Adaptive Two-pass Rank Order Based Filter to Remove Impulse Noise in Highly Corrupted Images

Xiaoyin Xu*, Eric L. Miller, Dongbin Chen, and Mansoor Sarhadi

Abstract

In this paper, we present an adaptive two-pass rank order based filter to remove impulse noise in highly corrupted images. When noise ratio is high, rank order based filter, median filter for example, usually cannot produce satisfactory results. Better results can be obtained by applying the filter twice, which we call two-pass filter. To further improve the performance, we develop an adaptive two-pass rank order based filter. Between the passes of filtering, an adaptive process is used to detect irregularities in spatial distribution of the estimated impulse noise. The adaptive process then selectively replaces some pixels changed by the first pass of filtering by their original observed pixel values. These pixels are then kept unchanged during the second filtering. In combination, the adaptive process and the second filter eliminate more impulse noise and restore some pixels that are mistakenly altered by the first filtering. As a final result, the reconstructed image maintains a higher degree of fidelity and has a smaller amount of noise. The idea of adaptive two-pass processing can be applied on many rank order based filters, such as center weighted median filter, adaptive center weighted median filter, lower-upper-middle (LUM) filter, and soft-decision rank-order-mean (SD-ROM) filter. Results from computer simulations are used to demonstrate the performance of the algorithm.

Index Terms

Median filter, center weighted median filter, lower-upper-middle filter, SD-ROM filter, impulse noise, error index matrix, spatial distribution of impulse noise.

X. Xu and E.L. Miller are with the Center for Subsurface Sensing and Imaging Systems, Department of Electrical and Computer Engineering, Northeastern University, Boston, MA 02115, USA. D. Chen and M. Sarhadi are with the Department of Systems Engineering, Brunel University, UK. *Corresponding author: X. Xu, email: xxu@ece.neu.edu.

EDICS 2-NFLT

I. Introduction

In image processing, median filter is usually used to remove impulse noise [1]. Compared with convolutional filters, the median filter is more robust such that a single very unrepresentative pixel in the filter window will not affect the median value significantly. On the other hand, since the median must actually be one of the pixels in the filter window, the median filter does not create new pixel values when the filter crosses an edge. For this reason the median filter is better at preserving sharp discontinuities than spatial average filter [2]. One problem of the median filter is that it may alter pixels undisturbed by the noise [2, 3] and causes edge jitter [4, 5] and streaking [6]. Modifications of the median filter have been proposed to overcome these shortcomings. Basically the task is to decide when to apply the median filter and when to keep pixels unchanged [7]. Among those are center weighted median filters [8–11] which gives the current pixel a large weight and the ultimate output is chosen between the median and the current pixel value, detail-preserving median filters [12], and rank-ordered mean filter [13] which excludes the current pixel itself from the median filter, progressive switching median filter (PSM) [14], soft decision based filter [7, 15] and predication-based filter [16]. A different kind of weighted median filter has been proposed by Yin et al. [17] where a nonnegative integer weight is assigned to each position in the filter window. Recently, impulse noise removal based on fuzzy logic has been attracting research effort [18]. In general, these methods can be classified to two categories. The first category is adaptive window methods, which adaptively choose the window used to filter each pixel [4, 19]. The second category is non-adaptive window methods, such as those presented in [20, 21]. All the above mentioned filters belong to rank order based filter.

In this paper, we propose a method that improve the performance of the median filter and other rank order based filter by analyzing the spatial distribution of the estimated impulse noise. The impulse noise is estimated by subtracting the rank order based filter output from the observation. Using an underlying rank order based filter, be it standard median filter, center weighted median filter, or SD-ROM filter, our method looks for irregularities in the spatial distribution of the estimated impulse noise over a subset of the image. The subset can be a column or a row. Once an irregularity is detected, some pixels changed by the first filtering are replaced by their original pixel values and kept unchanged during the second pass of the filter. Irregularities are detected adaptively in the framework of hypothesis test, by scanning the image in a specific direction. Our method aims to achieve two objectives. First, the algorithm uses two-pass rank order based filter to remove more impulse noise when noise ratio is high. Second, by exploiting spatial distribution of the estimated impulse noise the algorithm corrects some errors made by the first filtering. By doing so, improved results are obtained in terms of better visual appreciation and higher peak signal-to-noise ratio (PSNR).

In this paper, we describe our method in detail based on standard median filter. As we will show later on, it is straight-forward to generalize our method to other rank order based filters because the method is independent of the underlying rank order based filter. The paper is organized as follows. Section II discusses impulse noise models and the standard median filter. Then it presents the algorithm based on standard median filter. Section III demonstrates the performance of our algorithm using examples. We also extend the idea of adaptive two-pass filtering to other popular rank order based filters and show that improved results are obtained in each case. Finally, conclusions are given in Section IV.

II. Algorithm

In this section, at first we briefly describe impulse noise model and the standard median filter. Then we introduce an adaptive two-pass median filtering process. And it will be made clear that the key part of our method is independent of the underlying impulse noise detection scheme and therefore our method can be generalized to other rank order based filters in a straight-forward manner.

A. Noise model

At first, we define noise ratio as

$$\text{Noise ratio} = \frac{\text{Number of pixels of impulse noise}}{\text{Total number of pixels in the image}}\ (\%). \tag{1}$$

Impulse noise can be described by their probability distributions in space and amplitude, which are assumed to be independent from each other. Denoting an $M \times N$ matrix $\mathbf{U}$ as the impulse noise, mathematically we can write

$$\mathbf{U} = \mathbf{U}_{POS} \odot \mathbf{U}_{AMP} \tag{2}$$

where matrix $\mathbf{U}_{POS}$, a matrix of size $M \times N$, represents the positions of the impulse noise, $\mathbf{U}_{AMP}$, a matrix of size $M \times N$, represents the amplitudes of the impulse noise at each pixel position, and $\odot$ denotes the point-by-point multiplication. For the noise model, we assume that the impulse noise satisfies a binary distribution at each pixel $\mathbf{U}_{POS}(m,n)$

$$\begin{aligned} \text{Prob}\{\mathbf{U}_{POS}(m,n) = 1\} &= q \\ \text{Prob}\{\mathbf{U}_{POS}(m,n) = 0\} &= 1-q, \quad m = 1,\ldots,M, n = 1,\ldots,N \end{aligned} \tag{3}$$

where $0 \leq q \leq 1$. The binary distribution indicates that at position $[m,n]$, the probability that there is an impulse noise is q and probability is $1-q$ that there is no impulse noise.

Intuitively, q equals the noise ratio. The amplitude of impulse noise may have a Gaussian distribution, a uniform distribution or a fixed-value.

B. Standard median filter and types of error

Consider an image $\mathbf{S}$ and an observation $\mathbf{X}$ of size $M \times N$

$$\mathbf{X} = \mathbf{S} + \mathbf{U} \tag{4}$$

where $\mathbf{U}$ is the impulse noise. In the above model, noise is additive to the signal. In simulation, clipping at pixel value 255 is applied to keep the simulated corrupted pixel value within the range from 0 to 255, for an 8-bit monochrome image. Median filter is applied over a window surrounding the current pixel $\mathbf{X}(m,n)$ such that

$$\begin{aligned} \mathbf{Y}(m,n) &= \mathcal{MF}(\mathbf{X}(m,n), W) \\ &= \text{median}\{\mathbf{X}(m-k, n-l), (k,l) \in W\}, \quad m = 1,\ldots,M, n = 1,\ldots,N \end{aligned} \tag{5}$$

where W is a predetermined window. Usually, W is chosen to be 3×3, 5×5, or 7×7 [2]. In detecting and removing impulse noise, median filter makes error. First type of error occurs when there is noise but the median filter does not detect it, it is called **Type I** error. Type I error is also called a *miss* in other literature [22]. The second type of errors happens when the median filter detects an impulse noise when there is actually no noise, it is called **Type II** error or a *false alarm.* For example, assuming a signal is given by

$$\mathbf{S} = \begin{bmatrix} 1 & 5 & 1 \\ 1 & 5 & 1 \\ 1 & 5 & 1 \end{bmatrix} \tag{6}$$

and there is no noise, at position $[2,2]$, the 3×3 median filter will generate the result

$$\mathbf{Y} = \begin{bmatrix} 1 & 5 & 1 \\ 1 & 1 & 1 \\ 1 & 5 & 1 \end{bmatrix} \tag{7}$$

which is Type II error. Besides the above two types of errors, median filter also makes a third type of error, which we call *over-correcting* and label it **Type III** error. This type of error happens when there is impulse noise of low amplitude and the median filter removes the impulse noise and replaces it with the median value of in the filter window. When the median value is not as close to the true pixel value as the noisy pixel is, an over-correcting error occurs. Using the above signal for example and assuming there is a low amplitude impulse noise such as

$$\mathbf{U} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 0 \end{bmatrix} \tag{8}$$

then the observation will be

$$\mathbf{X} = \begin{bmatrix} 1 & 5 & 1 \\ 1 & 7 & 1 \\ 1 & 5 & 1 \end{bmatrix}. \tag{9}$$

To find the correct value at position $(2, 2)$, the 3×3 median filter will generate a result of

$$\mathbf{Y} = \begin{bmatrix} 1 & 5 & 1 \\ 1 & 1 & 1 \\ 1 & 5 & 1 \end{bmatrix} \tag{10}$$

instead of the original signal. In this case, the output is incorrect, what's more, the output of median filter has a larger error than the original noisy observation. Therefore, it is beneficial to adaptively choose when to keep median filter output and when to keep the original pixel value.

C. Adaptive two-pass median filter to detect impulse noise

To improve the performance of the median filter, we introduce an adaptive two-pass median filter (ATPMF). The idea of adaptive two-pass median filter is to use first median filter to clean up the image and obtain an estimation of the **spatial distribution and**

amplitude of the impulse noise. Using the estimated impulse noise, an adaptive process is carried out to selectively replace some pixels by the original pixel values of $\mathbf{X}$. These pixels are kept unchanged in the second median filtering, which is used to remove the remaining noise.

To facilitate discussion, we introduce two operators. The first operator Ω takes a vector or matrix input and marks nonzero elements of the input by one and assigns zeros to other elements,

$$\Omega(x) = \begin{cases} 1, & \text{if } x \neq 0, \\ 0, & \text{if } x = 0. \end{cases} \tag{11}$$

For example

$$\Omega\left(\begin{bmatrix} 1 & 0 & 3 \\ 0 & 5 & 2 \\ 3 & 0 & 0 \end{bmatrix}\right) = \begin{bmatrix} 1 & 0 & 1 \\ 0 & 1 & 1 \\ 1 & 0 & 0 \end{bmatrix}. \tag{12}$$

The second operator Γ operates on a vector $\mathbf{u} = [u_1, \ldots, u_N]^T$, where T denotes the transpose, and returns the positions of the first k smallest elements of $\mathbf{u}$

$$\mathbf{v}_{k\times 1} = \Gamma(\mathbf{u}, k), \quad k = 1, \ldots, N. \tag{13}$$

For example, if $\mathbf{u} = [2, 1, 4, 3, 6, 5, 7]^T$ then

$$\Gamma(\mathbf{u}, 3) = [2, 1, 4]^T. \tag{14}$$

We now describe the three steps of our algorithm, see Fig. 1. Step 1 is the standard median filtering, (5). Matrix $\mathbf{E}_1$, which we call error index matrix (EIM), records pixel positions of $\mathbf{Y}$ different from $\mathbf{X}$. Pixels at these positions are supposedly contaminated by impulse noise. Step 2 analyzes $\mathbf{Y}$ and $\mathbf{E}_1$ to determine which pixels are most-likely over-corrected by the first median filter. The over-corrected pixels shall be replaced by their original pixel values and kept unchanged in the third step. Step 2 also generates the second error index matrix $\mathbf{E}_2$, it determines which pixels shall stay unchanged in step 3. Step 3 carries out the second pass of filtering wherever $\mathbf{E}_2(m, n) = 0$.

Step 1. For each pixel $\mathbf{X}(m,n)$, applying median filter

$$\mathbf{Y}(m,n) = \mathcal{MF}(\mathbf{X}(m,n), W_1) \quad (15)$$

and generates an error index matrix

$$\mathbf{E}_1 = \Omega(\mathbf{X} - \mathbf{Y}).$$

Step 2. Adaptively replace some original pixels of $\mathbf{Y}$ by original pixels of $\mathbf{X}$

$$\tilde{\mathbf{Y}} = \Psi(\mathbf{Y}, \mathbf{E}_1) \quad (17)$$

$$\mathbf{E}_2 = \Omega(\mathbf{Y} - \tilde{\mathbf{Y}}) \quad (18)$$

where Ψ represents the adaptive processor shown in Fig. 2.

Step 3. For each pixel $\tilde{\mathbf{Y}}(m,n)$

$$\mathbf{Z}(m,n) = \begin{cases} \mathcal{MF}(\tilde{\mathbf{Y}}(m,n), W_2), & \text{if } \mathbf{E}_2(m,n) = 0 \\ \mathbf{X}(m,n), & \text{if } \mathbf{E}_2(m,n) = 1 \end{cases}$$

Fig. 1. Steps of adaptive method.

Details of step 2 is given in Fig. 2. Here the notation $\mathbf{X}(:,n)$ stands for the n-th column of $\mathbf{X}$, the same for $\mathbf{Y}(:,n)$. In step 2-a, predetermined parameter a controls the threshold of detecting a column over-corrected and b controls how many pixels will be replaced by their original values. Step 2-b and 2-c loops over sub-images to remedy over-correction. In step 3, $\mathbf{E}_2$ forces the median filter to avoid pixels that are purposely recovered in step 2. In step 1 and 3, W_1 and W_2 may be different. In Fig. 2 we apply the ATPMF by column. It can be easily modified to proceed by row. In step 2, first, columnwise noise ratio is estimated by

$$\lambda(n) = \frac{\sum_{m=1}^{M} \mathbf{E}_1(m,n)}{M}, \quad n = 1, \ldots, N. \quad (22)$$

Second we can compute the mean μ_λ and standard deviation σ_λ of $\lambda(n)$. For an impulse noise satisfying binary distribution, by De Moivre-Laplace theorem [23], as M goes large, $\lambda(n)$ will approximate a Gaussian distribution $N(\mu_\lambda, \sigma_\lambda)$ where $\mu_\lambda = Mp$ and $\sigma_\lambda = Mp(1-p)$. **It is noted that step 2 is independent of the filter used in step 1. Step 2 is carried out based on the estimated impulse noise.** Therefore, we can replace the standard median

2-a Choose parameter $\eta = a\sigma_\lambda$ where $a > 0$ and $b > 0$

2-b FOR $n = 1 : N$

- $\lambda(n) = \frac{\sum_{m=1}^{M} \mathbf{E}_1(m,n)}{M}$
- IF $(\lambda(n) - \mu_\lambda) > \eta$
  - Let
$$\mathbf{e} = \mathbf{X}(:,n) - \mathbf{Y}(:,n) \tag{20}$$
  - Let $K = \text{round}(\lambda(n) - \mu_\lambda + b\sigma_\lambda)$
  - $\mathbf{v} = \Gamma(\mathbf{e}, K)$
  -
$$\tilde{\mathbf{Y}}(m,n) = \begin{cases} \mathbf{X}(m,n) & \text{for } m \in \mathbf{v}, m = 1, \ldots, M \\ \mathbf{Y}(m,n) & \text{otherwise.} \end{cases}$$
- ENDIF

2-c ENDFOR

Fig. 2. Adaptive processor Ψ to detect impulse noise.

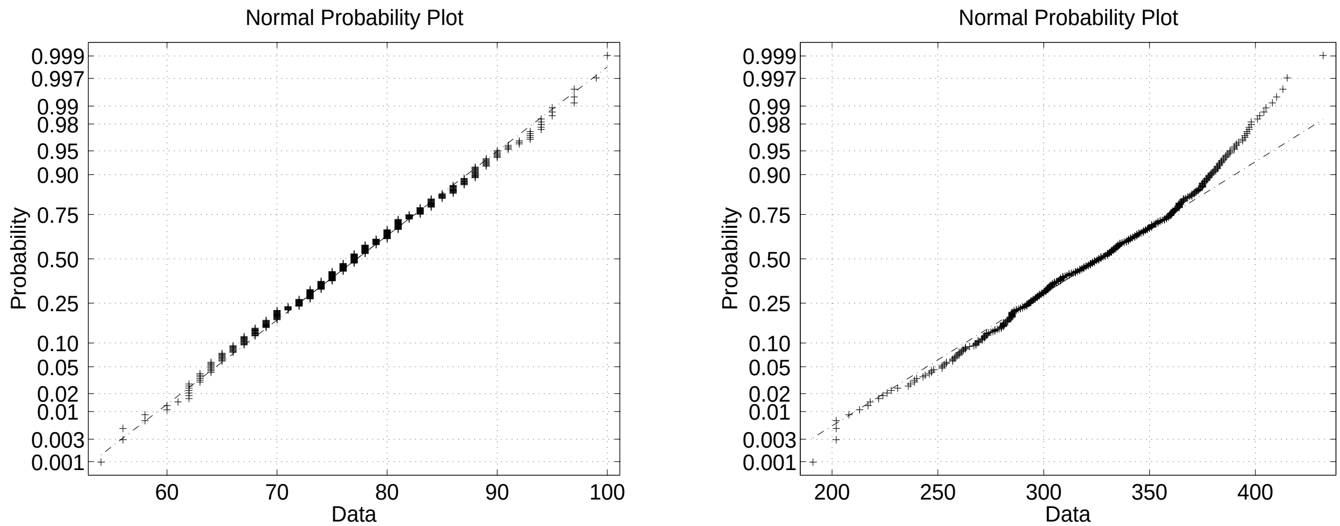
filter in step 1 by other rank order based filter, as we will show in next section.

Fig. 3(a)–(c) show a true image, a noisy observation $\mathbf{X}$, and the restored image by the 3×3 median filter. The impulse noise has a Gaussian distribution in amplitude and a noise ratio of 25%. Fig. 3(d) shows the result obtained by ATPMF. It is seen that the ATPMF produces a cleaner image than the standard median filter.

Fig. 4(a) plots $\lambda(n)$ using the normplot command of the MATLAB, assuming the impulse noise is known *a priori*. It is seen that the curve matches a Gaussian distribution very well. Fig. 4(b) plots $\lambda(n)$ of the example shown in Fig. 3. It is seen that the estimated number of noisy pixels per column is close to a Gaussian distribution. Therefore when $\lambda(n)$ differs too much from μ_λ, it is believed that some pixels of column n are over-corrected by the first median filter and should be replaced by the original pixels. Using μ_λ and σ_λ, at each column we make a one-side parameter test

$$\begin{aligned} \mathcal{H}_0 &: \lambda(n) \sim N(\mu_\lambda, \sigma_\lambda^2) \\ \mathcal{H}_1 &: \lambda(n) \sim N(\mu_\lambda + A, \sigma_\lambda^2) \end{aligned} \tag{23}$$

Fig. 3. Standard median filter to remove impulse noise, (a) original image, (b) noisy observation, (c) restored image by the 3×3 median filter, (d) restored image by the ATPMF. Noise ratio is 25%. Impulse noise has a Gaussian distribution with mean 30 and standard deviation of 5.



(a) (b)

Fig. 4. (a) True columnwise distribution of noisy pixels, (b) estimated columnwise distribution of noisy pixels by the 3×3 median filter. A straight line of '+' represents a Gaussian distribution.

or equivalently (after subtracting μ_λ from the observation)

$$\begin{aligned} \mathcal{H}_0 &: \lambda(n) \sim N(0, \sigma_\lambda^2) \\ \mathcal{H}_1 &: \lambda(n) \sim N(A, \sigma_\lambda^2) \end{aligned} \tag{24}$$

where $A > 0$ and is unknown, to detect columns containing unlikely large number of impulse noise. The null hypothesis indicates that the number of pixels containing impulse noise at n-th column is reasonably small and most of those pixels altered by the median filter truly contains impulse noise. The alternative hypothesis represents the case that there is a excessively large number of pixels altered by the median filter and therefore it is likely some of these pixels do not contain impulse noise and are mistakenly changed by median filtering. Although A is unknown, knowing that $A > 0$ allows us to find a uniformly most powerful (UMP) test [24] using the Neyman-Pearson criterion. The Neyman-Pearson test is to decide $\mathcal{H}_1$ if [22]

$$\frac{p(\lambda;\mathcal{H}_1)}{p(\lambda;\mathcal{H}_0)} = \frac{\frac{1}{\sqrt{2\pi}\sigma_\lambda}\exp\left[-\frac{(\lambda-A)^2}{2\sigma_\lambda^2}\right]}{\frac{1}{\sqrt{2\pi}\sigma_\lambda}\exp\left[-\frac{\lambda^2}{2\sigma_\lambda^2}\right]} > \eta \tag{25}$$

where η is a preset threshold. Taking the logarithm of and simplifying (25) we have

$$A\lambda > \sigma_\lambda^2 \ln\eta + \frac{A^2}{2}. \tag{26}$$

Since $A > 0$, we then have

$$\lambda > \frac{\sigma_\lambda^2}{A}\ln\eta + \frac{A}{2} \equiv \eta'. \tag{27}$$

The probability of false-alarm is

$$P_{FA} = Q\left(\frac{\eta'}{\sigma_\lambda}\right) \tag{28}$$

where η' can be decided by pre-selecting the P_{FA} and Q is the complementary cumulative distribution function of a standard Gaussian probability density function. Then solving the following equation gives η'

$$\eta' = \sigma_\lambda Q^{-1}(P_{FA}) \tag{29}$$

which is independent of A. After finding η' from (29), the probability of detection is given by

$$P_D = Q\left(\frac{\eta' - A}{\sigma_\lambda}\right). \tag{30}$$

The above test is UMP in the sense that for a fixed P_{FA}, it yields the highest P_D among all the tests [24]. As expected, increasing η (or η') reduces both P_{FA} and P_D, and vice versa. In Section III we investigate the effect of different η on the performance of the ATPMF.

D. Choice of a and b in step-2

In step-2, a and b can be chosen based on how tight the Gaussian distribution in columnwise noise ratio shall be. In other words, we can choose how much deviation is acceptable between the $\lambda(n)$ and μ_λ before we declare there is over-correction in column n. For example, we can choose $a = 1$ so that if a columnwise noise ratio deviates from the mean value by one standard deviation, we will consider that column being over-corrected by the first filtering. Similarly we can set $b = 1$ to correct that many pixels in the column.

E. Different implementations

In the above algorithm, the adaptive process is carried out column by column. On the other hand, the adaptive process can be implemented by rows. Fig. 5 shows the restored images of "Lena" by the 5×5 median filter and by an ATPMF implemented in columns and in rows. For the adaptive two-pass filter, we set $a = 1$ and $b = 1$. Comparing the two images of ATPMF, we see there is little difference between them, yet they both are much better than the standard median filter output.

F. Computational issue

Because of the increased steps of computation, the computational time of applying adaptive two-pass filtering is longer than one-pass filtering. The most time-consuming part of our method is to run the underlying rank order based filter. The adaptive part of our method, i.e., step 2 in Fig. 1, takes much less time to complete.

III. Examples

In this section we use some examples to demonstrate the performance of our algorithm and generalize the idea of our method to other rank order based filters such as center weighted median filter, adaptive weighted median filter, lower-upper-middle (LUM) filter, and SD-ROM filter. Quantitatively, we use PSNR to compare the restored images with and without using the adaptive process. For a final output image $\mathbf{Z}$, PSNR is defined as

$$\text{PSNR} = 10 \log_{10} \sum_{m=1}^{M} \sum_{n=1}^{N} \frac{255^2}{(\mathbf{Z}(m,n) - \mathbf{S}(m,n))^2} \quad \text{(dB)}.$$

(a) (b)

(c) (d)

Fig. 5. Different implementation, (a) noisy observation of “Lena”, noise ratio is 35%, (b) restored image by a standard 5×5 filter, PSNR = 20.8517 (dB), (c) restored image by ATPMF implemented in columns, PSNR = 22.8672 (dB), (d) restored image by ATPMF implemented in rows, PSNR = 22.7692 (dB).

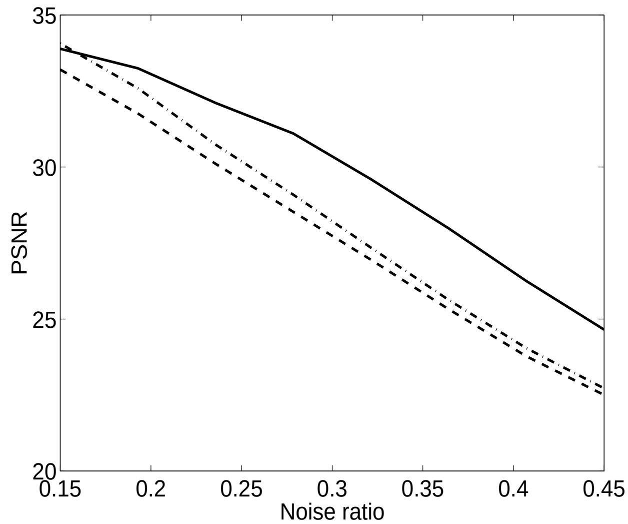


Fig. 6. PSNR of "Lena". Dash line, 3×3 median filter; dash-dot line, two-pass 3×3 median filter; solid line, ATPMF. For the ATPMF, the underlying median filter window is 3×3 and $a = 1$, $b = 1$.

A. Detect impulse noise based on median filter

Fig. 6 compares PSNR of the standard median filter, the two-pass median filter and the ATPMF to process image "Lena". It is seen that at high noise ratios, the ATPMF results have much higher PSNR than those of the standard median filter. It is interesting to observe that by simple two-pass median filtering, results become modestly better. By introducing the adaptive process between two median filters, much improved PSNR are obtained at all noise ratios, especially at high noise ratio, above 30%. In the ATPMF a and b are set to 1. For the second example, we use the "boat" image. Fig. 7 shows the original image, a noisy observation at ratio noise of 30%, restored images by the 3×3 median filter and the ATPMF. Fig. 8 shows the result of processing the "boat" image contaminated by impulse noise with a uniform distribution in amplitude.

(a)

(b)

(c)

(d)

Fig. 7. Median filter to remove impulse noise, (a) original image, (b) noisy observation, noise ratio is 30%, (c) restored image by the 3×3 median filter, PSNR = 24.3833 (dB), (d) restored image by the ATPMF, PSNR = 25.8376 (dB). Adaptive process is implemented in columns with $a = 1$, $b = 1$.

(a) (b)

Fig. 8. Restored "boat" image by, (a) the 3×3 median filter, PSNR = 25.0612 (dB), (b) an ATPMF implemented in rows, $a = 1$, $b = 1$, PSNR = 27.7221 (dB). Noise ratio is 20%. Noise has a uniform distribution between 0 and 80.

B. Effect of η on the ATPMF performance

In the above section we showed that by modeling the detection problem as a one side parameter hypothesis testing, we can find a UMP test based on Neyman-Pearson criterion. Here we investigate the effect of η on the performance of the ATPMF. Fig. 9 plots the PSNR of the "Lena" image restored by a two pass median filter and the ATPMF for different η. As η increases, PSNR increases at first and then decreases.

C. Generalization to other rank order based filters

In recent years, many algorithms have been proposed to reduce impulse noise. Some of them are center weighted median filter [10,11,17], adaptive center weighted median filter [13], lower-upper-middle filter [25], and soft-decision rank-order-mean (SD-ROM) filter [15]. In this section, we apply our algorithm to these filters and show that improved results are

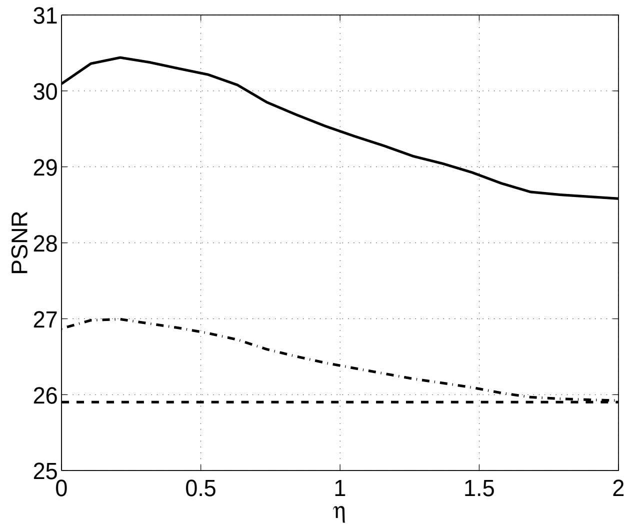


Fig. 9. PSNR of "Lena" for different η, dash line, standard median filter; dash-dot line, two-pass median filter; solid line, ATPMF. Note that standard median filter is not affect by the choice of η, therefore its plot is a straight-line. Noise ratio is 25% and noise has a Gaussian distribution.

obtained in each case, respectively.

Table I shows the results of removing impulse noise with a random amplitude satisfying a Gaussian distribution. In the table, the second row shows the PSNR of using standard 3×3 median filter (MF), center weighted median filter (CWMF) with a filter window of 3×3 and a center weight of 5, adaptive center weighted median filter (ACWMF), LUM filter of $(5, 5, 13)$ [25], and SD-ROM. The third row shows the result by applying our algorithm based on the corresponding filter. For all the examples, we set $a = 1$ and $b = 1$ in the adaptive process. It is seen that our algorithm improves the result of each filter, respectively. As expected, the better the performance of the underlying rank order based filter produces, the better result the adaptive two-pass filter produces.

Table II shows the results of processing an impulse noise with a uniform distribution between 0 and 120. Table III shows the results of processing an impulse noise with fixed-value of 100. From both tables we can see that our algorithm improves the performance of the underlying rank order based filters.

TABLE I
IMAGE OF "LENA", NOISE RATIO IS 30%, GAUSSIAN DISTRIBUTION IN AMPLITUDE.

	MF	CWMF	ACWMF	LUM	SD-ROM
PSNR (dB) (regular)	20.9160	22.8093	23.2441	20.9670	26.2251
PSNR (dB) (adaptive)	24.6247	25.7156	26.4832	25.2988	28.5644

TABLE II
IMAGE OF "LENA", NOISE RATIO IS 30%, UNIFORM DISTRIBUTION IN AMPLITUDE BETWEEN 0 AND 120.

	MF	CWMF	ACWMF	LUM	SD-ROM
PSNR (dB) (regular)	26.5685	28.0796	28.6759	26.9210	27.7217
PSNR (dB) (adaptive)	29.6517	29.9542	30.8357	30.2643	31.4640

D. Processing salt and pepper noise

Rank order based filters are very effective in removing salt and pepper noise. In our algorithm, the adaptive part can be easily modified to remove salt and pepper noise by change (20) in step-2 to

$$\mathbf{e} = |\mathbf{X}(:, n) - \mathbf{Y}(:, n)|. \tag{32}$$

Table IV shows the result of removing salt and pepper noise from the image of "Lena" using different rank order based filters. Comparing the results of before and after using adaptive process, we can see that there is a large improvement in the final results.

IV. CONCLUSION

We presented an adaptive rank order based filtering process to remove impulse noise in highly corrupted images. The adaptive filter is based on underlying filter such as standard median filter, center weighted median filter, adaptive weighted median filter, LUM filter,

TABLE III
IMAGE OF "LENA", NOISE RATIO IS 30%, IMPULSE NOISE WITH FIXED VALUE OF 100.

	MF	CWMF	ACWMF	LUM	SD-ROM
PSNR (dB) (regular)	21.5631	24.3777	23.0800	22.2382	29.0991
PSNR (dB) (adaptive)	26.2769	27.7248	27.2482	26.3723	31.6330

TABLE IV
IMAGE OF "LENA", NOISE RATIO IS 30%, SALT & PEPPER NOISE, GAUSSIAN DISTRIBUTION IN AMPLITUDE.

	MF	CWMF	AWMF	LUM	SD-ROM
PSNR (dB) (regular)	36.2007	39.5893	39.8517	37.5196	37.5220
PSNR (dB) (adaptive)	41.2992	41.9990	43.3003	41.7177	42.1703

and SD-ROM filter. The adaptive process detects irregularities in the spatial distribution of the estimated impulse noise. By analyzing the first error index matrix, the detection is implemented in the framework of hypothesis testing. The method is able to correct some false-alarms caused by the first filtering and remove remaining noise. We have shown that the adaptive filter performs better than using the underlying filter alone in removing impulse noise and reducing false-alarms. Using test images and comparing PSNR of the adaptive filter with those of the underlying filter, we demonstrated the improved performance of our method, especially at high noise ratios, on simulated images.

References

[1] I. Pitas and A.N. Venetsanopoulos, *Nonlinear Digital Filters: Principles and Applications*, Norwell, MA: Kluwer Academic Publishers, 1990.

[2] A.K. Jain, *Fundamentals of Digital Image Processing*, Englewood Cliffs, NJ:Prentice-Hall, 1989.

[3] G.R. Arce and R.E. Foster, "Detail-preserving ranked-order based filters for image processing," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 1, pp. 83–98, Jan. 1989.

[4] A.C. Bovik, T.S. Huang, and D.C. Munson, "Edge-sensitive image restoration using order-constrained least squares methods," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 33, no. 4, pp. 1253–1263, Oct. 1985.

[5] T.A. Nodes and N.C. Gallagher, Jr., "The output distribution of median type filters," *IEEE Trans. Comm.*, vol. COM-32, pp. 532–541, May 1984.

[6] A.C. Bovik, "Streaking in median filtered images," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 35, no. 4, pp. 493–503, Oct. 1985.

[7] D.A. F. Florencio and R.W. Shafer, "Decision-based median filter using local signal statistics," in *Proc. SPIE Symp. Visual Comm. Image Proc.*, Sept. 1994, vol. 2308, pp. 268–275.

[8] T. Chen and H. Wu, "Adaptive impulse detection using center-weighted median filters," *Signal Proc Lett.*, vol. 8, no. 1, pp. 1–3, Jan. 2001.

[9] R. Yang, L. Lin, M. Gabbouj, J. Astola, and Y. Neuvo, "Optimal weighted median filters under structural constraints," *IEEE Trans. Signal Proc.*, vol. 43, no. 3, pp. 591–604, Mar. 1995.

[10] T. Song, M. Gabbouj, and Y. Neuvo, "Center weighted median filters: some properties

and applications in image processing," *Signal Processing*, vol. 35, no. 3, pp. 213–229, 1994.

[11] S.-J. Ko and Y.-H. Lee, "Center weighted median filters and their applications to image enhancement," *IEEE Trans. Circuits Syst.*, vol. 38, no. 9, pp. 984–993, Sept. 1991.

[12] T. Sun and Y. Neuvo, "Detail-preserving median based filters in image processing," *Pattern Recogni. Lett.*, vol. 15, no. 4, pp. 341–347, Apr. 1994.

[13] E. Abreu, M. Lightstone, S.K. Mitra, and K. Arakawa, "A new efficient approach for the removal of impulse noise from highly corrupted images," *IEEE Trans. Image Processing*, vol. 5, no. 6, pp. 1012–1025, June 1996.

[14] Z. Wang and D. Zhang, "Progressive switching median filter for removal of impulsive noise from highly corrupted image," *IEEE Trans. Circuits and Systems*, vol. 46, no. 1, pp. 78–80, Jan. 1999.

[15] H.-L. Eng and K.-K. Ma, "Noise adaptive soft-switching median filter," *IEEE Trans. Image Processing*, vol. 10, no. 2, pp. 242–251, Feb. 2001.

[16] E.S. Hore, B. Qiu, and H.R. Wu, "Prediction based image restoration using a multiple window configuration," *Optical Engineering*, vol. 41(8), pp. 1855–1865, Aug. 2002.

[17] L. Yin, R. Yang, M. Gabbouj, and Y. Neuvo, "Weighted median filters: a tutorial," *IEEE Trans. Circuits Sys.*, vol. 43, no. 3, pp. 157–192, Mar. 1996.

[18] H. G. Senel, II R.A. Peters, and B.Dawant, "Topological median filters," *IEEE Trans. Image Processing*, vol. 11, no. 2, pp. 89–104, Feb. 2002.

[19] Y.H. Lee and S.A. Kassam, "Generalized median filtering and related nonlinear filtering techniques," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 33, no. 3, pp. 672–683, June 1985.

[20] G.R. Arce and M.P. McLoughlin, "Theoretical analysis of max/median filters," *IEEE*

Trans. Acoust., Speech, Signal Processing, vol. 35, no. 1, pp. 60–69, Jan. 1987.

[21] I. Pitas and A.N. Venetsanopoulos, “Nonlinear mean filters in image processing,” *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 34, no. 3, pp. 573–584, June 1986.

[22] C.W. Helstrom, *Elements of Signal Detection and Estimation*, New York, NY:Prentice-Hall, 1995.

[23] S. C. Port, *Theoretical Probability for Applications*, New York, NY:John Wiley & Sons, 1994.

[24] S.M. Kay, *Fundamentals of Statistical Signal Processing: Detection Theory*, Englewood Cliffs, NJ:Prentice-Hall, 1993.

[25] R.C. Hardi and C.G. Boncelet, “LUM filters: a class of rank-order-based filters for smoothing and sharpening,” *IEEE Trans. Signal Processing*, vol. 41, no. 3, pp. 1061–1076, Mar. 1993.