.973/0.972	38.98/0.980/0.980	39.19/0.978/0.977	38.89/0.980/0.979
brain_17	29.57/0.827/0.807	37.65/0.974/0.973	39.27/0.982/0.983	38.31/0.976/0.976	39.33/0.980/0.976
brain_18	30.95/0.822/0.811	37.89/0.974/0.974	39.26/0.981/0.982	39.76/0.979/0.978	38.04/0.980/0.978
brain_19	30.76/0.826/0.809	38.05/0.975/0.975	39.27/0.982/0.983	39.76/0.978/0.978	37.35/0.981/0.977
brain_20	29.19/0.833/0.816	38.25/0.976/0.975	39.35/0.982/0.983	39.06/0.977/0.977	38.52/0.981/0.976
brain_21	30.06/0.833/0.828	38.51/0.976/0.974	39.27/0.982/0.981	39.90/0.980/0.978	38.23/0.980/0.978
brain_22	30.72/0.837/0.834	38.74/0.977/0.974	39.15/0.981/0.980	39.92/0.979/0.978	40.40/0.982/0.978
brain_23	28.77/0.842/0.829	38.77/0.977/0.976	39.28/0.982/0.983	36.47/0.971/0.970	40.21/0.982/0.972
brain_24	28.42/0.843/0.831	38.67/0.977/0.977	39.47/0.982/0.983	37.24/0.973/0.972	39.36/0.982/0.971
brain_25	28.80/0.844/0.847	38.81/0.977/0.976	39.55/0.982/0.982	39.34/0.977/0.977	37.51/0.981/0.977
brain_26	30.08/0.842/0.840	39.03/0.978/0.978	39.43/0.982/0.983	35.78/0.969/0.969	37.96/0.982/0.970
brain_27	29.80/0.840/0.841	39.11/0.978/0.978	39.36/0.982/0.983	36.54/0.972/0.972	38.46/0.981/0.970
brain_28	29.60/0.844/0.858	39.18/0.978/0.978	39.32/0.982/0.982	39.60/0.978/0.978	40.00/0.981/0.976
brain_29	30.38/0.842/0.849	39.29/0.978/0.980	39.40/0.982/0.984	35.21/0.966/0.966	38.97/0.981/0.964
brain_30	29.41/0.848/0.853	39.41/0.979/0.980	39.54/0.982/0.985	35.26/0.969/0.967	39.93/0.982/0.963
Average	29.55/0.832/0.829	38.01/0.974/0.972	38.73/0.978/0.978	38.37/0.976/0.976	38.77/0.980/0.976

0.20dB than NLM filtering, and is superior to Gaussian filtering by 0.73dB on average. We also observed that the performance of the hybrid singular value thresholding is not stable. Compared with other methods, the SSIM results also show that our method achieves performance gains up to 3.4% and 6.3%, respectively. In addition, the proposed method is slightly inferior to NLM filtering and hybrid thresholding in FSIM, but is significantly superior to Gaussian filtering.

For visual comparisons, Figures 4-6 show the denoised images of six different images with different noise levels, in which zoomed local regions are also given. As can be observed from them, the results by our method and Gaussian filtering are significantly better than NLM filtering and

hybrid thresholding, especially in detail-preserving. The reason is that the non-local averaging behavior of NLM leads to oversmoothed results and the minimum variance estimate used in hybrid thresholding also smooths detail information. Note that Gaussian filtering obviously outperforms NLM in visual quality, while NLM obtains higher quantitative performance than Gaussian filtering. This observation is significantly different from the case in dealing with natural images. One possible reason is that medical images can be effectively modeled by local prior used in Gaussian filtering. Additionally, it also indicates that quantitative metrics (PSNR, SSIM, and FSIM) widely used in medical image denoising may not be sufficient for evaluating the image quality.

We have also observed an interesting phenomenon as follows. Intuitively, the output of a denoising method should depend on the input. In other words, the denoised result from an input image with a higher noise level generally should has a lower quality. From Tables 1-3, we can observe that the performance of NLM and Gaussian filterings is relatively stable, while the PSNR values of the inputs (noisy images) are highly fluctuated. A possible reason for this phenomenon is that there exists a high self-similarity in medical images, which makes NLM and Gaussian filterings more stable. However, despite its stability, NLM results in a detail loss.

VI. CONCLUSIONS

This paper presents a new method for medical image denoising that combines biquadratic polynomial surface construction and low-rank approximation techniques. To the best of our knowledge, the biquadratic polynomial with minimum error constraints is first used to reduce noises in medical images. With the reconstruction error constraint, the construction coefficients are determined by minimizing the gradient and the approximation accuracy of the surface. To improve the performance of biquadratic polynomial based denoising, a new singular value thresholding strategy is applied to further suppress the residual noise. The thresholding is achieved by optimizing an objective function with a constraint. The feasibility and effectiveness of our method have been demonstrated by experimental results on a real clinical data set. In our future work, we will focus on further improving the real-time performance of the method.

REFERENCES

[1] S. Li, H. Yin, and L. Fang, "Group-sparse representation with dictionary learning for medical image denoising and fusion," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 12, pp. 3450–3459, Dec. 2012.

[2] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D, Nonlinear Phenomena*, vol. 60, nos. 1–4, pp. 259–268, Nov. 1992.

[3] Y. Wang and H. Zhou, "Total variation wavelet-based medical image denoising," *Int. J. Biomed. Imag.*, vol. 2006, Feb. 2006, Art. no. 89095

[4] X. Zhang and X. Wu, "Image interpolation by adaptive 2-D autoregressive modeling and soft-decision estimation," *IEEE Trans. Image Process.*, vol. 17, no. 6, pp. 887–896, Jun. 2008.

[5] A. Buades, B. Coll, and J. M. Morel, "A review of image denoising algorithms, with a new one," *Multiscale Model. Simul.*, vol. 4, no. 2, pp. 490–530, Jan. 2005.

[6] C.-A. Deledalle, V. Duval, and J. Salmon, "Non-local methods with shape-adaptive patches (NLM-SAP)," *J. Math. Imag. Vis.*, vol. 43, no. 2, pp. 103–120, Jun. 2012.

[7] Q. Guo, C. Zhang, Q. Liu, Y. Zhang, and X. Shen, "Image interpolation based on nonlocal self-similarity," *Science Asia*, vol. 40, no. 2, pp. 168–174, 2014.

[8] S. Mallat, *A Wavelet Tour of Signal Processing: The Sparse Way*. New York, NY, USA: Academic, 2008.

[9] Q. Guo and S. Yu, "Image denoising using a multivariate shrinkage function in the curvelet domain," *IEICE Electron. Express*, vol. 7, no. 3, pp. 126–131, 2010.

[10] M. Aharon, M. Elad, and A. Bruckstein, "*rm* K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Trans. Signal Process.*, vol. 54, no. 11, pp. 4311–4322, Nov. 2006.

[11] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.

[12] H. M. Nguyen, X. Peng, M. N. Do, and Z.-P. Liang, "Denoising MR spectroscopic imaging data with low-rank approximations," *IEEE Trans. Biomed. Eng.*, vol. 60, no. 1, pp. 78–89, Jan. 2013.

[13] M. D. Does, J. L. Olesen, K. D. Harkins, T. Serradas-Duarte, D. F. Gochberg, S. N. Jespersen, and N. Shemesh, "Evaluation of principal component analysis image denoising on multi-exponential MRI relaxometry," *Magn. Reson. Med.*, vol. 81, pp. 3503–3514, Feb. 2019.

[14] F. Shi, J. Cheng, L. Wang, P.-T. Yap, and D. Shen, "LRTV: MR image super-resolution with low-rank and total variation regularizations," *IEEE Trans. Med. Imag.*, vol. 34, no. 12, pp. 2459–2466, Dec. 2015.

[15] E. Candás and B. Recht, "Exact matrix completion via convex optimization," *Commun. ACM*, vol. 55, no. 6, pp. 111–119, Jun. 2012.

[16] S. Gu, Q. Xie, D. Meng, W. Zuo, X. Feng, and L. Zhang, "Weighted nuclear norm minimization and its applications to low level vision," *Int. J. Comput. Vis.*, vol. 121, no. 2, pp. 183–208, Jan. 2017.

[17] M. Zhang and C. Desrosiers, "Structure preserving image denoising based on low-rank reconstruction and gradient histograms," *Comput. Vis. Image Understand.*, vol. 171, pp. 48–60, Jun. 2018.

[18] M. Zhang and C. Desrosiers, "High-quality image restoration using low-rank patch regularization and global structure sparsity," *IEEE Trans. Image Process.*, vol. 28, no. 2, pp. 868–879, Feb. 2019.

[19] I. Markovsky, *Low Rank Approximation: Algorithms, Implementation, Applications*. London, U.K.: Springer: 2012.

[20] Q. Guo, C. Zhang, Y. Zhang, and H. Liu, "An efficient SVD-based method for image denoising," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 26, no. 5, pp. 868–880, May 2016.

[21] Q. Guo, S. Gao, X. Zhang, Y. Yin, and C. Zhang, "Patch-based image inpainting via two-stage low rank approximation," *IEEE Trans. Vis. Comput. Graphics*, vol. 24, no. 6, pp. 2023–2036, Jun. 2018.

[22] L. Gondara, "Medical image denoising using convolutional denoising autoencoders," in *Proc. IEEE 16th Int. Conf. Data Mining Workshops (ICDMW)*, Barcelona, Spain, Dec. 2016, pp. 241–246.

[23] Q. Yang, P. Yan, Y. Zhang, H. Yu, Y. Shi, X. Mou, M. K. Kalra, Y. Zhang, L. Sun, and G. Wang, "Low-dose CT image denoising using a generative adversarial network with wasserstein distance and perceptual loss," *IEEE Trans. Med. Imag.*, vol. 37, no. 6, pp. 1348–1357, Jun. 2018.

[24] S. P. Kyathanahally, A. Döring, and R. Kreis, "Deep learning approaches for detection and removal of ghosting artifacts in MR spectroscopy," *Magn. Reson. Med.*, vol. 80, no. 3, pp. 851–863, Sep. 2018.

[25] J. H. Choi, O. A. Elgendy, and S. H. Chan, "Optimal combination of image denoisers," *IEEE Trans. Image Process.*, vol. 28, no. 8, pp. 4016–4031, Aug. 2019.

[26] X. Li, C. Zhang, Y. Yue, and K. Wang, "Cubic surface fitting to image by combination," *Sc. China Inf. Sci.*, vol. 53, pp. 1287–1295, Jul. 2010.

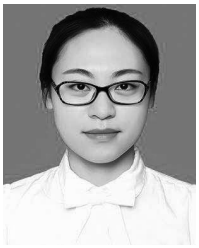
[27] Z. Caiming, Z. Xin, L. Xuemei, and C. Fuhua, "Cubic surface fitting to image with edges as constraints," in *Proc. IEEE Int. Conf. Image Process.*, Melbourne, VIC, Australia, Sep. 2013, pp. 1046–1050.

[28] D. L. Donoho and I. M. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," *J. Amer. Stat. Assoc.*, vol. 90, no. 432, pp. 1200–1224, Dec. 1995.

[29] J. Korhonen and J. You, "Peak signal-to-noise ratio revisited: Is simple beautiful?" in *Proc. 4th Int. Workshop Qual. Multimedia Exper.*, Yarra Valley, Australia, Jul. 2012, pp. 37–38.

[30] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, Apr. 2004.

[31] L. Zhang, L. Zhang, X. Mou, and D. Zhang, "FSIM: A feature similarity index for image quality assessment," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2378–2386, Aug. 2011.
[32] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*. Jersey City, NJ, USA: Pearson, 2017.
[33] F. Zhang, H. Fan, P. Liu, and J. Li, "Image denoising using hybrid singular value thresholding operators," *IEEE Access*, vol. 8, pp. 8157–8165, 2020.



LINLIN JI received the M.S. degree from the Hubei University of Technology, Wuhan, China, in 2019. She was with Gaoxunziyuan Education Technology Company, as a Software Engineer. She is currently with the Shandong Provincial Key Laboratory of Digital Media Technology, Shandong University of Finance and Economics. Her research interests include image processing and computing geometry.

QIANG GUO (Member, IEEE) received the B.S. degree in information and computing science from the Shandong University of Technology, Zibo, China, in 2002, and the M.S. and Ph.D. degrees in computer science from Shanghai University, Shanghai, China, in 2005 and 2010, respectively. He is currently a Professor at the Shandong Provincial Key Laboratory of Digital Media Technology and the School of Computer Science and Technology, Shandong University of Finance and Economics, Jinan, China. His research interests include image restoration, sparse representation, and object detection.

MINGLI ZHANG (Member, IEEE) received the Ph.D. degree in image processing and machine learning from the École de technologie supérieure (ÉTS), University of Quebec, Montreal, in 2017. She is currently a Postdoctoral Research Fellow at the Mcgill Centre for Integrative Neuroscience/Ludmer Centre for NeuroInformatics and Mental Health, Montreal Neurological Institute, Mcgill University, involving in brain image analysis. Her researches focus on designing and application of high-performance machine learning models to solve problems in the fields of computer vision, biomedical imaging, and natural images.