Salt-and-Pepper Noise Removal by Median-type Noise Detectors and Detail-preserving Regularization

Raymond H. Chan, Chung-Wa Ho, and Mila Nikolova

Abstract

This paper proposes a two-phase scheme for removing salt-and-pepper impulse noise. In the first phase, an adaptive median filter is used to identify pixels which are likely to be contaminated by noise (noise candidates). In the second phase, the image is restored using a specialized regularization method that applies only to those selected noise candidates. In terms of edge preservation and noise suppression, our restored images show a significant improvement compared to those restored by using just nonlinear filters or regularization methods only. Our scheme can remove salt-and-pepper-noise with noise level as high as 90%.

Index Terms

Impulse noise, adaptive median filter, edge-preserving regularization.

I. INTRODUCTION

Impulse noise is caused by malfunctioning pixels in camera sensors, faulty memory locations in hardware, or transmission in a noisy channel. See [5] for instance. Two common types of impulse noise are the salt-and-pepper noise and the random-valued noise. For images corrupted by salt-and-pepper noise (respectively random-valued noise), the noisy pixels can take only the maximum and the minimum

values (respectively any random value) in the dynamic range. There are many works on the restoration of images corrupted by impulse noise. See, for instance, the nonlinear digital filters reviewed in [1]. The median filter was once the most popular nonlinear filter for removing impulse noise, because of its good denoising power [5] and computational efficiency [16]. However, when the noise level is over 50%, some details and edges of the original image are smeared by the filter [20].

Different remedies of the median filter have been proposed, e.g. the adaptive median filter [17], the multi-state median filter [11], or the median filter based on homogeneity information [12], [21]. These so-called "*decision-based*" or "*switching*" filters first identify possible noisy pixels and then replace them by using the median filter or its variants, while leaving all other pixels unchanged. These filters are good at *detecting* noise even at a high noise level. Their main drawback is that the noisy pixels are replaced by some median value in their vicinity without taking into account local features such as the possible presence of edges. Hence details and edges are not recovered satisfactorily, especially when the noise level is high.

For images corrupted by Gaussian noise, least-squares methods based on edge-preserving regularization functionals [4], [9], [10], [22] have been used successfully to preserve the edges and the details in the images. These methods fail in the presence of impulse noise because the noise is heavy tailed. Moreover the restoration will alter basically all pixels in the image, including those that are not corrupted by the impulse noise. Recently, non-smooth data-fidelity terms (e.g. ℓ_1) have been used along with edge-preserving regularization to deal with impulse noise [19].

In this paper, we propose a powerful two-stage scheme which combines the variational method proposed in [19] with the adaptive median filter [17]. More precisely, the noise candidates are first identified by the adaptive median filter, and then these noise candidates are selectively restored using an objective function with an ℓ_1 data-fidelity term and an edge-preserving regularization term. Since the edges are preserved for the noise candidates, and no changes are made to the other pixels, the performance of our combined approach is much better than that of either one of the methods. Salt-and-pepper noise with noise ratio as high as 90% can be cleaned quite efficiently.

The outline of the paper is as follows. The adaptive median filter and the edge-preserving method are reviewed in Section II. Our denoising scheme is presented in Section III. Experimental results and conclusions are presented in Sections IV and V, respectively.

II. ADAPTIVE MEDIAN FILTER AND EDGE-PRESERVING REGULARIZATION

A. Review of the adaptive median filter

Let $x_{i,j}$, for $(i,j) \in \mathcal{A} \equiv \{1,\ldots,M\} \times \{1,\ldots,N\}$, be the gray level of a true M-by-N image $\mathbf{x}$ at pixel location (i,j), and $[s_{\min}, s_{\max}]$ be the dynamic range of $\mathbf{x}$, i.e. $s_{\min} \leq x_{i,j} \leq s_{\max}$ for all $(i,j) \in \mathcal{A}$. Denote by $\mathbf{y}$ a noisy image. In the classical salt-and-pepper impulse noise model, the observed gray level at pixel location (i,j) is given by

$$
y_{i,j} = \begin{cases}
s_{\min}, & \text{with probability } p, \\
s_{\max}, & \text{with probability } q, \\
x_{i,j}, & \text{with probability } 1 - p - q,
\end{cases}
$$

where $r = p + q$ defines the noise level. Here we give a brief review of the filter.

Let $S_{i,j}^w$ be a window of size $w \times w$ centered at (i,j), i.e.

$$
S_{i,j}^w = \{(k,l) \ : \ |k - i| \leq w \text{ and } |j - l| \leq w\}
$$

and let $w_{\max} \times w_{\max}$ be the maximum window size. The algorithm tries to identify the noise candidates $y_{i,j}$, and then replace each $y_{i,j}$ by the median of the pixels in $S_{i,j}^w$.

Algorithm I (Adaptive Median Filter): For each pixel location (i,j), do

1. Initialize $w = 3$.
2. Compute $s_{i,j}^{\min,w}$, $s_{i,j}^{\mathrm{med},w}$ and $s_{i,j}^{\max,w}$, which are the minimum, median and maximum of the pixel values in $S_{i,j}^w$ respectively.
3. If $s_{i,j}^{\min,w} < s_{i,j}^{\mathrm{med},w} < s_{i,j}^{\max,w}$, then go to Step 5. Otherwise, set $w = w + 2$.
4. If $w \leq w_{\max}$ go to Step 2. Otherwise, we replace $y_{i,j}$ by $s_{i,j}^{\mathrm{med},w_{\max}}$.
5. If $s_{i,j}^{\min,w} < y_{i,j} < s_{i,j}^{\max,w}$, then $y_{i,j}$ is not a noise candidate, else we replace $y_{i,j}$ by $s_{i,j}^{\mathrm{med},w}$.

The adaptive structure of the filter ensures that most of the impulse noise are detected even at a high noise level provided that the window size is large enough. Notice that the noise candidates are replaced by the median $s_{i,j}^{\mathrm{med},w}$, while the remaining pixels are left unaltered.

B. Variational method for impulse noise cleaning

In [19], images corrupted by impulse noise are restored by minimizing a convex objective function $F_{\mathbf{y}} : \mathbb{R}^{M \times N} \rightarrow \mathbb{R}$ of the form

$$
\begin{aligned}
F_{\mathbf{y}}(\mathbf{u}) = \sum_{(i,j) \in \mathcal{A}} |u_{i,j} - y_{i,j}| \\
+ \frac{\beta}{2} \sum_{(i,j) \in \mathcal{A}} \sum_{(m,n) \in \mathcal{V}_{i,j}} \varphi(u_{i,j} - u_{m,n}),
\end{aligned}
\tag{1}
$$

where $\mathcal{V}_{i,j}$ is the set of the four closest neighbors of (i, j), not including (i, j). It was shown in [18] and [19] that under mild assumptions and a pertinent choice of β, the minimizer $\hat{\mathbf{u}}$ of $F_{\mathbf{y}}$ satisfies $\hat{u}_{i,j} = y_{i,j}$ for most of the uncorrupted pixels $y_{i,j}$. Furthermore, all pixels $\hat{u}_{i,j}$ such that $\hat{u}_{i,j} \neq y_{i,j}$ are restored so that edges and local features are well preserved, provided that φ is an edge-preserving potential function. Examples of such functions are:

$$
\varphi(t) = \sqrt{\alpha + t^2}, \quad \alpha > 0,
$$

$$
\varphi(t) = |t|^{\alpha}, \quad 1 < \alpha \leq 2,
$$

see [3], [4], [10], [14]. The minimization algorithm works on the residuals $\mathbf{z} = \mathbf{u} - \mathbf{y}$. It is sketched below:

Algorithm II:

1. Initialize $z_{ij}^{(0)} = 0$ for each $(i, j) \in \mathcal{A}$.

2. At each iteration k, calculate, for each $(i, j) \in \mathcal{A}$,

$$
\xi_{i,j}^{(k)} = \beta \sum_{(m,n) \in \mathcal{V}_{i,j}} \varphi'(y_{i,j} - z_{m,n} - y_{m,n}),
$$

where $z_{m,n}$, for $(m, n) \in \mathcal{V}_{i,j}$, are the latest updates and φ' is the derivative of φ.

3. If $\left|\xi_{i,j}^{(k)}\right| \leq 1$, set $z_{i,j}^{(k)} = 0$. Otherwise, solve for $z_{i,j}^{(k)}$ in the nonlinear equation

$$
\beta \sum_{(m,n) \in \mathcal{V}_{i,j}} \varphi'(z_{i,j}^{(k)} + y_{i,j} - z_{m,n} - y_{m,n}) = \mathrm{sign}(\xi_{i,j}^{(k)}).
\tag{2}
$$

The updating of $z_{i,j}^{(k)}$ can be done in a red-black fashion, and it was shown in [19] that $\mathbf{z}^{(k)}$ converges to $\hat{\mathbf{z}} = \hat{\mathbf{u}} - \mathbf{y}$, where the restored image $\hat{\mathbf{u}}$ minimizes $F_{\mathbf{y}}$ in (1). If we choose $\varphi(t) = |t|^{\alpha}$, the nonlinear equation (2) can be solved by Newton's method with quadratic convergence by using a suitable initial guess derived in [6].

III. Our method

Many denoising schemes are "*decision-based*" median filters, see for example, [11], [12], [24]. This means that the noise candidates are first detected by some rules and are replaced by the median output or its variants. For instance, in Algorithm I, the noise candidate $y_{i,j}$, $(i,j) \in \mathcal{N}$, is replaced by $s_{i,j}^{\mathrm{med},w}$. These schemes are good because the uncorrupted pixels will not be modified. However, the replacement methods in these denoising schemes cannot preserve the features of the images, in particular the edges are smeared.

In contrast, Algorithm II can preserve edges during denoising but it has problem in detecting noisy patches, i.e., a connected region containing many noisy pixels. If one wishes to smooth out all the noisy patches, one has to increase β, see [7] for the role of β. As a result, the values of some pixels near edges will be distorted.

Combining both methods will avoid the drawbacks of either one of them. The aims of our method are to correct noisy pixels and preserve edges in the image. In the following, we denote the restored image by $\hat{\mathbf{x}}$.

Algorithm III:

1. *(Noise detection):* Denote by $\tilde{\mathbf{y}}$ the image obtained by applying an adaptive median filter to the noisy image $\mathbf{y}$. Noticing that noisy pixels take their values in the set $\{s_{\min}, s_{\max}\}$, we define the noise candidate set as

$$\mathcal{N} = \{(i,j) \in \mathcal{A} : \tilde{y}_{i,j} \neq y_{i,j} \text{ and } y_{i,j} \in \{s_{\min}, s_{\max}\}\}.$$

   The set of all uncorrupted pixels is $\mathcal{N}^c = \mathcal{A} \setminus \mathcal{N}$.

2. *(Replacement):* Since all pixels in $\mathcal{N}^c$ are detected as uncorrupted, we naturally keep their original values, i.e., $\hat{x}_{i,j} = y_{i,j}$ for all $(i,j) \in \mathcal{N}^c$. Let us now consider a noise candidate, say, at $(i,j) \in \mathcal{N}$. Each one of its neighbors $(m,n) \in \mathcal{V}_{i,j}$ is either a correct pixel, i.e., $(m,n) \in \mathcal{N}^c$ and hence $\hat{x}_{m,n} = y_{m,n}$; or is another noise candidate, i.e., $(m,n) \in \mathcal{N}$, in which case its value must be restored. The neighborhood $\mathcal{V}_{i,j}$ of (i,j) is thus split as $\mathcal{V}_{i,j} = (\mathcal{V}_{i,j} \cap \mathcal{N}^c) \cup (\mathcal{V}_{i,j} \cap \mathcal{N})$. Noise candidates are restored by minimizing a functional of the form (1), but restricted to the noise candidate set $\mathcal{N}$:

$$F_{\mathbf{y}}\big|_{\mathcal{N}}(\mathbf{u}) = \sum_{(i,j)\in\mathcal{N}} \left[|u_{i,j} - y_{i,j}| + \frac{\beta}{2}(S_1 + S_2) \right] \tag{3}$$

where

$$S_1 = \sum_{(m,n)\in\mathcal{V}_{i,j}\cap\mathcal{N}^c} 2\cdot\varphi(u_{i,j}-y_{m,n}),$$

$$S_2 = \sum_{(m,n)\in\mathcal{V}_{i,j}\cap\mathcal{N}} \varphi(u_{i,j}-u_{m,n}).$$

The restored image $\hat{\mathbf{x}}$ with indices $(i,j)\in\mathcal{N}$ is the minimizer of (3) which can be obtained by using Algorithm II but restricted onto $\mathcal{N}$ instead of onto $\mathcal{A}$. As in (1), the ℓ_1 data-fidelity term $|u_{i,j}-y_{i,j}|$ discourages those wrongly detected uncorrupted pixels in $\mathcal{N}$ from being modified to other values. The regularization term (S_1+S_2) performs edge-preserving smoothing for the pixels indexed by $\mathcal{N}$.

Let us emphasize that Step 1 of our method can be realized by any reliable impulse noise detector, such as the multi-state median filter [11] or the improved detector [24], etc. Our choice, the adaptive median filter, was motivated by the fact that it provides a good compromise between simplicity and robust noise detection, especially for high level noise ratios. The pertinence of this choice can be seen from the experimental results in [13] (where the noise level is 50%) or Figures 3(h) and 4(h) (where the noise level is 70%).

IV. SIMULATIONS

A. Configuration

Among the commonly tested 512-by-512 8-bit gray-scale images, the one with homogeneous region (*Lena*) and the one with high activity (*Bridge*) will be selected for our simulations. Their dynamic ranges are $[0,255]$. In the simulations, images will be corrupted by "salt" (with value 255) and "pepper" (with value 0) noise with equal probability. Also a wide range of noise levels varied from 10% to 70% with increments of 10% will be tested. Restoration performances are quantitatively measured by the peak signal-to-noise ratio (PSNR) and the mean absolute error (MAE) defined in [5, p. 327]:

$$\text{PSNR} = 10\log_{10}\frac{255^2}{\frac{1}{MN}\sum_{i,j}(r_{i,j}-x_{i,j})^2},$$

$$\text{MAE} = \frac{1}{MN}\sum_{i,j}|r_{i,j}-x_{i,j}|,$$

where $r_{i,j}$ and $x_{i,j}$ denote the pixel values of the restored image and the original image, respectively.

For Algorithm I (the adaptive median filter), the maximum window size $w_{\max}$ should be chosen such that it increases with the noise level in order to filter out the noise. Since it is not known a priori, we

TABLE I

MAXIMUM WINDOW SIZE $w_{\max}$ IN ALGORITHM I.

noise level	$w_{\max} \times w_{\max}$
$r < 25\%$	5×5
$25\% \leq r < 40\%$	7×7
$40\% \leq r < 60\%$	9×9
$60\% \leq r < 70\%$	13×13
$70\% \leq r < 80\%$	17×17
$80\% \leq r < 85\%$	25×25
$85\% \leq r \leq 90\%$	39×39

tried different $w_{\max}$ for any given noise level, and found that $w_{\max}$ given in Table I are sufficient for the filtering. We therefore set $w_{\max} = 39$ in all our tests. We remark that with such choice of $w_{\max}$, almost all the salt-and-pepper noise are detected in the filtered images.

For Algorithm II (the variational method in [19]), we choose $\varphi(t) = |t|^\alpha$ as the edge-preserving function. We observe that if α is small ($1 < \alpha < 1.1$), most of the noise is suppressed but staircases appear. If α is large ($\alpha > 1.5$), the fine details are not distorted seriously but the noise cannot be fully suppressed. The selection of α is a trade-off between noise suppression and detail preservation [19]. In the tests, the best restoration results are not sensitive to α when it is between 1.2 and 1.4. We therefore choose $\varphi(t) = |t|^{1.3}$, and β is tuned to give the best result in terms of PSNR.

For our proposed Algorithm III, the noise candidate set $\mathcal{N}$ should be obtained such that most of the noise are detected. This again amounts to the selection of $w_{\max}$. As mentioned, $w_{\max} = 39$ can be fixed for most purposes. Then we can restore those noise pixels $y_{i,j}$ with $(i,j) \in \mathcal{N}$. As in Algorithm II, the edge-preserving function $\varphi(t) = |t|^{1.3}$ will be used. That leaves only the parameter β to be determined. Later, we will demonstrate that our proposed algorithm is very robust with respect to β and thus we fix $\beta = 5$ in all the tests.

For comparison purpose, Algorithm I, Algorithm II, the standard median (MED) filter, and also recently proposed filters like the progressive switching median (PSM) filter [23], the multi-state median (MSM) filter [11], the noise adaptive soft-switching median (NASM) filter [12], the directional difference-based switching median (DDBSM) filter [15], and the improved switching median (ISM) filter [24] are also tested. For MED filter, the window sizes are chosen for each noise level to achieve its best performance. For MSM filter, the maximum center weights of 7, 5 and 3 are tested for each noise level. For ISM filter,
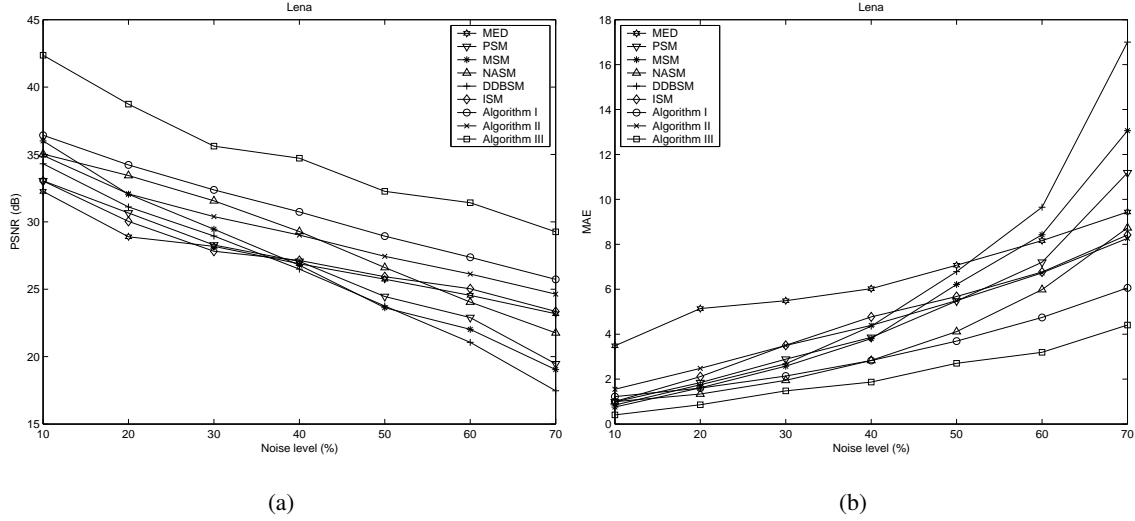
Fig. 1.   Results in PSNR and MAE for the *Lena* image at various noise levels for different algorithms.

the convolution kernels K_5, K_7 and K_9 and filtering window sizes of 9×9 and 11×11 are used. The decision thresholds in PSM, MSM, DDBSM, ISM filters are also tuned to give the best performance in terms of PSNR.

B. Denoising Performance

We summarize the performance of different methods in Figures 1 and 2. From the plots, we see that all the methods have similar performance when the noise level is low. This is because those recently proposed methods focus on the noise detection. However when the noise level increases, noise patches will be formed and they may be considered as noise free pixels. This causes difficulties in the noise detection algorithm. With erroneous noise detection, no further modifications will be made to the noise patches, and hence their results are not satisfactory.

On the other hand, our proposed denoising scheme achieves a significantly high PSNR and low MAE even when the noise level is high. This is mainly based on the accurate noise detection by the adaptive median filter and the edge-preserving property of the variational method of [19].

In Figures 3 and 4, we present restoration results for the 70% corrupted *Lena* and *Bridge* images. Among the restorations, except for our proposed one, Algorithm I gives the best performance in terms of noise suppression and details preservation. As mentioned before, it is because the algorithm locates the noise accurately. In fact, about 70.2% and 70.4% pixels are detected as noise candidates in *Lena* and *Bridge* respectively by Algorithm I. However, the edges are jittered by the median filter. For Algorithm
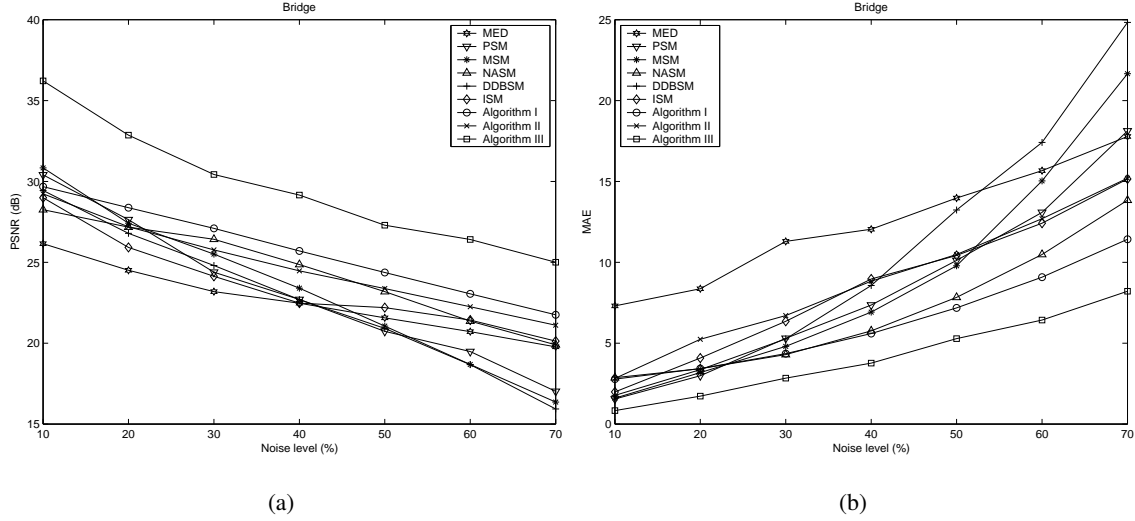
Fig. 2. Results in PSNR and MAE for the *Bridge* image at various noise levels for different algorithms.

II, much of the noise is suppressed but the blurring and distortion are serious. This is because every pixel has to be examined and may have been altered. Compared with all the algorithms tested, our proposed Algorithm III is the best one. It has successfully suppressed the noise with the details and the edges of the images being preserved very accurately.

Finally, to demonstrate the excellent performance of our proposed filter, 90% corrupted *Lena* and *Bridge* are restored by Algorithm I and by our Algorithm III, see Figure 5. We can clearly see the visual differences and also the improvement in PSNR by using our algorithm.

C. Robustness with respect to β

For Algorithm II, the choice of β is crucial in the restoration. To show that our Algorithm III is robust with respect to β, $0.5 \leq \beta \leq 10$ are tested for noise levels 30%, 50% and 70%, see Figure 6. From the plots, we see that the PSNR is very stable when $1 \leq \beta \leq 10$. Hence one can set $\beta = 5$ for all denoising problems in practice. If one further use $\varphi(t) = |t|^{1.3}$ as we did in our tests, and set $w_{\max} = \min\{M, N\}$ (which will be able to detect all salt-and-pepper noise), then our algorithm is parameter free.

D. Computational Complexity

We end this section by considering the complexity of our algorithm. Our algorithm requires two phases: noise detection and replacement. Noise detection is done by Algorithm I, the adaptive median filter. Like other median-type filters, it is relatively fast. Although $w_{\max}$ may be quite large, the loop in
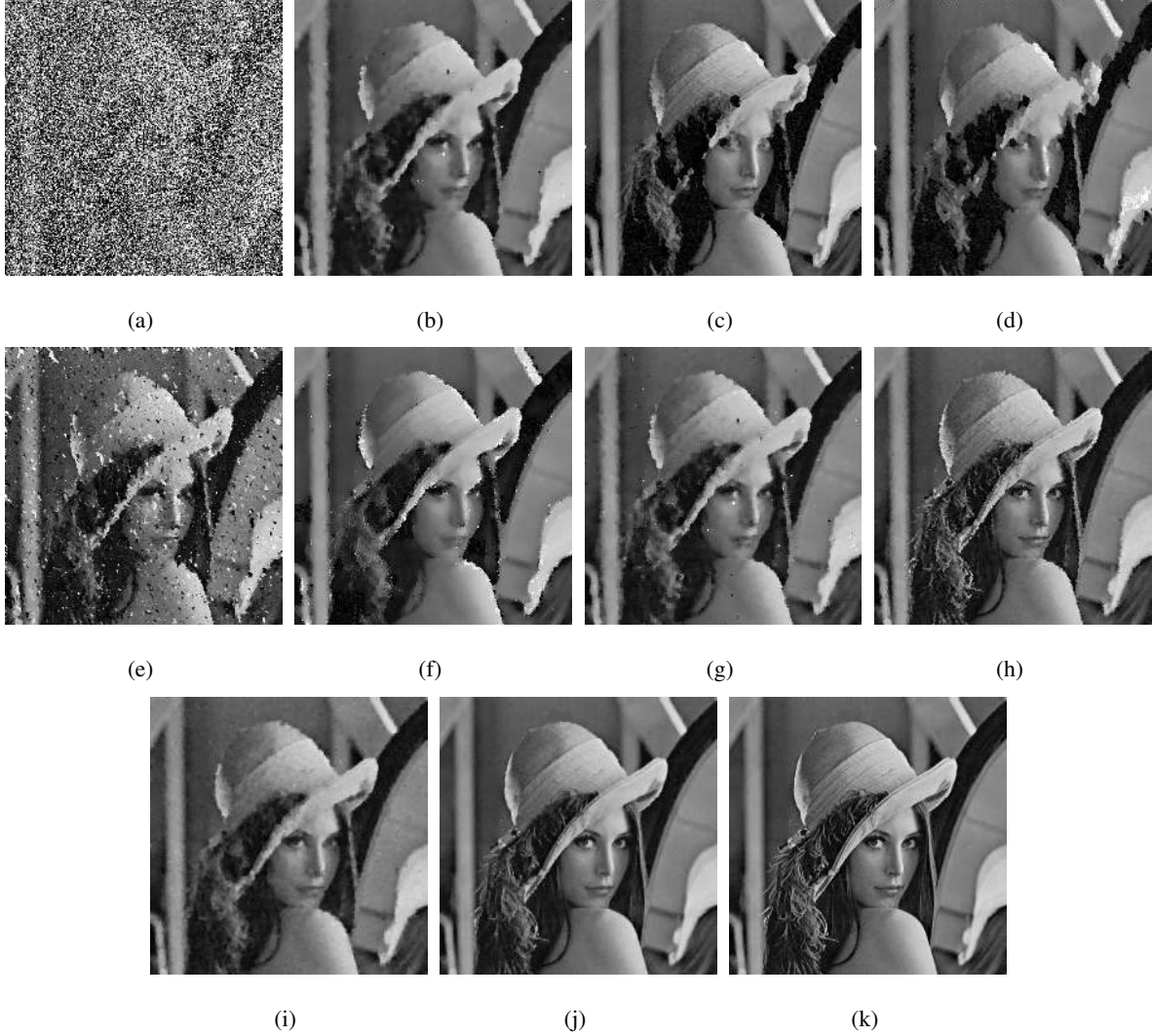
Fig. 3. Restoration results of different filters: (a) Corrupted *Lena* image with 70% salt-and-pepper noise (6.7 dB), (b) MED filer (23.2 dB), (c) PSM filter (19.5 dB), (d) MSM filter (19.0 dB), (e) DDBSM filter (17.5 dB), (f) NASM filter (21.8 dB), (g) ISM filter (23.4 dB), (h) Algorithm I (25.8 dB), (i) Algorithm II (24.6 dB), (j) Our proposed algorithm (29.3 dB), and (k) Original image.

Algorithm I is automatically stopped at Step 3 when the noise level is not high. The replacement step is the most time-consuming part of our algorithm as it requires the minimization of the functional in (3). It is equivalent to solving the nonlinear equation (2) for each pixel in the noise candidate set, see [6]. In Table II, we compare the CPU time needed for all three algorithms when MATLAB 6.5 (R13) is used on a PC equipped with AMD 1.8 GHz CPU and 224MB RAM memory. We see that our Algorithm III is about 20 to 90 times slower than Algorithm I.
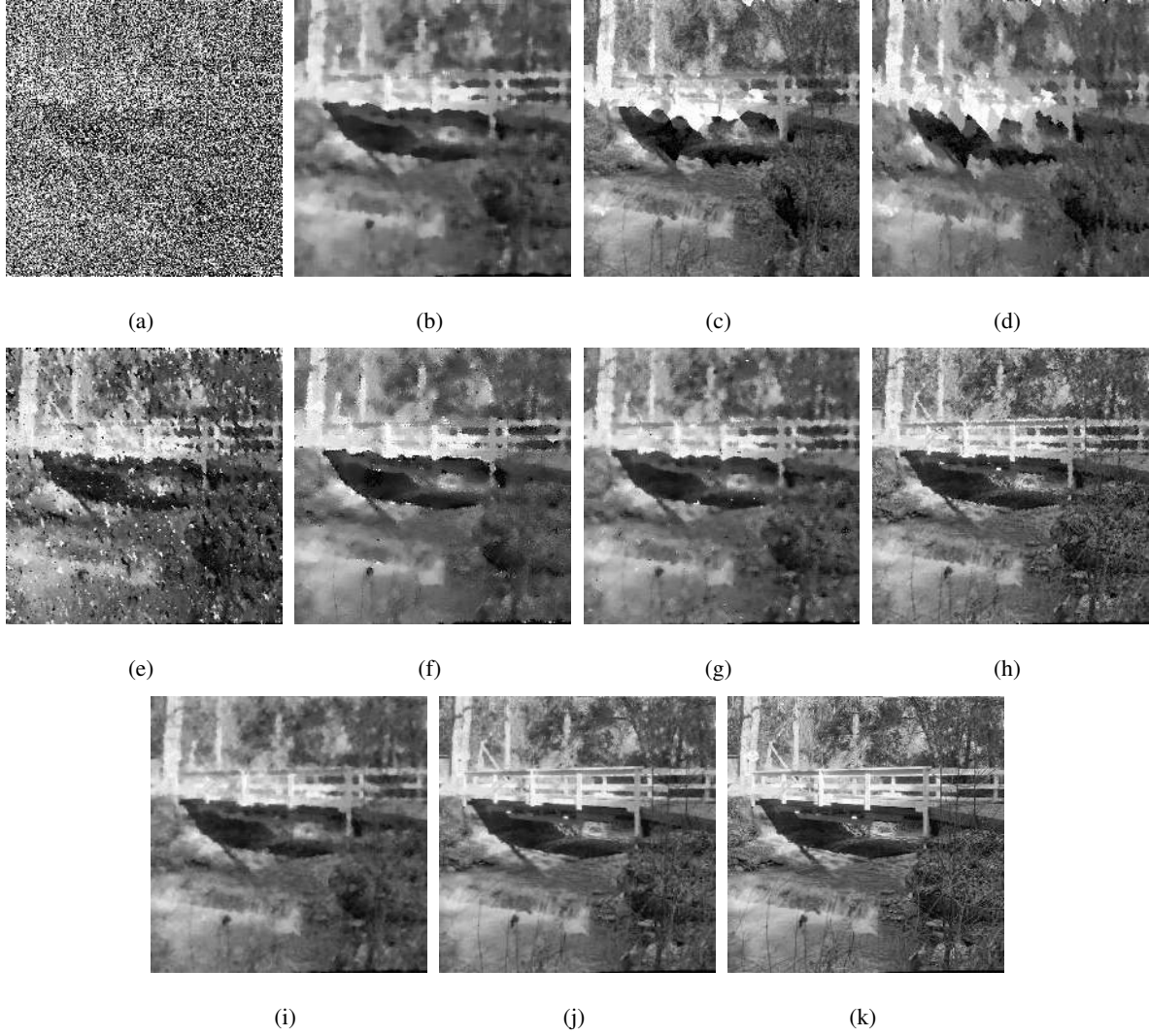
Fig. 4. Restoration results of different filters: (a) Corrupted *Bridge* image with 70% salt-and-pepper noise (6.8 dB), (b) MED filer (19.8 dB), (c) PSM filter (17.0 dB), (d) MSM filter (16.4 dB), (e) DDBSM filter (15.9 dB), (f) NASM filter (19.9 dB), (g) ISM filter (20.1 dB), (h) Algorithm I (21.8 dB), (i) Algorithm II (21.1 dB), (j) Our proposed algorithm (25.0 dB), and (k) Original image.

TABLE II

COMPARISON OF CPU TIME IN SECONDS.

Image	Noise level	Algo. I	Algo. II	Algo. III
Lena	70%	23	6865	2009
	90%	311	> 12000	6917
Bridge	70%	56	8003	4263
	90%	311	> 12000	10070

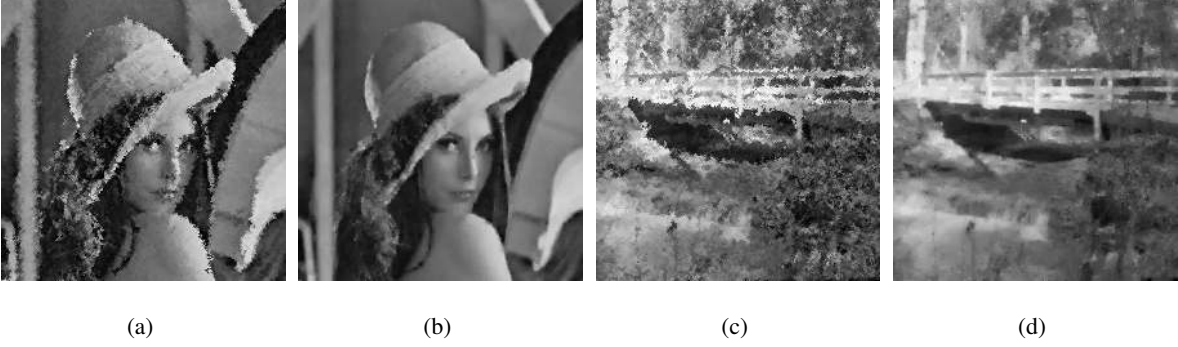|			
(a)	(b)	(c)	(d)

Fig. 5. Restorations of 90% corrupted images: (a) *Lena* by Algorithm I (21.1 dB), (b) *Lena* by Algorithm III (25.4 dB), (c) *Bridge* by Algorithm I (18.1 dB), and (d) *Bridge* by Algorithm III (21.5 dB).
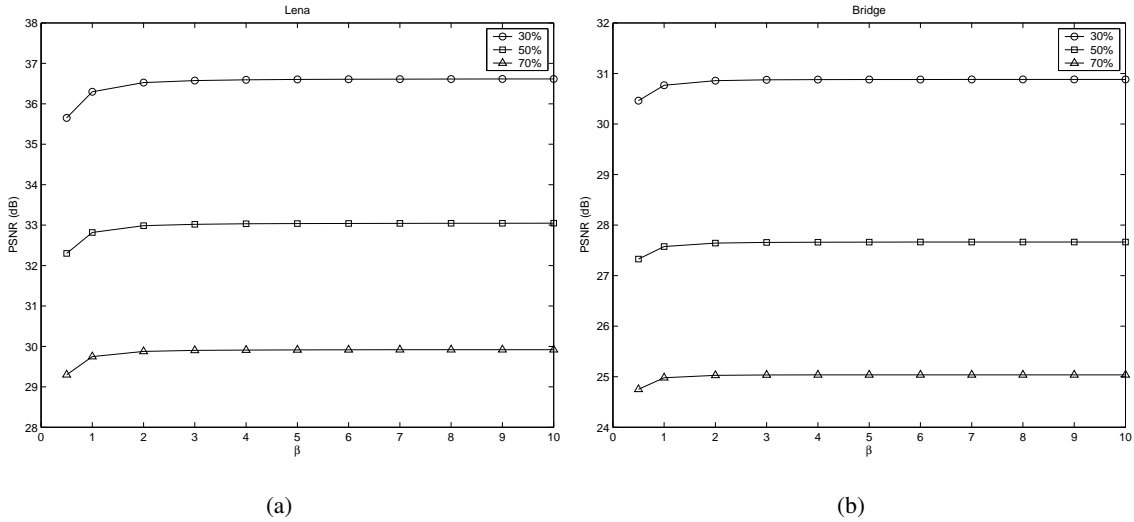

(a)	(b)

Fig. 6. PSNR of restored images by our Algorithm III for different β: (a) *Lena* image and (b) *Bridge* image.

We emphasize however that the main contribution of our paper is a method that is capable of restoring images corrupted by salt-and-pepper noise with extremely high noise ratio. Our method can be used as a post-processing image enhancement procedure that improves on the images obtained by fast algorithms such as the adaptive median filter; or as a pre-processing procedure that cleans up images before dimensionality reduction in data mining [2].

Our computational cost can be reduced further by better implementations of minimization routines for solving (3), see for examples, the continuation method [9] and the primal-dual formulation [8] for TV minimization.

V. Conclusion

In this paper, we propose a decision-based, details preserving restoration method. It is the ultimate filter for removing salt-and-pepper noise. Experimental results show that our method performs much better than median-based filters or the edge-preserving regularization methods. Even at a very high noise level ($\leq 90\%$), the texture, details and edges are preserved accurately. One can further improve our results by using different noise detectors and regularization functionals that are tailored to different types of noises, such as the random-valued impulse noise or impulse-plus-Gaussian noise. These extensions together with fast solvers for (3) will be given in our forthcoming papers.

References

[1] J. Astola and P. Kuosmanen, *Fundamentals of Nonlinear Digital Filtering.* Boca Raton, CRC, 1997.

[2] E. Bingham and H. Mannila, "Random projection in dimensionality reduction: applications to image and text data," *Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-2001)*, August 26-29, 2001, San Francisco, CA, USA, pp. 245–250.

[3] M. Black and A. Rangarajan, "On the unification of line processes, outlier rejection, and robust statistics with applications to early vision," *International Journal of Computer Vision*, 19 (1996), pp. 57–91.

[4] C. Bouman and K. Sauer, "On discontinuity-adaptive smoothness priors in computer vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17 (1995), pp. 576–586.

[5] A. Bovik, *Handbook of Image and Video Processing*, Academic Press, 2000.

[6] R. H. Chan, C.-W Ho, and M. Nikolova, "Convergence of Newton's method for a minimization problem in impulse noise removal", *Journal of Computational Mathematics*, 22 (2004), pp. 168–177.

[7] T. F. Chan and S. Esedoḡlu, "Aspects of total variation regularized L^1 function approximation," Department of Mathematics, UCLA, CAM Report (04-07), 2004.

[8] T. F. Chan, G. H. Golub and P. Mulet, "A nonlinear primal-dual method for total variation-based image restoration," *SIAM Journal on Scientific Computing*, 20 (1999), pp. 1964–1977.

[9] T. F. Chan, H. M. Zhou, and R. H. Chan, "A continuation method for total variation denoising problems," *Proceedings of SPIE Symposium on Advanced Signal Processing: Algorithms, Architectures, and Implementations, ed. F. T. Luk*, 2563 (1995), pp. 314–325,

[10] P. Charbonnier, L. Blanc-Féraud, G. Aubert, and M. Barlaud, "Deterministic edge-preserving regularization in computed imaging," *IEEE Transactions on Image Processing*, 6 (1997), pp. 298–311.

[11] T. Chen and H. R. Wu, "Space variant median filters for the restoration of impulse noise corrupted images," *IEEE Transactions on Circuits and Systems II*, 48 (2001), pp. 784–789.

[12] H.-L. Eng and K.-K. Ma, "Noise adaptive soft-switching median filter," *IEEE Transactions on Image Processing*, 10 (2001), pp. 242–251.

[13] R. C. Gonzalez and R. E. Woods, *Digital Image Processing Second Edition*, Prentice Hall, 2001; and *Book Errata Sheet (July 31, 2003)*, http://www.imageprocessingbook.com/downloads/errata_sheet.htm.

[14] P. J. Green, "Bayesian reconstructions from emission tomography data using a modified EM algorithm," *IEEE Transactions on Medical Imaging*, MI-9 (1990), pp. 84–93.

[15] Y. Hashimoto, Y. Kajikawa and Y. Nomura, "Directional difference-based switching median filters," *Electronics and Communications in Japan*, 85 (2002), pp. 22–32.

[16] T. S. Huang, G. J. Yang, and G. Y. Tang, "Fast two-dimensional median filtering algorithm," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, 1 (1979), pp. 13–18.

[17] H. Hwang and R. A. Haddad, "Adaptive median filters: new algorithms and results," *IEEE Transactions on Image Processing*, 4 (1995), pp. 499–502.

[18] M. Nikolova, "Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers," *SIAM Journal on Numerical Analysis*, 40 (2002), pp. 965–994.

[19] M. Nikolova, "A variational approach to remove outliers and impulse noise," *Journal of Mathematical Imaging and Vision*, 20 (2004), pp. 99–120.

[20] T A. Nodes and N. C. Gallagher, Jr. "The output distribution of median type filters," *IEEE Transactions on Communications*, COM-32, 1984.

[21] G. Pok, J.-C. Liu, and A. S. Nair, "Selective removal of impulse noise based on homogeneity level information," *IEEE Transactions on Image Processing*, 12 (2003), pp. 85–92.

[22] C. R. Vogel and M. E. Oman, "Fast, robust total variation-based reconstruction of noisy, blurred images," *IEEE Transactions on Image Processing*, 7 (1998), pp. 813–824.

[23] Z. Wang and D. Zhang, "Progressive switching median filter for the removal of impulse noise from highly corrupted images," *IEEE Transactions on Circuits and Systems II*, 46 (1999), pp. 78–80.

[24] S. Zhang and M. A. Karim, "A new impulse detector for switching median filters," *IEEE Signal Processing Letters*, 9 (2002), pp. 360–363.