Image Denoising by a Local Clustering Framework

Partha Sarathi Mukherjee[1] and Peihua Qiu[2]

[1]Department of Mathematics, Boise State University, Boise, ID 83725

[2]Department of Biostatistics, University of Florida, Gainesville, FL 32611

Abstract

Images often contain noise due to imperfections of image acquisition techniques. Noise should be removed from images so that the details of image objects (e.g., blood vessels, inner foldings, or tumors in human brain) can be clearly seen, and the subsequent image analyses are reliable. With broad usage of images in many disciplines like medical science, image denoising has become an important research area. In the literature, there are many different types of image denoising techniques, most of which aim to preserve image features, such as edges and edge structures, by estimating them explicitly or implicitly. Techniques based on explicit edge detection usually require certain assumptions on the smoothness of the image intensity surface and the edge curves which are often invalid especially when the image resolution is low. Methods that are based on implicit edge detection often use multi-resolution smoothing, weighted local smoothing, and so forth. For such methods, the task of determining the correct image resolution or choosing a reasonable weight function is challenging. If the edge structure of an image is complicated or the image has many details, then these methods would blur such details. This paper presents a novel image denoising framework based on local clustering of image intensities and adaptive smoothing. The new denoising method can preserve complicated edge structures well even if the image resolution is low. Theoretical properties and numerical studies show that it works well in various applications.

*K*ey Words: Clustering, edges, edge structures, image denoising, image details, jump regression analysis, local smoothing, nonparametric regression.

1   Introduction

Over the last few decades, medical science has been using different types of images of human body parts for better understanding of their functions. Types of medical images include X-rays, computer tomography (CT), ultrasound, and so forth. Recently, magnetic resonance images (MRI) become popular. These medical images often contain noise due to hardware imperfections of the image acquisition techniques. Noise in images prevents the doctors from seeing the image details (e.g., blood vessels, inner foldings, or tumors in brain) clearly. So, for better medical diagnosis, noise should be removed in such a way that important image features, including the complicated edge structures and the fine details of the image objects, are preserved. Because of the increasing popularity of medical imaging,

image denoising with edges and other details preserved has become an important research area, which is the focus of the current paper.

In the literature, there are various types of image denoising techniques (Gonzalez and Woods 1992, Qiu 2005). One major type aims to preserve edge structures by detecting the edge curves explicitly. For instance, Qiu and Mukherjee (2010) proposed an image denoising technique that detected the edges first, estimated them locally by a pair of intersecting half lines, and then locally smoothed observed image intensities in a neighborhood whose pixels were on one side of the estimated edge curve. Qiu and Mukherjee (2012) proposed a 3-D image denoising method to handle a similar image denoising problem in 3-D cases. In practice, however, edge structures could be too complicated to be approximated well by local half lines. Furthermore, the complexity of the true image intensity surface at different places could be quite different, which is hard to accommodate by the methods just mentioned. For these reasons, most existing image denoising methods based on explicit detection of edge pixels would blur complicated edge structures and other fine details of image objects. Another major type of image denoising techniques does not detect edges explicitly. Instead, they obtain certain information about edges from the observed image intensities and use such information in their smoothing processes. For example, bilateral filtering methods (e.g., Chu et al. 1998, Tomasi and Manduchi 1998) use the edge information to assign weights in a weighted local smoothing procedure. Anisotropic diffusion methods (e.g., Perona and Malik 1990, Barash 2002) use the edge information to control the direction and the amount of local smoothing. One major drawback of these methods is that even though they assign small weights to certain pixels in their smoothing procedures, those pixels still get some weights and thus edges would be blurred, specially around the places with complicated edge structures. The methods based on Markov random field (MRF) modeling (e.g., Geman and Geman 1984, Godtliebsen and Sebastiani 1994) use the edge information by introducing a line process, whereas the methods by minimizing the total variation (TV) (e.g., Rudin et al. 1992, Keeling 2003, Wang and Zhou 2006) use the edge information by introducing a penalty term in their minimization problems. These global smoothing methods often blur local structures as well. Wavelet transformation methods (e.g., Chan et al. 2000, Portilla et al. 2003) are based on predefined basis functions and the denoised image is a linear combination of those basis functions. Performance of these methods depends heavily on how the basis functions are selected. Polzehl and Spokoiny (2000) proposed an adaptive weighted smoothing method where they considered a sequence of circular neighborhoods at each pixel. This method tries to choose smaller neighborhoods near edges and larger neighborhoods in the continuity regions. However, it is a challenging task to choose a reasonable neighborhood size, especially around places with complicated edge

structures. Takeda et al. (2007) proposed an adaptive smoothing method based on the so-called steering kernel regression, where the shape and size of the neighborhood for local smoothing depended on local edge information. One major drawback of this method is that even though neighborhoods are elongated along the edges, they often contain pixels on both sides of edges, resulting in image blurring. The non-local means approach (e.g., Buades et al. 2005, Coupe et al. 2008) and the approach based on jump regression analysis (e.g., Gijbels et al. 2006, Qiu 1998, 2009) would blur complicated edge structures as well. There are many other denoising methods in the literature. For example, the pointwise shape adaptive method by Foi et al. (2007), scale-space methods described in ter Haar Romney (2003), methods using channel or orientation spaces by Felsberg (2006), Franken and Duits (2009) and so on. See Qiu (2005, 2007) and Katkovnik et al. (2006) for a more detailed discussion on this topic.

In this paper, we propose a novel image denoising procedure which can well preserve major edge structures and other fine details of image objects (e.g., inner foldings and tumors in brain MRIs). Our proposed procedure is based on local clustering of pixels using their observed image intensities. The rationale of this procedure can be explained intuitively as follows. A 2-D monochrome image can be regarded as a surface of the image intensity function, and it is reasonable to assume that this surface is piecewisely continuous (c.f., Qiu 2007, Mukherjee and Qiu 2011). For example, in a brain MRI image, there are three major regions: gray matter, white matter, and cerebro-spinal fluid (CSF). In each region, it is reasonable to assume that the image intensity surface is continuous. Therefore, a neighborhood of a given pixel often contains more than one such regions, and we should only smooth data in the region that contains the given pixel when denoising the image. The proposed method consists of two major steps. First, we cluster the pixels in a neighborhood into two groups based on their observed image intensity values, and decide whether the two groups of image intensities are significantly different from each other. Second, the true image intensity at the given pixel is estimated by a weighted average of the observed image intensities at pixels located in the same group as the given pixel.

One major advantage of the proposed method is that it does not require explicit edge detection, and the complicated edge structures can be preserved well. As a demonstration, let us consider the following toy example in which the true image of size 64×64 has two intensity levels only. Figure 1 presents a noisy version of that image. To estimate the true image intensity at a given pixel P, we consider a circular neighborhood N_1. It can be seen that there are several disjoint black regions in N_1. The existing methods based on local edge estimation, such as the one in Qiu and Mukherjee (2010), will not work well in this case because the true edges cannot be approximated well by one or two lines in N_1.

If we use a smaller neighborhood N_2 to avoid the above problem, then the image would be under-smoothed in the sense that noise cannot be removed efficiently. Furthermore, as mentioned earlier, the task of choosing an appropriate neighborhood size is challenging. As a comparison, our proposed method first clusters all pixels in N_1 into two groups (i.e., the white group and the black group), and then it estimates the true image intensity at P by a weighted average of image intensities in the white group of N_1 only. As long as the pixel clustering is reliable, this method can preserve complicated edge structures and other fine details of image objects well.
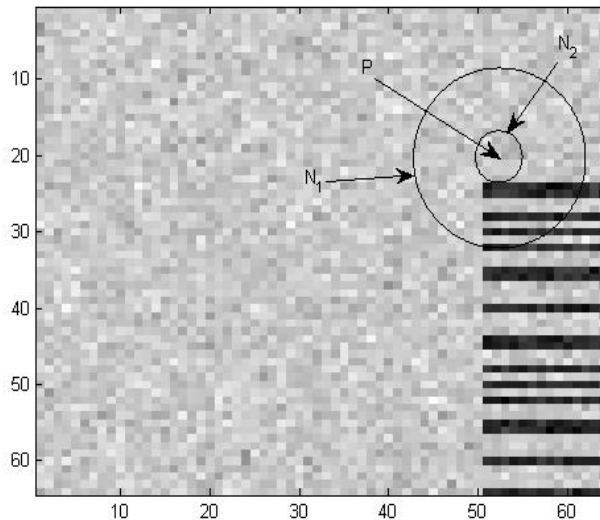

Figure 1: A toy example with a true image of two intensity levels, where the two neighborhoods N_1 and N_2 are considered when estimating the true image intensity at a given pixel P.

The remaining part of the article is organized as follows. The proposed method is described in detail in Section 2. Some of its statistical properties are discussed in Section 3. Section 4 presents some numerical examples concerning its performance in comparison with some state-of-the-art image denoising methods. Proofs of the two theorems in Section 3 are provided in a supplementary file.

2 Proposed Methodology

Although our proposed method can work in both two-dimensional (2-D) and 3-D cases, we describe it here in 2-D cases only for simplicity. Our description is given in five parts. The underlying regression model of the image denoising problem is described in Section 2.1. The 1-D classification procedure based on maximizing a separation measure is described in Section 2.2. An adaptive weighted smoothing procedure based on the 1-D classification

is described in Section 2.3. A modification of this smoothing procedure is described in Section 2.4. Selection of some procedure parameters is discussed in Section 2.5.

2.1 The underlying regression model

As we mentioned in Section 1, a monochrome 2-D image can be regarded as a 2-D image intensity surface that is usually discontinuous at the boundaries of image objects. In the framework of jump regression analysis (cf., Qiu 2005), the observed 2-D image can be described by the following 2-D regression model

$$\xi_{ij} = f(x_i, y_j) + \varepsilon_{ij}, \text{ for } i, j = 1, 2, \ldots, n, \tag{1}$$

where $\{(x_i, y_j) = (i/n, j/n), \ i, j = 1, 2, \ldots, n\}$ are the equally spaced design points (or pixels) in the design space $\Omega = [0, 1] \times [0, 1]$, ε_{ij} are i.i.d. random errors with mean 0 and unknown variance σ^2, $f(x, y)$ is an unknown regression function denoting the image intensity at (x, y), and $N = n^2$ is the sample size. We further assume that there exists a finite partition $\{\Lambda_l, \ l = 1, 2, \ldots, P\}$ of the design space Ω such that (i) each Λ_l is a connected region in Ω, (ii) $f(x, y)$ is continuous in $\Lambda_l \backslash \partial \Lambda_l$, for $l = 1, 2, \ldots, P$, where $\partial \Lambda_l$ is the boundary point set of Λ_l, (iii) $\bigcup_{l=1}^{P} \Lambda_l = \Omega$, and (iv) there exist at most finitely many points $\{(x_k^*, y_k^*), k = 1, 2, \ldots, K^*\}$ in $[\bigcup_{i=1}^{P} \partial \Lambda_i] \bigcap \Omega$ such that for each point (x_k^*, y_k^*) with $k = 1, 2, \ldots, K^*$, there are $\Lambda_{k_1}^*, \Lambda_{k_2}^* \in \{\Lambda_l, l = 1, 2, \ldots, P\}$ satisfying (a) $(x_k^*, y_k^*) \in [\Lambda_{k_1}^* \bigcap \Lambda_{k_2}^*]$, and (b) $\lim_{(x,y)\to(x_k^*,y_k^*),(x,y)\in\Lambda_{k_1}^*} f(x, y) = \lim_{(x,y)\to(x_k^*,y_k^*),(x,y)\in\Lambda_{k_2}^*} f(x, y)$. We call $[\bigcup_{i=1}^{P} \partial \Lambda_i] \bigcap \Omega$ the jump location curves (JLCs) of $f(x, y)$. Obviously, the JLCs are just (step) edge curves in image processing.

2.2 1-D classification by maximizing a separation measure

To decide whether two regions (Λ_l's) intersect within a neighborhood of a given pixel $(x, y) \in \Omega$, let us consider its circular neighborhood

$$O(x, y; h_n) = \{(u, v) : (u, v) \in \Omega, \sqrt{(u-x)^2 + (v-y)^2} \le h_n\},$$

where h_n is a bandwidth parameter. In $O(x, y; h_n)$, we first cluster pixels into two groups based on their observed intensity values ξ_{ij}'s, and then decide whether the observed image intensity values of the two groups of pixels are significantly different or not. Since the observed image intensity values are scalars, the first step is actually a 1-D clustering problem in which the number of clusters is known to be two. In this step, we only consider clustering the pixels into two groups instead of three or more because of the following two reasons. First, in many real-life images including the medical images, the situation that more than

two Λ_l's intersect near a single point is rare. Second, consideration of two clusters, instead of three or more, simplifies the procedure and its computation significantly.

In the literature, there are lots of methods for clustering, including the connectivity-based clustering (e.g., Ward 1963, Sibson 1973, Defays 1977), centroid-based clustering (e.g., MacQueen 1967, Lloyd 1957), and distribution-based classification (e.g., Dempster et al. 1977). In our procedure, theoretically speaking, any reasonable clustering algorithm can be used. Incidentally, our problem of 1-D classification into two groups is relatively simpler than the general problem of classifying multi-dimensional data into two or more groups. Here we propose a new clustering algorithm which is numerically convenient and efficient. We use it in all numerical examples presented in this paper. Remember that in our clustering problem, we need to classify all pixels in $O(x, y; h_n)$ into two groups based on the observed image intensities ξ_{ij}'s. A solution is called 'optimal' if the separation of the observed intensity values in the two related groups reaches the maximum. Intuitively, if $O(x, y; h_n)$ indeed contains two continuity regions of the image intensity surface only, then the within-group variability of the observed image intensities would be small and the between-group variability would be large. Consequently, the ratio of between-group variability and within-group variability would be large. On the other hand, if $O(x, y; h_n)$ intersects only one continuity region of the image intensity surface, then that ratio would be relatively small. Therefore, we can use this ratio as a separation measure of the two groups, and it can also be used as an indicator whether a given pixel (x, y) is close to a JLC. To classify pixels in $O(x, y; h_n)$ into two groups using the observed image intensities, we can introduce a thresholding parameter s and decide that the (i, j)-th pixel (x_i, y_j) belongs to group 1 if $\xi_{ij} \leq s$ and to group 2 otherwise. The value of s can be chosen from the interval

$$I(x, y; h_n) = \left(\min_{(x_i, y_j) \in O(x, y; h_n)} \xi_{ij}, \max_{(x_i, y_j) \in O(x, y; h_n)} \xi_{ij} \right)$$

For each s value in that interval, it divides $O(x, y; h_n)$ into the following two subsets:

$$O_1(x, y; h_n, s) = \{(x_i, y_j) : (x_i, y_j) \in O(x, y; h_n) \text{ and } \xi_{ij} \leq s\}$$
$$O_2(x, y; h_n, s) = \{(x_i, y_j) : (x_i, y_j) \in O(x, y; h_n) \text{ and } \xi_{ij} > s\}.$$

These two subsets are non-empty and disjoint, they may not form connected regions (cf., Figure 1), and thus their boundaries may not be two connected curves. Clearly, the 'optimal' value of s can be approximated by

$$S_0 = \arg\max_{s \in I(x, y; h_n)} T(x, y; h_n, s), \tag{2}$$

where

$$T(x, y; h_n, s) = \frac{|O_1(x, y; h_n, s)|(\bar{\xi}_1 - \bar{\xi})^2 + |O_2(x, y; h_n, s)|(\bar{\xi}_2 - \bar{\xi})^2}{\displaystyle\sum_{(x_i, y_j) \in O_1(x, y; h_n, s)} (\xi_{ij} - \bar{\xi}_1)^2 + \sum_{(x_i, y_j) \in O_2(x, y; h_n, s)} (\xi_{ij} - \bar{\xi}_2)^2}, \tag{3}$$

6

$\overline{\xi}$, $\overline{\xi}_1$ and $\overline{\xi}_2$ are the sample averages of the observed image intensities in $O(x, y; h_n)$, $O_1(x, y; h_n, s)$ and $O_2(x, y; h_n, s)$, respectively. Please note that the numerator and the denominator on the right hand side of (3) measure the between-group and within-group variability of the observed image intensities in $O_1(x, y; h_n, s)$ and $O_2(x, y; h_n, s)$, respectively. Also, the value of S_0 depends on (x, y) and the choice of h_n. We skip this information in notation for simplicity. Since the number of pixels in $O(x, y; h_n)$ is finite, S_0 must exist. In practice, we can calculate the values of $T(x, y; h_n, s)$ at G regularly spaced s values in $I(x, y; h_n)$, and choose the s value that maximizes the G values of $T(x, y; h_n, s)$ as an approximation of S_0. Of course, the larger the value of G is chosen, the better the approximation is. From our numerical experience, it works well for most images if we choose $G = 5$, and larger values of G can hardly improve the performance of the approximation. For this reason, in all our numerical examples in this paper, we use $G = 5$. As we mentioned earlier, $T(x, y; h_n, S_0)$ can also be used as a measure of the likelihood that $O(x, y; h_n)$ contains two continuity regions of the image. To this end, we claim that $O(x, y; h_n)$ indeed contains two continuity regions of the image if

$$T(x, y; h_n, S_0) > u_n, \tag{4}$$

where u_n is a threshold parameter. Otherwise, we claim that $O(x, y; h_n)$ contains only one continuity region of the image.

One important aspect to note here is that if $O(x, y; h_n)$ is contained in a continuity region of the image intensity surface, then local clustering is not necessary in that neighborhood. Moreover, we should use a relatively large neighborhood to smooth more observations at such a location for better removal of the noise. Therefore, the efficiency of the proposed procedure can be improved by running a rough pilot check to determine whether local clustering is necessary in $O(x, y; h_n)$. To this end, one reasonable approach is to check whether the sample standard deviation of all ξ_{ij}'s in $O(x, y; h_n)$ is about the same or smaller than σ. If the answer is positive, then we may decide not to run local clustering in $O(x, y; h_n)$ because the chance to have an edge curve in $O(x, y; h_n)$ is small in such cases. In practice, we usually do not know σ. Instead, it needs to be estimated from the observed data. One simple estimator of σ is

$$\widehat{\sigma} = \sqrt{\frac{1}{n^2} \sum_{i,j=1}^{n} \left(\xi_{ij} - \widehat{f}_{LCK}(x_i, y_j) \right)^2}, \tag{5}$$

where $\widehat{f}_{LCK}(x_i, y_j)$ is the local constant kernel (LCK) estimator of $f(x_i, y_j)$ defined as

$$\widehat{f}_{LCK}(x, y) = \sum_{(x_i, y_j) \in O(x, y; h_n^*)} \xi_{ij} K \left(\frac{x_i - x}{h_n^*}, \frac{y_j - y}{h_n^*} \right), \tag{6}$$

K is a 2-D density kernel function defined in a unit circle, and h_n^* is a bandwidth parameter. In our numerical examples, we choose $K(x,y) \propto \exp[-(x^2 + y^2)]\mathbf{1}(x^2 + y^2 \leq 1)$ which is the truncated 2-D Gaussian kernel function, and $h_n^* = 1/n$. Here, the selection of K and h_n^* is not critical because we only want a rough estimator of σ. After σ is estimated by $\widehat{\sigma}$, we can decide that local clustering in $O(x,y; h_n)$ is not necessary if the sample standard deviation of all ξ_{ij}'s in $O(x,y; h_n)$ is smaller than or equal to $1.0\widehat{\sigma}$. Here, the constant 1.0 is chosen conservatively in the sense that even if it is decided that local clustering is necessary in $O(x,y; h_n)$, there is still a possibility that the difference between the two resulting clusters is found to be insignificant by the criterion (4).

2.3 Proposed image denoising procedure

After the pixels in $O(x,y; h_n)$ are divided into two significantly different groups $O_1(x,y; h_n, S_0)$ and $O_2(x,y; h_n, S_0)$, the true image intensity $f(x,y)$ can be estimated by a weighted average of all ξ_{ij} in the group that contains (x,y), where the weights are determined by a similarity measure between the pixels (x_i, y_j) in the related group and the given pixel (x,y). Without the loss of generality, let us assume that (x,y) belongs to $O_1(x,y; h_n, S_0)$. One similarity measure can be quantified by considering small neighborhoods of size $\widetilde{h}_n$ (usually smaller than h_n) around the two pixels (x_i, y_j) and (x,y), and then calculating the L_2 distance of the observed intensity values in those neighborhoods. In this article, we choose similarity measure $\widetilde{W}_{ij} = \exp\left(-\frac{\|\widetilde{O}(x_i, y_j) - \widetilde{O}(x,y)\|_2^2}{2\widehat{\sigma}^2 |\widetilde{O}(x_i, y_j)| B_n}\right)$, where $\|\widetilde{O}(x_i, y_j) - \widetilde{O}(x,y)\|_2$ is the L_2 distance of the observed intensity values in the circular neighborhoods of radius $\widetilde{h}_n$ around (x,y) and (x_i, y_j), and B_n is a tuning parameter controlling the smoothness of the denoising procedure. Then, our proposed estimator of $f(x,y)$ is

$$\widehat{f}(x,y) = \frac{\sum\limits_{(x_i, y_j) \in O_1(x,y; h_n, S_0)} \widetilde{W}_{ij} \xi_{ij}}{\sum\limits_{(x_i, y_j) \in O_1(x,y; h_n, S_0)} \widetilde{W}_{ij}}. \tag{7}$$

If we decide not to do pixel clustering in $O(x,y; h_n)$ because the criterion (4) does not hold, then $f(x,y)$ can still be estimated by $\widehat{f}(x,y)$ above except that $O_1(x,y; h_n, S_0)$ needs to be replaced by $O(x,y; h_n)$ in (7).

Regarding $\widetilde{h}_n$, if it is chosen too large, then some fine details of the image could be lost. On the other hand, if it is chosen too small, then the estimator $\widehat{f}(x,y)$ could be too noisy. Based on our numerical experience, we suggest choosing $\widetilde{h}_n = 1.0/n$ at places where we need to go through the clustering step, and choosing $\widetilde{h}_n = 1.5/n$ at places where we do not need to go through that step.

2.4   A modification of the proposed image denoising procedure

As mentioned earlier, bigger neighborhoods should be used in the continuity regions of the image, compared to the neighborhoods used around edges. In Section 2.2, we propose the criterion that the given pixel (x, y) is considered to be in continuity regions if the sample standard deviation of all ξ_{ij} in $O(x, y; h_n)$ is smaller than or equal to $1.0\widehat{\sigma}$, where $\widehat{\sigma}$ is the estimated error standard deviation. So, in such cases, we consider using a bigger circular neighborhood of radius $c_1 h_n$. Then, we further check whether the sample standard deviation of all ξ_{ij} in $O(x, y; c_1 h_n)$ is still smaller than $1.0\widehat{\sigma}$. If the answer is positive, then we use the formula (7) to compute $\widehat{f}(x, y)$, after $O_1(x, y; h_n, S_0)$ is replaced by $O(x, y; c_1 h_n)$ and B_n is replaced by $c_2 B_n$. Otherwise, we compute $\widehat{f}(x, y)$ by (7) with $O_1(x, y; h_n, S_0)$ replaced by $O(x, y; h_n)$. Typically, c_1, and c_2 should be chosen larger than 1.0. From our numerical experience, we suggest using $c_1 = 3.0$, and $c_2 = 10.0$, and these values are used in all numerical examples presented in this article. The modified image denoising procedure is summarized as follows.

Modified Image Denoising Procedure

Step 1: Get an estimate of σ by the formula (5).

Step 2: For a given pixel (x, y), calculate the sample standard deviation $SD(x, y; h_n)$ of all ξ_{ij} in $O(x, y; h_n)$. If $SD(x, y; h_n) \geq \widehat{\sigma}$, go to Step 3. Otherwise, calculate $SD(x, y, c_1 h_n)$. If $SD(x, y; c_1 h_n) < \widehat{\sigma}$, then compute $\widehat{f}(x, y)$ by (7) after $O_1(x, y; h_n, S_0)$ is replaced by $O(x, y; c_1 h_n)$ and B_n is replaced by $c_2 B_n$. On the other hand, if $SD(x, y; c_1 h_n) \geq \widehat{\sigma}$, then compute $\widehat{f}(x, y)$ by (7) after $O_1(x, y; h_n, S_0)$ is replaced by $O(x, y; h_n)$.

Step 3: Cluster the pixels in $O(x, y; h_n)$ using their observed image intensity values by the procedure described in Section 2.2. If $T(x, y; h_n, S_0) \leq u_n$, then compute $\widehat{f}(x, y)$ by (7) after $O_1(x, y; h_n, S_0)$ is replaced by $O(x, y; h_n)$. Otherwise, compute $\widehat{f}(x, y)$ by (7) directly.

2.5   Selection of procedure parameters

Our proposed image denoising procedure contains three parameters h_n, u_n, and B_n. Because the performance of the proposed procedure depends on the values of these parameters, they should be chosen properly. One commonly used approach in the image processing literature is to try different values of the parameters and choose the ones with the best visual impression. Of course, this approach is usually subjective and inconvenient to use. In this

paper, we suggest using the cross-validation (CV) procedure to choose the parameters. By this procedure, we first define the CV score

$$\text{CV}(h_n, u_n, B_n) = \frac{1}{n^2} \sum_{i,j=1}^n \left(\xi_{ij} - \widehat{f}_{-i,-j}(x_i, y_j) \right)^2, \tag{8}$$

where $\widehat{f}_{-i,-j}(x_i, y_j)$ denotes the estimate of $f(x_i, y_j)$ when the (i, j)-th pixel (x_i, y_j) is not used in the estimation step. Then, the minimizers of $\text{CV}(h_n, u_n, B_n)$ defined in (8) are used as the chosen values of h_n, u_n, and B_n. There are some other methods for choosing these parameters, including the Mallow's C_p, bootstrap, and so forth (e.g., Marron 1988, Loader 1999, Hall and Robinson 2009). In this paper, the above CV procedure is used for its simplicity.

3   Some Statistical Properties

In this section, we discuss some statistical properties of the proposed image denoising procedure. In our discussion, a point (x, y) is called a *singular point* if one of the following two conditions are satisfied. (i) There exists some $\nu > 0$ such that, for any $0 < \widetilde{\nu} < \nu$, the circular neighborhood of (x, y) with radius $\tilde{\nu}$ contains more than two Λ_l's (cf., Section 2.1). (ii) The jump size of f at (x, y) is 0, i.e. (x, y) is one of those (x_k^*, y_k^*), $k \in \{1, 2, \ldots, K^*\}$ defined in Section 2.1. Next, we introduce some notations.

$$
\begin{aligned}
\Omega_\epsilon &= [\epsilon, 1 - \epsilon] \times [\epsilon, 1 - \epsilon], \\
J_\epsilon &= \{(x, y) : (x, y) \in \Omega, d_E((x, y), (x^*, y^*)) \le \epsilon \text{ for some } (x^*, y^*) \in D\}, \\
S_\epsilon &= \{(x, y) : (x, y) \in \Omega, d_E((x, y), (x^*, y^*)) \le \epsilon \text{ for a singular point } (x^*, y^*) \in D\}, \\
\Omega_{\bar{J},\epsilon} &= \Omega_\epsilon \backslash J_\epsilon, \\
\Omega_{\bar{S},\epsilon} &= \Omega_\epsilon \backslash S_\epsilon,
\end{aligned}
$$

where ϵ is a small positive constant, d_E denotes the Euclidean distance, and D denotes the set of points on the JLCs. Then, we have two theorems stated below, and their proofs are provided in a supplementary file.

Theorem 1 *Assume that f has continuous first-order derivatives over $(0, 1) \times (0, 1)$ except on the JLCs, its first order derivatives have one-sided limits at non-singular points of the JLCs, $\{\varepsilon_{ij}\}$ are i.i.d. and have the common distribution $N(0, \sigma^2)$, where $0 < \sigma < \infty$, $h_n = o(1)$, $1/nh_n = o(1)$, $u_n = \kappa + \delta$, where $\kappa = \left(\frac{\phi^2(0)}{\Phi(0)(1-\Phi(0))} \right) \Big/ \left[1 - \left(\frac{\phi^2(0)}{\Phi(0)(1-\Phi(0))} \right) \right]$, δ is any positive number, and ϕ and Φ are the probability density function and the cumulative distribution function of the $N(0, 1)$ distribution, respectively. Then, we have*

(i) *if $(x, y) \in \Omega_{\bar{J},\epsilon}$, then $T(x, y; h_n, S_0) \le u_n$, a.s.*

10

(ii) *on the other hand, if a non-singular point $(x, y) \in J_{h_n}$, and the minimum jump size of the JLC within $O(x, y; h_n)$ is larger than $4\kappa\sigma^2$, then $T(x, y; h_n, S_0) > u_n$, a.s.*

Theorem 2 *Under the assumptions in Theorem 1, if we further assume that $B_n = O(h_n^{1/2})$, then for any non-singular $(x, y) \in \Omega_{\bar{S},\epsilon}$, we have $\widehat{f}(x, y) = f(x, y) + O(h_n^{1/2})$, a.s.*

4 Numerical Studies

In this section, we present some numerical results concerning the performance of the proposed image denoising method, denoted as NEW, in comparison with three state-of-art image denoising methods that are widely used in the literature. The three competing methods include the image denoising method based on total variation minimization (cf., Rudin et al. 1992), denoted as TV, the adaptive image smoothing method using the steering kernel (cf., Takeda et al. 2007), denoted as ASSK, and the optimized non-local means image denoising method (cf., Coupe et al. 2008), denoted as ONLM. The TV method has a regularization parameter, which controls the amount of smoothing and edge preservation. The ASSK method is accomplished by adaptive smoothing using various neighborhoods with different shapes and sizes determined by the observed image intensities. This is an iterative procedure, and it has two parameters to choose: one is a global smoothing parameter and the other is the number of iterations. The ONLM method has two bandwidth parameters and another smoothing parameter to choose. Our proposed denoising method NEW has three parameters h_n, u_n, and B_n to choose.

The numerical study presented here includes one artificial image, one real fingerprint image, and one real magnetic resonance image (MRI) of human brain. First, the true artificial image with 64×64 pixels is presented in the first image of Figure 2. Its image intensities range from 0 to 1.3125. This image has a comb-like fine structure near the right boundary, several thin lines, and a rectangular object. Moreover, there is a small 'L'-like structure in the lower-middle part of the image. Then, we generate noisy versions of the true artificial image by adding i.i.d. noise from the $N(0, \sigma^2)$ distribution with $\sigma = 0.05, 0.10, 0.15$ and 0.20. These noisy versions are presented in the second, third, fourth, and fifth panels of Figure 2.

Because the methods TV, ASSK, and ONLM do not provide data-driven procedures to chose their procedure parameters, to make a fair comparison, we search their procedure parameters by minimizing the estimated MISE value, defined to be the sample mean of

$$\text{ISE}(\widehat{f}, f) = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left(\widehat{f}(x_i, y_j) - f(x_i, y_j) \right)^2,$$
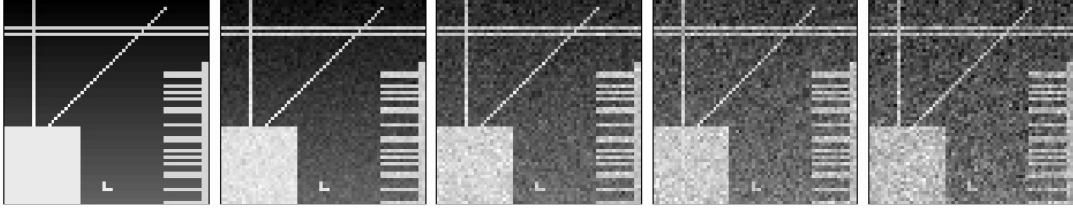
Figure 2: The first panel presents the true artificial image, the second, third, fourth, and fifth panels present its noisy versions when $\varepsilon_{ij} \stackrel{i.i.d.}{\sim} N(0, \sigma^2)$, where $\sigma = 0.05, 0.10, 0.15,$ and 0.20, respectively.

computed from 100 replicated simulations, where $\widehat{f}$ denotes the denoised image by a related denoising method. While the MISE criterion measures the overall performance of an image denoising procedure, it cannot tell us how well the edges and other fine details of the image are preserved. To measure the preservation of such fine details of the image, Hall and Qiu (2007) defined a measure of jump size (JS) of an image. Its discretized version for the true image intensity function f can be written as

$$\text{JS}(f) = \frac{1}{(n-2)^2} \sum_{i=2}^{n-1} \sum_{j=2}^{n-1} |f(x_i', y_j') - f(x_i'', y_j'')|,$$

where (x_i', y_j') and (x_i'', y_j'') are two immediately neighboring pixels of (x_i, y_j) on its two different sides along the estimated gradient direction of f at (x_i, y_j). Obviously, if (x_i, y_j) is an edge pixel, then $|f(x_i', y_j') - f(x_i'', y_j'')|$ is close to the jump size of f at (x_i, y_j). If (x_i, y_j) is a continuity pixel of f, then $|f(x_i', y_j') - f(x_i'', y_j'')|$ is close to 0. Thus, $\text{JS}(f)$ is a reasonable measure of the accumulative jump magnitude of f along the JLCs. After $\widehat{f}$ is obtained by an image denoising method, we can compute $\text{JS}(\widehat{f})$ using the estimated gradient directions of f. Then,

$$\text{EP}(\widehat{f}) = |\text{JS}(f) - \text{JS}(\widehat{f})|/\text{JS}(f)$$

is a reasonable measure of the edge preservation for the image denoising method in question. In the literature, there are a number of different methods to estimate the gradient of f. Since we are interested in the gradient directions rather than their magnitudes, a computationally simple filter should serve our purpose well. In all numerical examples presented in this paper, the 3×3 Sobel filter (cf., Qiu 2005, Section 4.4.3) is used when estimating f_x' and f_y'.

The numerical results for the artificial image is presented in Table 1, where the first row in each entry presents the estimated MISE value and its standard error (in parenthesis), the second row presents the estimated EP value and its standard error (in parenthesis), and

the third row presents the searched procedure parameter values. When comparing two methods in terms of MISE, if their estimated MISE values are MISE_1 and MISE_2 with standard errors SE_1 and SE_2, respectively, and if $\text{MISE}_1 < \text{MISE}_2$, then a commonly used practical guideline is that we conclude that method 1 is significantly better than method 2 when $\text{MISE}_2 - \text{MISE}_1 > \nu(\text{SE}_1 + \text{SE}_2)$, where $\nu > 0$ is a given number. In practice, people often choose ν from the interval $[1, 3]$. In our numerical examples, we use $\nu = 2$. Similar comparisons can be made among different methods in terms of EP. We consider the performance of the denoising method 1 to be better than method 2 if (i) $\text{MISE}_1 < \text{MISE}_2$ significantly, or (ii) $\text{MISE}_1 \approx \text{MISE}_2$ and $\text{EP}_1 < \text{EP}_2$ significantly.

Table 1: In each entry, the first line presents the estimated MISE value based on 100 simulations and the corresponding standard error (in parenthesis), the second line presents the value of EP and its standard error (in parenthesis), and the third line presents the searched procedure parameter values. This table is about the artificial image shown in Figure 2. The best method in each case is indicated by *italicized numbers*.

σ	TV	ASSK	ONLM	NEW
0.05	0.0010 (0.0000)	0.0011 (0.0001)	0.0009 (0.0001)	*0.0001 (0.0000)*
	0.0038 (0.0028)	0.0841 (0.0050)	0.0165 (0.0043)	*0.0034 (0.0025)*
	40.0	0.14, 5	8, 1, 0.05	*0.0938, 2.0, 3.0*
0.10	0.0038 (0.0002)	0.0037 (0.0001)	0.0017 (0.0001)	*0.0004 (0.0001)*
	0.0299 (0.0078)	0.1732 (0.0121)	0.0610 (0.0090)	*0.0064 (0.0052)*
	20.0	0.18, 5	8, 1, 0.10	*0.0938, 2.0, 3.0*
0.15	0.0082 (0.0003)	0.0067 (0.0003)	0.0030 (0.0002)	*0.0019 (0.0003)*
	0.0513 (0.0137)	0.1810 (0.0167)	0.0957 (0.0117)	*0.0457 (0.0093)*
	13.0	0.21, 6	10, 1, 0.15	*0.0938, 2.0, 2.0*
0.20	0.0137 (0.0006)	0.0112 (0.0006)	*0.0055 (0.0004)*	0.0097 (0.0012)
	0.0925 (0.0187)	0.2500 (0.0245)	*0.1003 (0.0173)*	0.1712 (0.0212)
	10.0	0.23, 6	*10, 1, 0.20*	0.0938, 2.0, 1.0

From Table 1, we see that in the cases when $\sigma = 0.05$, 0.10, and 0.15, the proposed method NEW outperforms all of its competitors in terms of both MISE and EP. When $\sigma = 0.20$, NEW is better than ASSK in terms of both MISE and EP, it is better than TV in terms of MISE, but TV has a smaller EP value than NEW, and ONLM is better than NEW in terms of both MISE and EP. One realization of each of the denoised images by NEW and its three competitors when $\sigma = 0.10$ are presented in the first row of Figure 3.

Their mean deviation images, defined as $\sum_{r=1}^{100}(\widehat{f}_r - f)/100$ are presented in the second row of Figure 3. If an image denoising method performs well, then there should not be any non-random pattern in the corresponding deviation image. From Figure 3 we see that NEW indeed performs better than its competitors in this case.
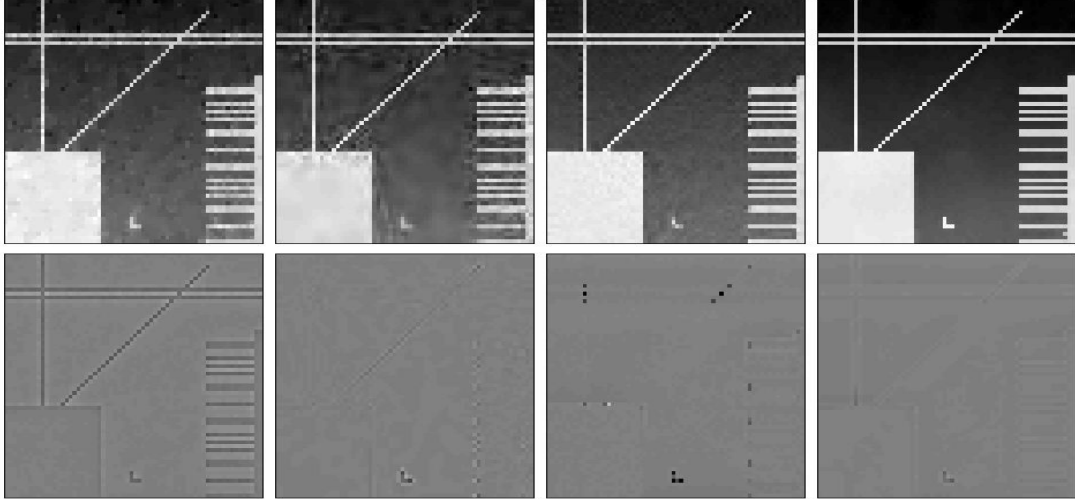

Figure 3: The first row shows the denoised images by TV, ASSK, ONLM and NEW in the artificial image example when $\sigma = 0.10$. The second row shows corresponding mean deviation images defined as $\sum_{r=1}^{100}(\widehat{f}_r - f)/100$, where $\widehat{f}_r$'s are the denoised images by a method in consideration.

Next, we consider a real fingerprint image with 346×346 pixels and a real MRI image of human brain with 319×342 pixels. In both images, the image intensity values range from 0 to 255, and i.i.d. noise from the distribution $N(0, \sigma^2)$ is added to them, where σ is chosen to be 5, 10, 15, and 20, representing different levels of noise. The noiseless images and several noisy versions are shown in Figure 4. Then, we apply the four image denoising procedures to these two examples, and their parameters are chosen in the same way as those in the previous example. The results corresponding to Table 1 and Figure 3 are presented in Tables 2 and 3 and Figures 5 and 6.

From Table 2, we see that NEW uniformly outperforms ASSK in terms of both MISE and EP in the fingerprint example. NEW is better than ONLM when $\sigma = 5$, 10, and 15 in terms of both MISE and EP. When $\sigma = 20$, ONLM is better than NEW in terms of MISE but its EP value is larger than that of NEW. It seems that TV is better than NEW in terms of both MISE and EP when $\sigma = 5$, their performance is similar when $\sigma = 10$, and NEW is slightly better than TV in terms of MISE and much better than TV in terms of EP when $\sigma = 15$ or 20. From Figure 5, it seems that the mean deviation image of NEW shows less
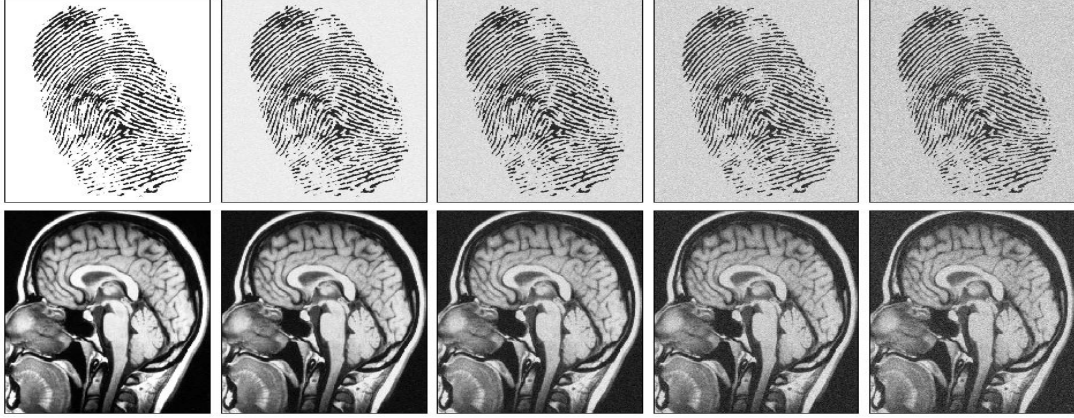
14

Figure 4: The first image in the first row shows the true fingerprint image, and the second, third, fourth, and fifth images are its noisy versions with $\sigma = 5, 10, 15,$ and 20, respectively. The second row shows the corresponding images in the MRI brain image example.

Table 2: In each entry, the first line presents the estimated MISE value based on 100 simulations and the corresponding standard error (in parenthesis), the second line presents the value of EP and its standard error (in parenthesis), and the third line presents the searched procedure parameter values. This table is about the fingerprint image shown in Figure 4. The best method(s) in each case is/are indicated by *italicized numbers*.

σ	TV	ASSK	ONLM	NEW
5	*12.4 (0.1)*	24.7 (0.1)	18.8 (0.1)	13.6 (0.2)
	0.0024 (0.0003)	0.0561 (0.0003)	0.0459 (0.0003)	0.0074 (0.0003)
	0.5	0.0025, 3	1, 1, 0.5	0.0043, 2.0, 0.025
10	*41.0 (0.3)*	85.3 (0.3)	63.3 (0.5)	*42.1 (0.4)*
	0.0131 (0.0006)	0.0917 (0.0007)	0.0174 (0.0005)	*0.0132 (0.0006)*
	0.2	0.0035, 4	10, 1, 15	*0.0043, 2.0, 0.2*
15	84.5 (0.6)	166.1 (0.7)	89.2 (0.6)	*82.9 (0.7)*
	0.0616 (0.0008)	0.1075 (0.0009)	0.0324 (0.0008)	*0.0136 (0.0008)*
	0.1	0.0040, 4	10, 1, 25	*0.0058, 3.0, 0.5*
20	135.9 (0.9)	257.6 (1.0)	*113.0 (0.9)*	134.0 (1.0)
	0.0641 (0.0012)	0.1071 (0.0013)	*0.0337 (0.0010)*	0.0129 (0.0011)
	0.08	0.0040, 4	*10, 1, 30*	0.0101, 3.0, 1.5

prominent pattern, compared to the other mean deviation images. Please note that in the mean deviation image of ASSK, although the non-random pattern is also weak, it contains
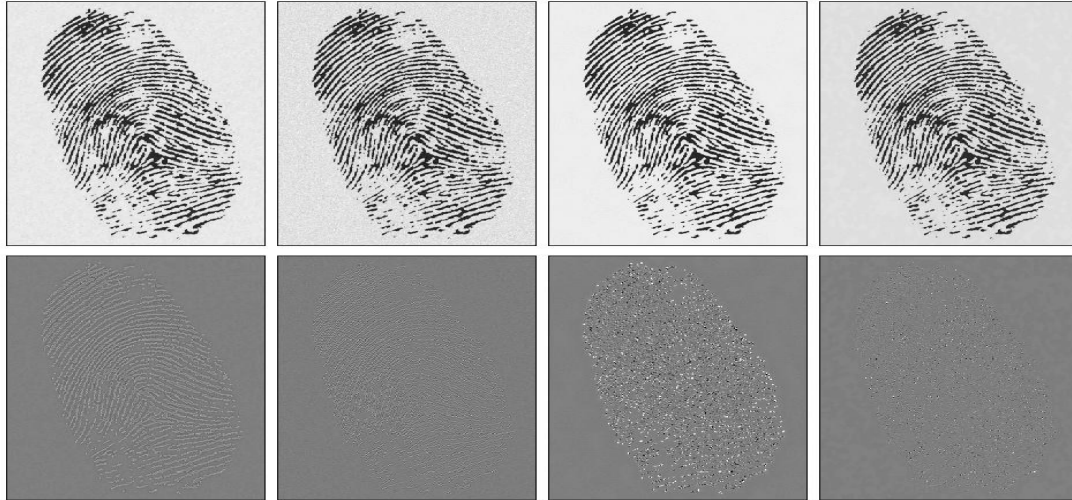
Figure 5: The first row shows the denoised fingerprint images by TV, ASSK, ONLM and NEW when $\sigma = 10$. The second row shows their mean deviation images.

Table 3: In each entry, the first line presents the estimated MISE value based on 100 simulations and the corresponding standard error (in parenthesis), the second line presents the value of EP and its standard error (in parenthesis), and the third line presents the searched procedure parameter values. This table is about the MRI brain image shown in Figure 4. The best method in each case is indicated by *italicized numbers*.

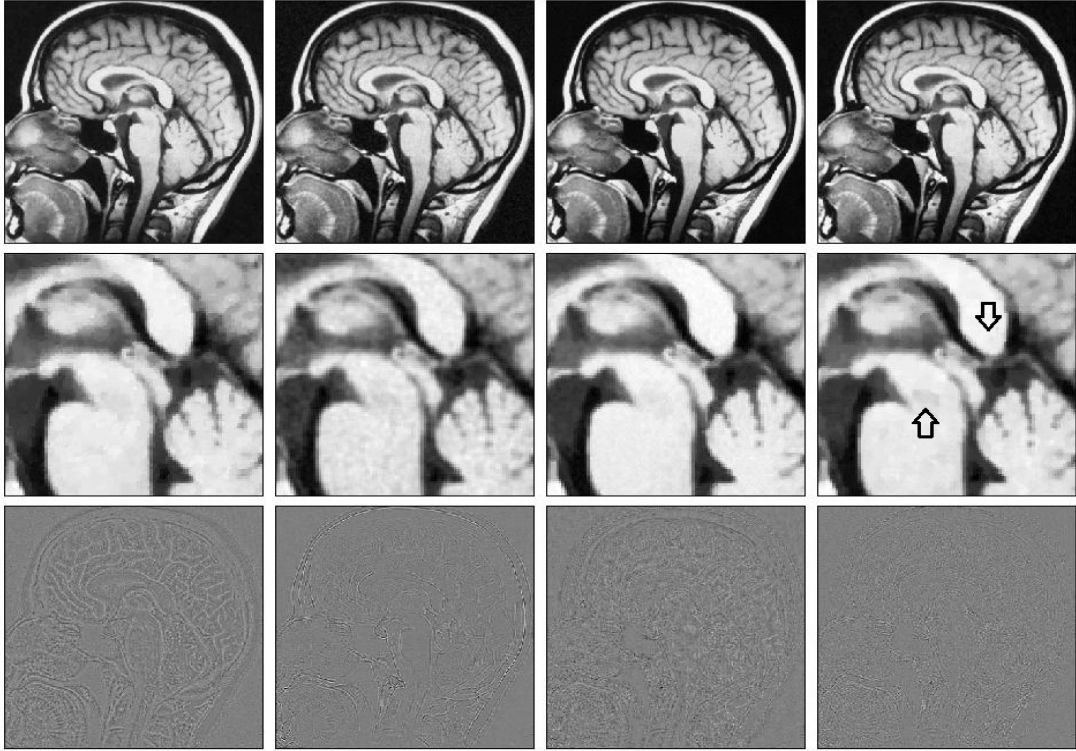| σ	TV	ASSK	ONLM	NEW
5	*14.4 (0.1)*	15.3 (0.1)	18.0 (0.1)	16.1 (0.2)
	0.0630 (0.0006)	0.0156 (0.0006)	0.0666 (0.0006)	0.0545 (0.0006)
	0.40	0.0045, 3	10, 1, 5	0.0088, 10.0, 0.5
10	38.7 (0.2)	38.8 (0.2)	34.0 (0.2)	*34.5 (0.2)*
	0.1253 (0.0009)	0.0182 (0.0009)	0.0861 (0.0010)	*0.0784 (0.0010)*
	0.15	0.0045, 5	10, 1, 10	*0.0088, 10.0, 2.0*
15	65.8 (0.5)	66.4 (0.4)	54.3 (0.5)	*56.0 (0.4)*
	0.1448 (0.0012)	0.0441 (0.0013)	0.0955 (0.0015)	*0.0822 (0.0016)*
	0.10	0.0055, 5	10, 1, 15	*0.0117, 10.0, 2.0*
20	93.8 (0.6)	93.3 (0.6)	*79.1 (0.6)*	77.5 (0.6)
	0.1739 (0.0017)	0.0461 (0.0016)	*0.0993 (0.0019)*	0.1172 (0.0017)
	0.07	0.0060, 5	*10, 1, 20*	0.0117, 10.0, 3.5

Figure 6: The first row shows the denoised MRI brain images by TV, ASSK, ONLM and NEW when $\sigma = 10$. The second row shows the zoomed-in middle portions of the denoised images. The third row shows the mean deviation images.

much noise. This observation is consistent with the results in Table 2 where the MISE value of ASSK is large when $\sigma = 10$. From Table 3, in the MRI brain image example, we see that NEW outperforms TV in terms of MISE and EP in all cases except the case when $\sigma = 5$ where TV is slightly better in terms of MISE. ASSK is much better than NEW at all noise levels in terms of EP, but it is only slightly better than NEW in terms of MISE when $\sigma = 5$ and worse than NEW at all other noise levels. The performance of ONLM and NEW is quite similar. From Figure 6, we can see that the denoised image by NEW is the best when $\sigma = 10$. This can be better seen from the images shown in the second row which are zoomed-in images of the middle portions of the ones shown in the first row. It can be seen that NEW removes noise well and preserves fine details better than its competitors. The mean deviation images in the third row of Figure 6 show that the one of NEW has the least prominent pattern compared to the ones of other methods, and the one of ASSK contains much noise.

In practice, the noise distribution may not be normal. Next, we consider the case when

17

the image is generated by a Poisson distributed observations as in cases of digital images and radiographic images. We still use the three test images as before. For the artificial image, since the image contrast is only 1.3125, we first multiply all the intensity values by 100 and round the resulting values to the nearest integers. Then, at each pixel, we generate a random integer from a Poisson distribution with the mean being the same as the true intensity value at that pixel. Finally, we scale the image back by dividing all generated intensity values by 100. For each pixel in the fingerprint and brain images, we directly generate random numbers from a Poisson distribution with the mean being the same as the true intensity value at that pixel. The first column of Figure 7 shows the noisy images. An interesting nature of these noisy images is the noise heterogeneity: the amount of noise in the background is smaller than that in the foreground. The image denoising methods are then applied to the noisy images in the same way as before. The corresponding results are presented in Table 4 and Figure 7. From Table 4, we see that NEW is better than its competitors in all cases in terms of MISE. Similar to the cases with the Gaussian noise, ASSK seems to preserve edges better in the case with the MRI brain image with the price of a weaker noise removal ability. This is confirmed by Figure 7, and the results are not much different from those in the Gaussian noise cases. The performance of the proposed method when the noise is uniform on a certain interval is similar, and the results are presented in the supplementary file.

Table 4: In each entry, the first line presents the estimated MISE value based on 100 simulations and the corresponding standard error (in parenthesis), the second line presents the value of EP and its standard error (in parenthesis), and the third line presents the searched procedure parameter values. This table is about the cases when the noise follows a Poisson distribution. The best method in each case is indicated by *italicized numbers*.

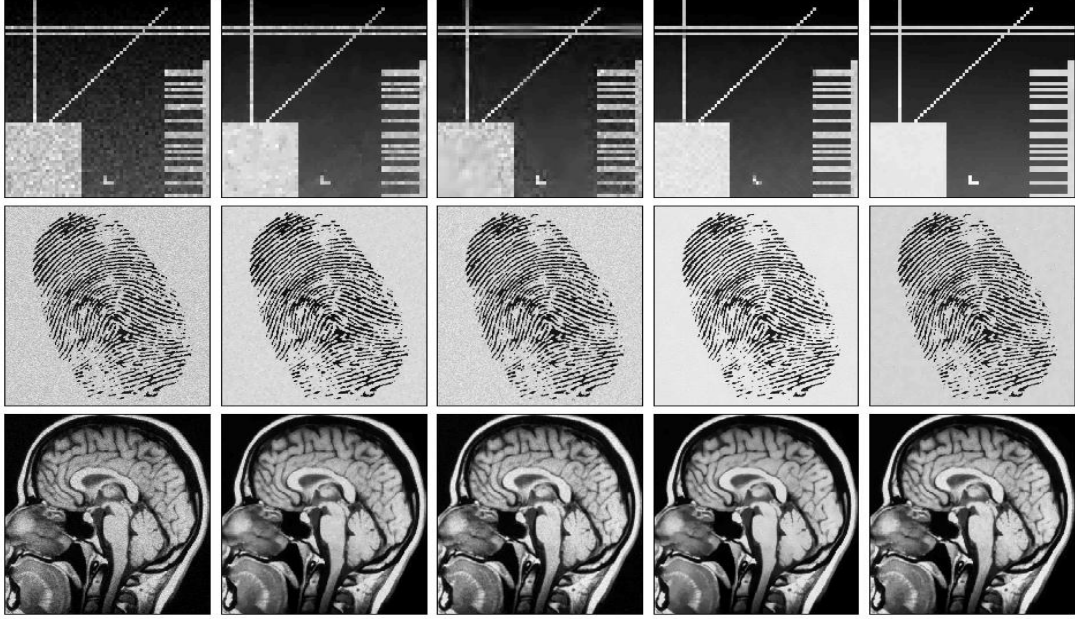| Image	TV	ASSK	ONLM	NEW
Artificial	0.0025 (0.0001)	0.0034 (0.0003)	0.0014 (0.0002)	*0.0002 (0.0001)*
	0.0116 (0.0058)	0.1182 (0.0100)	0.0105 (0.0061)	*0.0045 (0.0038)*
	25.0	0.13, 9	10, 1, 0.10	*0.0938, 2.0, 3.0*
Fingerprint	73.1 (0.5)	160.9 (0.7)	75.6 (0.5)	*68.4 (0.6)*
	0.0271 (0.0008)	0.1140 (0.0009)	0.0102 (0.0006)	*0.0192 (0.0009)*
	0.12	0.0040, 4	10, 1, 20	*0.0058, 3.0, 0.4*
Brain	47.5 (0.3)	41.5 (0.3)	41.6 (0.3)	*39.6 (0.3)*
	0.1289 (0.0010)	0.0304 (0.0011)	0.1296 (0.0012)	*0.0904 (0.0013)*
	0.14	0.0045, 5	5, 1, 15	*0.0117, 10.0, 2.0*

Figure 7: The columns from the left to the right present the noisy images and the denoised images by TV, ASSK, ONLM and NEW, respectively, in cases when the noise follows a Poisson distribution.

In Section 2.5, we proposed a CV procedure for choosing the parameters of the proposed image denoising method NEW. Next, we apply NEW to all the examples discussed so far involving Gaussian noise and choose its parameters by minimizing the CV score in (8). The results based on 100 replicated simulations are presented in Table 5. By comparing the results in this table with the corresponding ones in Tables 1–3, we can have several conclusions. (i) For the artificial image, NEW still outperforms the optimal performance of its competitors when $\sigma = 0.05$, 0.10 and 0.15 even if its parameters are chosen by the CV procedure. When $\sigma = 0.20$, the performance of NEW is similar to its optimal performance. (ii) For the fingerprint image, NEW is still better than ASSK and ONLM when $\sigma = 5$ and 10, it is better than ASSK when $\sigma = 15$, and it is better than TV and ASSK when $\sigma = 20$. (iii) For the MRI brain image, NEW is still better than ONLM when $\sigma = 5$, it is better than TV when $\sigma = 10$, it is better than TV and ASSK when $\sigma = 15$, and it is better than TV when $\sigma = 20$. The denoised images by NEW when its parameters are chosen by the CV procedure at the same noise levels considered by Figures 3, 5 and 6 are shown in Figure 8. By visual comparison, we can see that its denoised images in such cases look similar to those in the case when its parameters are chosen to be optimal.

Finally, we generalize our method directly to 3-D cases for denoising 3-D images. To

Table 5: In each entry, the first line presents the estimated MISE value based on 100 simulations and the corresponding standard error (in parenthesis), the second line presents the value of EP and its standard error (in parenthesis), and the third line presents the searched procedure parameter values by the CV procedure described in Section 2.5. This table considers the examples with Gaussian noise only.

Image	$\sigma = 0.05$	$\sigma = 0.10$	$\sigma = 0.15$	$\sigma = 0.20$
	0.0001 (0.0000)	0.0004 (0.0001)	0.0022 (0.0003)	0.0097 (0.0012)
Artificial	0.0034 (0.0025)	0.0095 (0.0061)	0.0810 (0.0103)	0.1712 (0.0212)
	0.0938, 2.0, 3.0	0.0938, 2.0, 2.0	0.0938, 2.0, 1.0	0.0938, 2.0, 1.0
	$\sigma = 5$	$\sigma = 10$	$\sigma = 15$	$\sigma = 20$
	13.6 (0.2)	46.1 (0.3)	95.7 (0.8)	137.4 (1.2)
Fingerprint	0.0124 (0.0003)	0.0268 (0.0006)	0.0076 (0.0009)	0.0049 (0.0013)
	0.0029, 3.0, 0.025	0.0029, 3.0, 0.2	0.0145, 3.0, 0.15	0.0173, 3.0, 0.3
	16.8 (0.1)	38.6 (0.3)	62.0 (0.4)	91.3 (0.8)
Brain	0.0176 (0.0007)	0.0666 (0.0011)	0.0844 (0.0014)	0.0937 (0.0021)
	0.0029, 10.0, 1.0	0.0175, 3.0, 1.0	0.0175, 3.0, 2.0	0.0175, 3.0, 2.5

Figure 8: The left, middle, and right panels present the denoised images by NEW when its parameters are chosen by the CV procedure described in Section 2.5 and when Gaussian noise with levels 0.10, 10, and 10, respectively, is added to the true test images.

check the performance of the proposed method in 3-D cases, we first download a noiseless T1-weighted 3-D brain phantom image from the BrainWeb database of the address http://www.bic.mni.mcgill.ca/brainweb/ (Cocosco et al. 1997, Kwan et al. 1996, 1999, Collins et al. 1998). This image has 20% intensity non-uniformity. We then consider a $32 \times 32 \times 32$ portion around the middle of the image to investigate the detail-preserving

ability of various denoising methods considered. The image contrast of that portion is 546.4886. Next, we generate its noisy version by adding random numbers from the distribution $N(0, 20^2)$. Then, we apply the 3-D versions of NEW, TV and ONLM to the noisy image in the same way as that in the 2-D case. Their MISE values based on 100 replicated simulations are 112.6, 155.0 and 119.4, respectively. Two cross-sections of the original 3-D image, its noisy version, and the denoised images are shown in Figure 9. From Figure 9, we see that NEW preserves image details better than its competitors in this case as well. In 3-D imaging applications, computation time is also an issue to consider because of a large number of voxels involved. Our proposed method can be improved in that regard if we can efficiently identify voxels where voxel clustering is necessary and we can improve the clustering algorithm. Moreover, in 3-D images, the edge structure can be complicated. Therefore, clustering voxels in a neighborhood into more than two groups can potentially improve the performance of our proposed method.
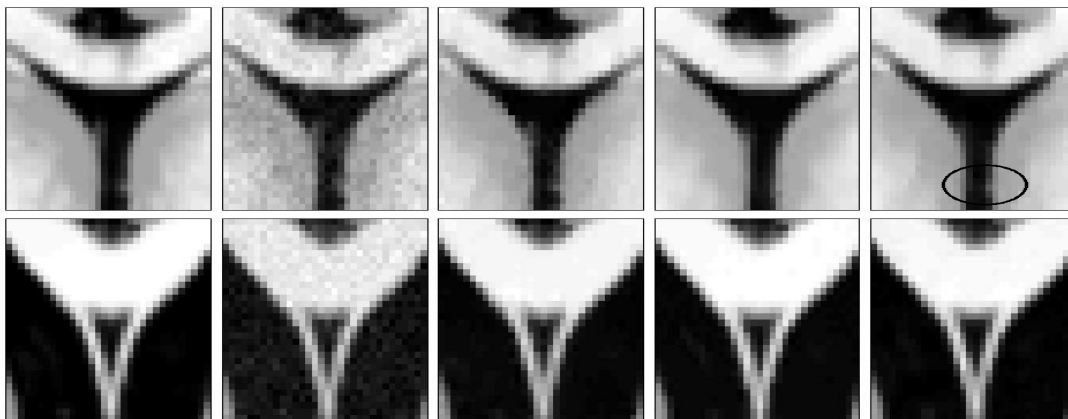

Figure 9: The columns from left to right present two cross-sections of a $32 \times 32 \times 32$ portion around the middle of a noiseless 3-D brain phantom, its noisy version and the denoised images by TV, ONLM and NEW, respectively.

5 Concluding Remarks

We have presented an image denoising procedure based on local pixel clustering. Theoretical properties and numerical results show that it should work well in applications. This procedure can be generalized to denoise other types of images used in medical science. However, there are still many places in this method that can be improved. For example, our proposed method uses the same smoothing parameters (e.g., h_n, u_n, and B_n) in the entire image. Intuitively, these parameter values should change over location, larger bandwidths

should be chosen in continuity regions of the image, and smaller bandwidths should be chosen around edges and other image structures. Computation time to denoise high resolution 3-D images should also be improved. Furthermore, in the current proposed image denoising procedure, pixels/voxels in a neighborhood of a given point are partitioned into two groups only. In some cases, specially in cases with 3-D images, three or more groups could be considered, although such a generalization would increase the computational complexity of the entire procedure. All these issues need to be addressed in our future research.

Acknowledgments

References

BARASH, D. (2002). A fundamental relationship between bilateral filtering, adaptive smoothing, and the nonlinear diffusion equation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. **24** 844–847.

BUADES, B., COLL, B. and MOREL, J.M. (2005). A review of image denoising algorithms, with a new one. *Multiscale Modeling & Simulation*. **4(2)** 490–530.

COCOSCO, C.A., KOLLOKIAN, V., KWAN, R.K.-S. and EVANS, A.C. (1997). Brain-Web: Online Interface to a 3D MRI Simulated Brain Database. *NeuroImage*. **5(4)**: part 2/4, S425, 1997 – *Proceedings of 3-rd International Conference on Functional Mapping of the Human Brain*; Copenhagen, May 1997.

COLLINS, D.L., ZIJDENBOS, A.P., KOLLOKIAN, V., SLED, J.G., KABANI, N.J., HOLMES, C.J. and EVANS, A.C. (1998). Design and Construction of a Realistic Digital Brain Phantom. *IEEE Transactions on Medical Imaging*. **17(3)** 463–468.

COUPE, P., YGER, P., PRIMA, S., HELLIER, P., KERVRANN, C. and BARILLOT, C. (2008). An optimized blockwise nonlocal means denoising filter for 3-D magnetic resonance images. *IEEE Transactions on Medical Imaging*. **27** 425–441.

CHU, C.K., GLAD, I.K., GODTLIEBSEN, F. and MARRON, J.S. (1998). Edge-preserving smoothers for image processing (with discussion). *Journal of the American Statistical Association*. **93** 526–556.

DEFAYS D. (1977). An efficient algorithm for a complete link method. *The Computer Journal (British Computer Society)* . **20(4)** 364–366.

DEMPSTER A.P., LAIRD N.M. and RUBIN D.B. (1977). Maximum Likelihood from Incomplete Data via the EM Algorithm. *Journal of the Royal Statistical Society. Series B (methodological)* . **39(1)** 1–38.

FELSBERG, M., FORSSEN, P. and SCHARR, H. (2006). Channel Smoothing: Efficient Robust Smoothing of Low-Level Signal Features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. **28(2)** 209–221.

FOI, A., KATKOVNIK, V., and EGIAZARIAN, K. (2007). Pointwise Shape-Adaptive DCT for High-Quality Denoising and Deblocking of Grayscale and Color Images. *IEEE Transactions on Image Processing.*, **16(5)** 1395–1411.

FRANKEN, E. and DUITS, R., P. (2009). Crossing-Preserving Coherence-Enhancing Diffusion on Invertible Orientation Scores. *International Journal of Computer Vision*. **85(3)** 253–278.

GEMAN, S. and GEMAN, D. (1984). "Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **6** 721–741.

GIJBEL, I., LAMBERT, A. and QIU, P. (2006). Edge-preserving image denoising and estimation of discontinuous surfaces. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. **28(7)** 1075–1087.

GODTLIEBSEN, F. and SEBASTIANI, G. (1994). Statistical methods for noisy images with discontinuities. *Journal of Applied Statistics*. **21** 459–477.

GONZALEZ, R.C. and WOODS, R.E. (1992). *Digital Image Processing*. Addison-Wesley Publishing Company Inc.

HALL, P. and ROBINSON, A.P. (2009). Reducing Variability of Crossvalidation for Smoothing-Parameter Choice. *Biometrika*. **96(1)** 175–186..

HALL, P. and QIU, P. (2007). Blind deconvolution and deblurring in image analysis. *Statistica Sinica*. **17** 1483–1509.

http://www.bic.mni.mcgill.ca/brainweb/.

KATKOVNIK, V., EGIAZARIAN, K., and ASTOLA, J. (2006). *Local Approximation Techniques in Signal and Image Processing*. SPIE Press book. PM 157.

KEELING, S. (2003). Total variation based convex filters for medical imaging. *Applied Mathematics and Computation*. **139(1)** 101–119.

KWAN, R.K.-S., EVANS, A.C. and PIKE, G.B. (1996). An Extensible MRI Simulator for Post-Processing Evaluation. *Visualization in Biomedical Computing* 1996; *Lecture Notes in Computer Science*, **1131** 135–140. Springer-Verlag.

KWAN, R.K.-S., EVANS, A.C. and PIKE, G.B. (1999). MRI simulation-based evaluation of image-processing and classification methods. *IEEE Transactions on Medical Imaging*. **18(11)** 1085–1097.

LLOYD, S.P. (1982). Least square quantization in PCM. *IEEE Transactions on Information Theory*. **28(2)** 129–137. (a Bell Telephone Laboratories Paper in 1957.)

LOADER, C.R. (1999). Bandwidth Selection: Classical or Plug-In?. *The Annals of Statistics*. **27(2)**, 415–438.

MACQUEEN, J.B. (1967). Some Methods for classification and Analysis of Multivariate Observations. *Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability*. 281–297.

MARRON, J.S. (1988). Automatic Smoothing Parameter Selection: A Survey. *Empirical Economics*. **13(3–4)**, 281–208.

MUKHERJEE, P.S. and QIU, P. (2011). 3-D Image Denoising By Local Smoothing And Nonparametric Regression. *Technometrics*. **53(2)**, 196–208.

PERONA, P. and MALIK, J.(1990). Scale-space and edge detection using anisotropic diffusion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. **12(7)** 629–639.

POLZEHL, J. and SPOKOINY, V.G. (2000). Adaptive weights smoothing with applications to image restoration. *Journal of the Royal Statistical Society (Series B)*. **62**, 335–354.

PORTILLA, J., STRELA, V., WAINWRIGHT, M. and SIMONCELLI, E.P. (2003). Image denoising using scale mixtures of gaussians in the wavelet domain. *IEEE Transactions on Image Processing*. **12(11)** 1338–1351.

QIU, P. (1998). Discontinuous regression surfaces fitting. *The Annals of Statistics*, **26**, 2218-2245.

QIU, P. (2005). *Image Processing and Jump Regression Analysis*. New York, John Wiley.

QIU, P. (2007). Jump surface estimation, edge detection, and image restoration. *Journal of the American Statistical Association*. **102** 745–756.

QIU, P. (2009). Jump-preserving surface reconstruction from noisy data. *Annals of the Institute of Statistical Mathematics*, **61**, 715–751.

QIU, P. and MUKHERJEE, P.S. (2010). Edge Structure Preserving Image Denoising. *Signal Processing*. **90(10)** 2851–2862.

QIU, P. and MUKHERJEE, P.S. (2012). Edge Structure Preserving 3-D Image Denoising By Local Surface Approximation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*. **34(8)** 1457–1468.

RUDIN, L., OSHER, S. and FATEMI, E. (1992). Nonlinear total variation based noise removal algorithms. *Physica D*. **60** 259–268.

SIBSON R. (1973). SLINK: an optimally efficient algorithm for the single-link cluster method. *The Computer Journal (British Computer Society)*. **16(1)** 30–34.

TER HAAR ROMENY, B. (2003). *Front-End Vision and Multi-Scale Image Analysis*. Springer.

TAKEDA, H., FARSIU, S. and MILANFAR, P. (2007). Kernel regression for image processing and reconstruction. *IEEE Transactions on Image Processing*. **16(2)** 349–366.

TOMASI, C. and MANDUCHI, R. (1998). Bilateral filtering for gray and color images. *Proceedings of the 1998 IEEE International Conference on Computer Vision*. 839–846. Bombay, India.

WANG, Y. and ZHOU, H. (2006). Total variation wavelet-based medical image denoising. *International Journal of Biomedical Imaging*. **2006** 1–6.

WARD J.H. (1963). Hierarchical Grouping to Optimize an Objective Function. *Journal of the American Statistical Association*. **58(301)** 236–244.