REMOVING RANDOM-VALUED IMPULSE NOISE WITH RELIABLE WEIGHT

QIYU JIN
School of mathematical science, Inner Mongolia University
No.235 Daxuexilu Road, 010021 Hohhot, Inner Mongolia, China

LI BAI
School of Computer Science, University of Nottingham
Nottingham NG8 1BB, UK

ION GRAMA
UMR 6205, Laboratoire de Mathématiques de Bretagne Atlantique
Université de Bretagne-Sud
Campus de Touhannic, BP 573, 56017 Vannes, France

QUANSHENG LIU*
UMR 6205, Laboratoire de Mathématiques de Bretagne Atlantique
Université de Bretagne-Sud
Campus de Touhannic, BP 573, 56017 Vannes, France
School of Computer and Software
Nanjing University of Information Science and Technology
Nanjing 210044, China

JIE YANG
Institute of Image Processing and Pattern Recognition, Shanghai Jiao Tong University
No. 800 Dongchuan Road, Minhang District, Shanghai 200240, China

(Communicated by Jean-Michel Morel)

ABSTRACT. In this paper, we present a patch based weighted means filter for removing an impulse noise by adapting the fundamental idea of the non-local means filter to the random-valued impulse noise. Our approach is to give a weight to a pixel in order to evaluate the probability that the pixel is contaminated by the impulse noise, which we call Reliable Weight of the pixel. With the help of the Reliable Weights we introduce the similarity function to measure the similarity among patches of the image contaminated by a random impulse noise. It turns out that the similarity function has significant anti impulse noise interference ability. We then incorporate the Reliable Weights and the similarity function into a filter designed to remove the random impulse noise. Under suitable conditions, we establish two convergence theorems to demonstrate that our method is feasible. Simulation results confirm that our filter is competitive compared to recently proposed methods.

2010 Mathematics Subject Classification. Primary: 62H35.

Key words and phrases. Random-valued impulse noise, Reliable Weight, Denoising, Optimal Weights Filter, Non-Local Means, Rank Ordered Absolute Difference.

The first author is supported by the National Natural Science Foundation of China (Grants No. 61661039, No.61661040, No. 61661038, No. 11401590, No. 11571052 and No. 11731012), China Scholarship Council (Grant No. 201806810001) and Hunan Provincial Natural Science Foundation of China (Grant No.2017JJ2271). The work has benefited from the support of the Centre Henri Lebesgue (CHL, ANR-11-LABX-0020-01).

* Corresponding author: Quansheng Liu.

171
1. **Introduction.** Random-valued impulse noise can be systematically introduced into digital images during acquisition and transmission [19]. Impulse noise is characterized by replacing a portion of an image’s pixel values with random values, leaving the remainder unchanged. In most applications, denoising is fundamental to subsequent image processing operations, such as edge detection, image segmentation, object recognition, etc. The goal of denoising is to effectively remove noise from a noisy image while keeping its features intact. To this end, a variety of techniques have been proposed.

The Median Filter and its extensions [37, 8, 27, 31, 41] can suppress noise with high computational efficiency. However, since such filters are applied to the entire image without prior identification of the corrupted pixels, they tend to modify pixels that are not affected by noise. Thus Median Filter and its extensions often remove desirable image details and blur the image.

In order to better remove impulse noise, we must consider the special properties of the noise. An important characteristic of this type of noise is that only part of the pixels are corrupted and the rest are noise free. This characteristic suggests that the noisy pixels should be detected first and filtered afterward, i.e., replaced by some estimated values, whereas the undisturbed pixels should be left unchanged. By using this approach, a whole new class of impulse noise filters has been built, for example, Switching Median (SM) Filter [41], Multistate Median (MSM) Filter [13], Adaptive Center Weighted Median (ACWM) Filter [12], the Peak-and-Valley Filter [47, 4], Signal-Dependent Rank-Order Mean (SD-ROM) Filter [1], Conditional Signal-Adaptive Median (CSAM) Filter [35], the Pixel-Wise Median of the Absolute Deviations (PWMAD) Filter [14], Modified Threshold Boolean Filter (MTBF) [2], Jarque-Bera Test Based Median (JM) Filter [6], Two-Output Nonlinear Filter [39], Iterative Median Filter [46], etc. The main drawback of these filters is that they just use median values or their variations to restore the noisy pixels, and hence they usually cannot preserve the image details even when the images are only mildly corrupted, say with noise ratio less than 30%.

Recently, many edge-preserving regularization methods have been proposed to remove impulse noise [34, 53, 36, 45, 54, 51, 30, 5, 48, 3, 32, 52, 23, 29]. For example, Nikolova [34] used a non-smooth data-fitting term along with edge-preserving regularization. In order to improve this variational method in removing impulse noise, a two-stage method was proposed in [9] and [10]. This method is efficient in dealing with high noise ratio, e.g., ratio as high as 90% for salt-and-pepper impulse noise and 50% for random-valued impulse noise, but its performance is impaired by the inaccuracy of the noise detector in the first phase. In order to find a better noise detector, especially for the random-valued impulse noise, Garnett et al. [18] introduced a local image statistic called ROAD (Rank-Ordered Absolute Difference) to identify the impulse noisy pixels and incorporated it into a filter designed to remove additive Gaussian noise. They thus proposed a TriF capable of removing mixed Gaussian and impulse noise. This method also performs well for removing impulse noise. However, when the noise level is high, it blurs the images significantly. Delon and Desolneux [15, 16], Li et al. [28] and Hu et al. [21] introduced patch-based approaches to deal with impulse noise and a mixture of Gaussian noise and impulse noise, respectively. Dong et al. [17] amplified the differences between noisy pixels and noise-free pixels in ROAD, thus introduced a modified ROAD statistic called ROLD (Rank-Ordered Logarithmic Difference). Xiong and Yin [50] proposed a detection mechanism for impulse noise and then constructed a filter which efficiently
removes impulse and Gaussian mixed noise. Lien et al. [30] employed a decision-tree-based impulse noise detector and an edge-preserving filter to reconstruct the intensity values of noisy pixels, whose hardware cost is low.

In the two-step approaches (first detecting the impulse noisy points then filtering the noise), one problem is that the distinction between noise free pixels and impulsive ones cannot always be detected, and some noise free pixels may be detected as noisy ones. In this paper we will develop a one step approach. In a conference communication [24] we introduced a new filter called Reliable Radiometric Weight Filter (RRWF), based on the weight optimization. The objective of this paper is to develop the theoretical aspects of this filter.

In our approach we do not try to classify pixels to noise-free pixels and impulsive ones by detection; instead we will introduce, for each pixel, what we call Reliable Weight which measures the probability that the pixel is noise-free; the Reliable Weight of a pixel reflects the reliability level for the pixel to be noisy free. To define the Reliable Weights, we first introduce a standardized version of the ROAD statistic, called SROAD, to obtain a more stable statistic for the detection of impulse noisy pixels; we then get Reliable Weight by using the Gaussian function and a truncation of the SROAD statistic. By means of the Reliable Weights, a similarity function between two patches is introduced to measure the similarity among patches of an image contaminated by an impulse noise, which has significant anti impulse noise interference ability. Finally, we incorporate the Reliable Weights and the similarity function into a filter designed to remove random-valued impulse noise. Extensive experimental results show that our method performs significantly better than many known techniques.

Like the filters introduced in Li et al. [28] and Hu et al. [21], the filter RRWF is a one-step patch-based approach in which we do not judge whether a pixel is noise-free, but put very small weights in the weighted means for pixels which are likely impulse noisy. However, compared with [21] (which is an improvement on [28]), there are two major differences in the present work: (1) The use of the SROAD statistic rather than the ROAD statistic makes the choice of the parameter $H$ much more stable (it is not sensible to the choice of the size $R$ of the windows used for the calculation of the SROAD statistic); accordingly the choice of the parameter $H$ is less sensitive to the value of $p$ (since the choice of $R$ depends the value of $p$). (2) In the weighted means here we use the optimal weights obtained in [25] instead of the usual Gaussian weights. Simulation results show that the performance of the filters are very similar for small values of $p$ ($\leq 40\%$), but the new filter performs significantly better for large value of $p$ ($p > 40\%$), especially for $p \geq 60\%$. The main point is that the ROAD statistic was used in [21], and the parameters proposed therein are no longer suitable for $p \geq 60\%$. Recall that the TriF introduced in [18] is also a one-step approach, but not patch-based; the patch-based filter proposed in [28] is an improvement of the TriF. Compared with the filters proposed in [18] and [28], in addition to the two aspects mentioned above, RRWF has one further advantage: the weights $w_H(x)$ and $w_H(y)$ (see (17) and (25)) used here are one-dimensional indexed ($x \in I$) rather than two-dimensional indexed ($(x_0, x) \in I^2$), in the sense that here we use the same weights $w_H(x)$ and $w_H,y$ for each research window centered at $x_0 \in I$, while in [18] and [28] we need to calculate the new weights $w_H(x_0, x)$ depending on both $x$ and $x_0$, when the center $x_0$ of the research window varies. This progress reduces very significantly the calculation time.
Two convergence theorems will be established in this paper, which not only justify the convergence of the filter RRWF to the original image $u$, but also give hints on the choice of parameters. More precisely we will first introduce an oracle filter by means of the similarity distance $|u(x) - u(x_0)|$ and prove its convergence, see Theorem 3.3. We then construct an estimate of the similarity distance $|u(x) - u(x_0)|$, which is proved to be convergent, see Theorem 3.2.

The outline of this paper is as follows. In Section 2, we give a brief review of the Optimal Weights Filter [25] and the TriF with ROAD statistic [18]. In Section 3, we introduce Reliable Weights, construct our new filter and establish its convergence. The computational algorithm, the choice of parameters and the simulation results to demonstrate the performance of the new filter are presented in Section 4. Section 5 is devoted to the proofs of the convergence theorems. Conclusions are drawn in Section 6.

2. Preliminaries.

2.1. Impulse noise model. An image containing random-valued impulse noise can be described as follows:

$$v(x) = \begin{cases} u(x), & \text{with probability } 1 - p; \\ n(x), & \text{with probability } p, \end{cases}$$

where $u(x), x \in I = \{0, \frac{1}{N}, \cdots, \frac{N-1}{N}\}^2$, is the original image, $n(x), x \in I$, are independent random variables uniformly distributed on $G = \{255 \times \frac{i}{G} | i = 0, 1, \cdots, G\}$, $G$ is a positive integer, and $p$ denotes the proportion of noisy pixels. In order to facilitate the theoretical explanation, we assume that the original image is defined continuously in the unit square $I_0 = [0, 1] \times [0, 1]$, but observed on the set $I$ of $N \times N$ pixels. The goal is to recover the original image $u(x_0)$, for any $x_0 \in I_0$, from the observed image $v(x), x \in I$.

There are two forms of impulse noise: salt and pepper noise ($G = 1$) and random impulse noise $G = 255$. The salt-and-pepper noise is simpler than random impulse noise $G = 255$. It is easy to find the noised pixels for salt-and-pepper noisy image because of that $n(x)$ takes only two values 0 and 255, but it is very difficult to estimate noisy pixels for random impulse noisy image. Therefore there are excellent noise detectors for detecting salt-and-pepper noise and for removing it even when the noise ratio is higher than 90% (see e.g. [9, 40, 43, 22, 42, 33, 20]). In this paper, we focus only on random-valued impulse noise with $G = 255$. The dynamic range of the images is $G = \{0, 1, \cdots, 255\}$, and the noise $n(x)$ is uniformly distributed on $G$.

2.2. Review on the optimal weights filter. In this section we briefly review the Optimal Weights Filter in order to adapt it for removing the impulse noise. Based on similarities among local patches, the Optimal Weights Filter [25] was initially introduced to deal with the Gaussian noise model,

$$v(x) = u(x) + \varepsilon(x), \quad x \in I,$$

where $u$ is the original image defined on the unit square $I_0$, $v$ the observed one defined on the lattice $I$, $\varepsilon$ is the Gaussian noise: for $x \in I$, $\varepsilon(x)$ are independent Gaussian variables with mean 0 and standard deviation $\sigma > 0$. For each point $x = (x_1, x_2) \in I_0$, let $x' = (x'_1, x'_2) = \left(\frac{[N x_1]}{N}, \frac{[N x_2]}{N}\right) \in I$ where $[t]$ denotes the
integral part of $t$, i.e. $x'$ is the closest point in $I$ of $x$, which lies in the left and lower side of $x$. For any point $x \in I_0$ and a positive odd integer $d$, denote by
\begin{equation}
N_{x,d} = \left\{ y \in I : \|y-x'\|_{\infty} \leq \frac{d - 1}{2N} \right\}
\end{equation}
the square window with center $x'$ containing $d \times d$ pixels of $I$, where $\| \cdot \|_{\infty}$ denotes the supremum norm: $\|z\|_{\infty} = \max \{|z_1|, |z_2|\}$ for $z = (z_1, z_2)$, with $d-1$ represents half of the edge size of the window. For simplicity, with abuse of language, we also say that $N_{x,d}$ is a window with centre $x$. Notice that by our notation, we have, for $x \in I_0$ and $t \in I$,
\begin{align*}
N_{x,t} = N_{x',d} \quad \text{and} \quad (x+t)' = x' + t.
\end{align*}

Windows of different sizes will be used. In the following, for $x_0, x \in I_0$ and positive odd integers $d, D, R$, we will use $N_{x,d}$ for similarity patches, $N_{x_0,D}$ for search windows, and $N_{x,R}$ for the detection window of the ROAD statistic and SROAD statistic which will be introduced later. For convenience, we extend the definition of the observed image $v$ to the whole unit square $I_0$ by setting $v(x) = v(x')$ for each $x \in I_0$. For each $x \in I_0$, the vector
\begin{equation}
v(N_{x,d}) = (v(y))_{y \in N_{x,d}}
\end{equation}
formed by the values $v(y)$ of the observed noisy image at pixels $y \in N_{x,d}$, arranged in the lexicographical order, will be called data patch or similarity patch centered at $x$. For any $x_0, x \in I_0$, define
\begin{equation}
\|v(N_{x_0,d}) - v(N_{x,d})\|_2^2 = \frac{1}{d^2} \sum_{t \in N_{x_0,0,d}} (v(x_0 + t) - v(x' + t))^2,
\end{equation}
which measures the similarity between the data patches $v(N_{x_0,d})$ and $v(N_{x,d})$.

The Optimal Weights Filter [25] is defined by
\begin{equation}
\text{OWF}(u)(x_0) = \sum_{x \in N_{x_0,D}} w(x_0, x) v(x), \quad \forall x_0 \in I_0,
\end{equation}
where
\begin{equation}
w(x_0, x) = \frac{\kappa_{tr} \left( \hat{p}_{x_0}(x) \right)}{\sum_{y \in N_{x_0,D}} \kappa_{tr} \left( \hat{p}_{x_0}(y) \right)},
\end{equation}
\begin{align*}
\hat{p}_{x_0}(x) &= \left( \|v(N_{x_0,d}) - v(N_{x,d})\|_2^2 - \sqrt{2}\sigma \right)_+^s,
\end{align*}
where $\hat{a} > 0$ is a number depending on $\{\hat{p}_{x_0}(x) \mid x \in N_{x_0,D}\}$ and $\sigma^2$, whose value can be calculated by the algorithm presented in Remark 1 below, and $\kappa_{tr}$ is the usual triangular kernel:
\begin{equation}
\kappa_{tr} (s) = (1 - |s|)_+, \quad s \in \mathbb{R},
\end{equation}
with $(\cdot)_+$ denoting the positive part function: $(a)_+ = \max\{a, 0\}$. The Optimal Weights Filter is constructed by minimizing a tight bound of the quadratic risk. It is shown in [25] that the optimal weights are given by the formula (6) via the triangular kernel (7). This minimization procedure gives also an exact formula for the bandwidth $\hat{a}$ as stated below.
Remark 1. According to Remark 1 of [25], the bandwidth \( \hat{a} > 0 \) is the solution of

\[
\sum_{x \in \mathcal{N}_{x_0,D}} \frac{1}{\sigma^2} \hat{\rho}_{x_0}(x)(\hat{a} - \hat{\rho}_{x_0}(x))_+ = 1,
\]

and can be calculated as follows. Sort the set \( \{ \hat{\rho}_{x_0}(x) \mid x \in \mathcal{N}_{x_0,D} \} \) in the ascending order \( 0 = \hat{\rho}_1 \leq \hat{\rho}_2 \leq \cdots \leq \hat{\rho}_M < \hat{\rho}_{M+1} = +\infty \), where \( M = \text{card} \mathcal{N}_{x_0,D} \). Let

\[
\hat{a}_k = \frac{\sigma^2 + \sum_{i=1}^{k} \hat{\rho}_i}{\sum_{i=1}^{k} \hat{\rho}_i}, \quad 1 \leq k \leq M,
\]

and

\[
k^* = \max \{ 1 \leq k \leq M \mid \hat{a}_k \geq \hat{\rho}_k \} = \min \{ 1 \leq k \leq M \mid \hat{a}_k < \hat{\rho}_k \} - 1,
\]

with the convention that \( \hat{a}_k = +\infty \) if \( \rho_k = 0 \) and that \( \min \emptyset = M + 1 \). The bandwidth \( \hat{a} > 0 \) can be expressed as \( \hat{a} = \hat{a}_{k^*} \). Moreover, \( k^* \) is also the unique integer \( k \in \{ 1, \cdots, M \} \) such that \( \hat{a}_k \geq \hat{\rho}_k \) and \( \hat{a}_{k+1} < \hat{\rho}_{k+1} \) if \( k < M \).

For an approximation of

\[
\rho^{2}_{u,x_0}(x) = (u(x) - u(x_0))^2,
\]

the law of large numbers hints that

\[
\rho^{2}_{u,x_0}(x) \approx \|v(N_{x_0,d}) - v(N_{x,d})\|^2_2 - 2\sigma^2,
\]

where \( \sigma^2 \) is the variance of Gaussian noise. The filter needs an estimate of \( \rho_{u,x_0}(x) \) without the square. As shown in [25], in practice, rather than extracting the root in (10), good denoising results are obtained by using the approximation

\[
\rho_{u,x_0}(x) \approx \hat{\rho}_{x_0}(x) = \left( \|v(N_{x_0,d}) - v(N_{x,d})\|^2_2 - \sqrt{2\sigma} \right)_+.
\]

The fact that \( \hat{\rho}_{x_0}(x) \) is a reasonable estimator of \( \rho_{u,x_0} \) is justified by the convergence results in [25] (cf. Theorems 3 and 4 of [25]).

2.3. Review on the ROAD statistic and the TriF. Garnett et al. [18] introduced the ROAD (Rank-Ordered Absolute Differences) statistic to detect points contaminated by impulse noise. For any pixel \( x_0 \in I \), the ROAD statistic at \( x_0 \) with the detection \( R \times R \) window \( \mathcal{N}_{x_0,R} \) is defined by

\[
\text{ROAD}(x_0) = \sum_{i=1}^{K} r_i(x_0),
\]

where \( r_i(x_0) \) is the \( i \)-th smallest term in the set \( \{ |v(x) - v(x_0)| : x \in \mathcal{N}_{x_0,R}, x \neq x_0 \} \), \( 2 \leq K < R^2 \) is a parameter. In [18] it is advised to use \( R = 3 \) and \( K = 4 \) (half of the cardinality of the detection window \( \mathcal{N}_{x_0,R} \setminus \{ x_0 \} \)). Note that if \( x_0 \) is an impulse noisy point, then the value of \( \text{ROAD}(x_0) \) is large; otherwise it is small. The impulsive weight

\[
w_I(x) = e^{-\frac{\text{ROAD}(x)^2}{2\sigma_f}}
\]
Removing random-valued impulse noise with Reliable Weight

is introduced to express how impulse-like the pixel $x$ is. The parameter $\sigma_I$ determines the approximate threshold above which to penalize high ROAD values. The joint impulsivity between $x_0$ and $x$ is defined by

$$J(x_0, x) = 1 - e^{-\frac{1}{2\sigma_I^2} \left( \frac{\text{ROAD}(x_0) + \text{ROAD}(x)}{2} \right)^2},$$

where the parameter $\sigma_J$ controls the shape of the function $J(x_0, x)$. It is obvious that $J(x_0, x)$ takes values in the interval $[0, 1]$. If at least one of $x_0$ or $x$ is impulse-like and has a high ROAD value with respect to $\sigma_J$, then $J(x_0, x) \approx 1$. If neither pixel is impulse-like, then neither has a high ROAD value, and $J(x_0, x) \approx 0$. The TriF (cf. [18]) is given by

$$\text{TriF}(x_0) = \frac{\sum_{x \in \mathcal{N}_{x_0}, D} w(x_0, x) v(x)}{\sum_{x \in \mathcal{N}_{x_0}, D} w(x_0, x)},$$

where

$$w(x_0, x) = w_S(x_0, x) w_r(x_0, x)^{1 - J(x_0, x)} w_I(x)^{J(x_0, x)},$$

$$w_S(x_0, x) = e^{-\frac{||x - x_0||^2}{2\sigma_S^2}},$$

$$w_r(x_0, x) = e^{-\frac{(v(x) - v(x_0))^2}{2\sigma_R^2}}.$$
the value of $R$ should be chosen). To overcome this difficulty, we introduce the standardization of the ROAD statistic which we call SROAD:

\begin{equation}
\text{SROAD}(x_0) = \frac{1}{K} \sum_{i=1}^{K} r_i(x_0), \quad x_0 \in I.
\end{equation}

We extend the definition of SROAD to $I_0$ by taking

\[ \text{SROAD}(x_0) = \text{SROAD}(x'_0), \quad x_0 \in I_0. \]

If in the definition we remove the factor $1/K$, we then get the ROAD statistic introduced by Garnett et al. [18]. The factor $1/K$ makes the statistic less sensible to the variation of the detection window size $R$, as well as to the choice of the parameter $K$. The important feature of SROAD statistic, compared to the ROAD statistic, is that it is an average measure for the impulse characteristic relatively independent of the detection window size. In the above example, the SROAD value of the center point is the same value 35, in both cases $R = 3$ (with $K = 4$) and $R = 5$ (with $K = 12$). Accordingly, the SROAD statistic is also less sensible to the impulse level $p$, since the choice of the detection window size $R$ depends on the impulse level.

We now define the Reliable Weight of a pixel $x$ in terms of the SROAD statistic:

\begin{equation}
w_H(x) = e^{-\frac{(\text{SROAD}(x) - b)_+^2}{2H^2}}, \quad x \in I_0,
\end{equation}

where $b$ is the threshold for SROAD($x$) under which the pixel $x$ is considered to be noise-free ( $x \in I$ is noise-free when SROAD($x$) < $b$, but in our algorithm we will not judge whether a pixel is noise-free), and $H$ is a parameter to determine...
Figure 2. Mean values of ROAD of noisy pixels and uncorrupted pixels in the Lena image as a function of the impulse noise probability, with standard deviation error bars.

Figure 3. Mean values of ROLD of noisy pixels and uncorrupted pixels in the Lena image as a function of the impulse noise probability, with standard deviation error bars.
Figure 4. Mean values of Reliable Weight of noisy pixels and uncorrupted pixels in the Lena image as a function of the impulse noise probability, with standard deviation error bars.

The Reliable Weight \( w_H(x) \) indicates the reliability for the information of the pixel \( x \) to be used. By definition the value of \( w_H(x) \) is located in the dynamic range \([0, 1]\). If \( w_H(x) = 1 \), we consider that the pixel \( x \) is noise-free and the intensity value will remain the same in the process of denoising (the weighted means), while \( w_H(x) = 0 \) means that the pixel \( x \) completely loses of information and will not contribute in the weighted means. When \( w_H(x) \in (0, 1) \), the pixel is partially reliable; larger the value of \( w_H(x) \), more the pixel \( x \) contributes in the weighted means. The reason why we introduce the threshold \( b \) in the definition of Reliable Weight is that the difference of intensity less than some value can be ignored. For example, for an 8-bit gray-level image, if the difference is less than 8, it is not visually noticeable [7]. So it is reasonable to set \( w_H(x) = 1 \) when SROAD \( \leq 4 \), namely \( b = 4 \), taking into account that for any \( x', x'' \in N_{x,R} \setminus \{x\} \) such that \( |u(x) - u(x')| \leq 4 \) and \( |u(x) - u(x'')| \leq 4 \), we have \( |u(x') - u(x'')| \leq 8 \). Simulation results confirm that it is reasonable to take \( b = 4 \) (see Fig. 1).

Figs. 2-4 show, for the Lena image, the mean value for each of the three statistics ROAD, ROLD, and Reliable Weights, on the set of noisy pixels and on the set of uncorrupted pixels, as a function of the impulse noise probability, with standard deviation error bars demonstrating the significance of the difference. The first chart illustrates the means of ROAD values with \( R = 3 \) (3 \times 3 windows) and \( K = 4 \) (cf. Garnett et al. [18]), the second one displays the means of ROLD values with \( R = 5 \) (5 \times 5 windows) and \( K = 12 \) (cf. Dong et al. [17]), and the last one gives the means of Reliable Weights values with our adaptive parameters. We can see from the charts
that the distance of the mean values of Reliable Weights between noisy pixels and uncorrupted pixels is larger than those for the ROAD and ROLD statistics (the difference is very significant while comparing with ROAD, and remains interesting while comparing with ROLD), so that it is easier to separate noisy pixels from noise-free pixels while using Reliable Weights rather than the ROAD statistic or ROLD statistic. Thus, the Reliable Weights improve the accuracy in the detection of impulse noisy points, compared with ROAD and ROLD. In the next section, we will explain how the Reliable Weights contribute into our filter.

3.2. Similarity function between two patches. For \( x_0, x \in I_0 \), we define the similarity function, also called radiometric distance between \( v(N_{x_0,d}) \) and \( v(N_{x,d}) \) by

\[
\|v(N_{x_0,d}) - v(N_{x,d})\|^2_{H,\kappa} = \frac{\sum_{t \in N_{0,d}} \kappa(t)w_H(x + t)w_H(x_0 + t)(v(x_0 + t) - v(x + t))^2}{\sum_{t \in N_{0,d}} \kappa(t)w_H(x + t)w_H(x_0 + t)},
\]

where the smoothing kernel \( \kappa \) is defined by

\[
\kappa(t) = \frac{(d - 1)/2}{\sum_{k = \max(1, j)}^{(d - 1)/2} (2k + 1)^2}
\]

if \( \|t\|_\infty = j \) for some \( j \in \{0, 1, \cdots, (d - 1)/2\} \), for \( t \in N_{0,d} \). It is possible to use the Gaussian kernel for the choice of \( \kappa \), but the results are then a bit less precise. We use the product \( w_H(x + t)w_H(x_0 + t) \) of Reliable Weights of the pixels \( x + t \) and \( x_0 + t \) to reduce the effects of impulse noise, so that the similarity function has significant anti impulse noise interference ability.

We will prove that the radiometric distance (18) converges to \( \rho_{u,x_0}(x) = |u(x_0) - u(x)| \) under the Hölder regularity condition.

**Definition 3.1.** The image function \( u \) on \( I \) is called Hölder if for some constants \( L > 0, \alpha > 0 \) and all \( x, y \in I_0 \),

\[
|u(x) - u(y)| \leq L \|x - y\|^\alpha.
\]

We need to suppose that the impulse noisy pixels can be separated from the noise-free ones by using the reliable weights \( w_H(x) \). We say that a point \( x \in I \) is contaminated by impulse noise is \( v(x) \neq u(x) \). Let \( B \) be the set of pixels \( x \in I \) contaminated by impulse noise, and \( \overline{B} = I \setminus B \) its complement. The following hypothesis implies that with high probability \( w_H(x) \) is small for all contaminated pixels \( x \in B \) and large for non-contaminated pixels \( x \in \overline{B} \), where \( H \) is a parameter whose value will be precised in theorems below.

**Hypothesis 1.** Assume that:

a) \( R \to \infty \) and

\[
\sup_{x \in B} w_H(x) \to 0 \text{ in probability as } R \to \infty.
\]

b) There exists some constant \( c > 0 \), such that

\[
\inf_{x \in \overline{B}} w_H(x) \geq c_H
\]

with a probability converging to 1 as \( R \to \infty \), where \( c_H \) is a constant depending only on \( H \).
Theorem 3.2. Assume Hypothesis 1 with $H$ the parameter in the definition of the reliable weights (17). Assume also the Hölder condition (20). Then for $x_0, x \in \mathbf{I}_0$,

\begin{equation}
\|v(N_{x_0,d}) - v(N_{x,d})\|_{H,\kappa}^2 \rightarrow \|u(x_0) - u(x)\|^2
\end{equation}

in probability, as $d \rightarrow \infty$ and $d/N \rightarrow 0$.

The proof is given in Section 5.1. Theorem 3.2 shows that $\|v(N_{x_0,d}) - v(N_{x,d})\|_{H,\kappa}$ is a convergent estimator of $|u(x_0) - u(x)|$. Consider the similarity distance given by

\begin{equation}
\hat{\rho}_{H,\kappa,x_0}(x) = \left(\|v(N_{x_0,d}) - v(N_{x,d})\|_{H,\kappa} - \eta\right)_+, \quad x_0, x \in \mathbf{I}_0,
\end{equation}

where $\eta > 0$ is a parameter. The estimate $\hat{\rho}_{H,\kappa,x_0}(x)$ will play the same role as the estimate $\hat{\rho}_{x_0}(x)$ defined by (11) used for the Gaussian noise case. The introduction of $\eta$ allows to increase the contribution in the weighted means of the patches which are very similar. By Theorem 3.2, $\hat{\rho}_{H,\kappa,x_0}(x)$ is a convergent estimator of

\begin{equation}
\rho_{x_0}(x) = (|u(x_0) - u(x)| - \eta)_+.
\end{equation}

3.3. Construction of reliable radiometric weight filter. In this section, a new filter is designed to remove random-valued impulse noise using the Reliable Weights and the similarity function.

Inspired by the construction of the Optimal Weights Filter [25] originally proposed to remove the Gaussian noise, for the impulse noise model we introduce the radiometric weight between the patches $v(N_{x_0,d})$ and $v(N_{x,d})$ by

\begin{equation}
w(x_0, x) = \frac{\kappa_{tr} \left(\frac{\hat{\rho}_{H,\kappa,x_0}(x)}{\hat{\rho}_{H}(x)}\right)}{\sum_{y \in N_{x_0,d}} \kappa_{tr} \left(\frac{\hat{\rho}_{H,\kappa,x_0}(y)}{\hat{\rho}_{H}(y)}\right)}, \quad x_0, x \in \mathbf{I}_0,
\end{equation}

where $\hat{\rho}_{H,\kappa,x_0}(x)$ is the estimate of $\rho_{x_0}(x)$ (see (22) and (23) respectively), the bandwidth $\hat{a}_H > 0$ is the solution of the equation

\begin{equation}
\sum_{x \in N_{x_0,d}} \frac{1}{\sigma^2} \hat{\rho}_{H,\kappa,x_0}(x)(\hat{a}_H - \hat{\rho}_{H,\kappa,x_0}(x))_+ = 1,
\end{equation}

with $\sigma > 0$ a parameter. The calculation of the bandwidth $\hat{a}_H > 0$ is done using the algorithm presented in Remark 1 with $\hat{\rho}_{x_0}(x)$ and $\hat{a}$ replaced by $\hat{\rho}_{H,\kappa,x_0}(x)$ and $\hat{a}_H$.

We will choose $\eta = \sqrt{2}b = 4\sqrt{2}$, see Section 4.1. In the original Optimal Weights Filter [25] designed to remove the Gaussian noise, $\sigma$ is the standard deviation of the Gaussian noise. In the impulse noise model, there is no Gaussian noise, so that $\sigma = 0$; in order to use the above formula we can suppose that in the impulse noise case there is still a Gaussian noise with small value of $\sigma > 0$. As differences of gray values smaller than 8 are not visually noticeable, it would be reasonable to take the value of $\sigma$ around 8; the existence of the impulse noise would increase the value of $\sigma$. Experiments show that a good choice of $\sigma$ is $\sigma = 10$ if $p \leq 50$. For a more complete discussion about the choice of $\sigma$ see Section 4.1.

The radiometric weight $w(x_0, x)$ decreases as the radiometric distance between $v(N_{x_0,d})$ and $v(N_{x,d})$ increases. The Reliable Weights $w_{H_i}(x)$ introduced in Section 3.1 can be used to improve the radiometric weights $w(x_0, x)$, in such a way that pixels with larger reliable weights contribute more in the weighted mean. Notice
Figure 5. The evolution of PSNR value as a function of the size of image. Here we take the $1024 \times 1024$ test image Male as example, and zoom it out into $512 \times 512$, $256 \times 256$, $128 \times 128$ and $64 \times 64$ images.

Figure 6. The evolution of PSNR value as a function of the size of similarity patch with level of the impulse noise $p = 20\%$.

that $H$ and $H_1$ may take different values. The choice of their values will be discussed in the next section. If $w_{H_1}(x_0) = 1$, we can consider that $x_0$ is noise free without risk (due to the choice $b = 4$), so that the intensity value of the pixel remains
unchanged. If $w_{H_1}(x_0) < 1$, we define the Reliable Radiometric Weight of $x$ with
respect to the central point $x_0$ as

$$\hat{w}_D(x_0, x) = \frac{w_{H_1}(x) \kappa_{tr} \left( \frac{\hat{\rho}_{H,\kappa,x_0}(x)}{\hat{\sigma}_H} \right)}{\sum_{y \in N_{x_0,D}} w_{H_1}(y) \kappa_{tr} \left( \frac{\hat{\rho}_{H,\kappa,x_0}(y)}{\hat{\sigma}_H} \right)}, \quad x_0 \in I_0, \ x \in I.$$  

Then the filter RRWF is defined by: for each $x_0 \in I_0$,

$$\hat{u}_D(x_0) = \begin{cases} \sum_{x \in N_{x_0,0}} \hat{w}_D(x_0, x) v(x) & \text{if } w_{H_1}(x_0) < 1, \\ v(x_0) & \text{if } w_{H_1}(x_0) = 1. \end{cases}$$  

As the radiometric weights $\hat{w}_D(x_0, x)$ take into account not only large impulses (which are detected with high reliability as impulse noisy points in a two-step approach) but also small impulses (which may be detected by error as noise-free in a two-step approach), the new filter can remove not only the larger outliers, but also smooth away smaller impulses without blurring edges. It can still remove most of the noise while preserving image details, even when the random-valued impulse noise ratio is as high as 60%.

To justify the convergence of the filter $\hat{u}_D(x_0)$, we introduce the oracle filter $u_D(x_0)$, which is obtained while replacing $\hat{\rho}_{H,\kappa,x_0}(x)$ by $\rho_{x_0}(x)$ in (25) and (26):

$$u_D(x_0) = \begin{cases} \sum_{x \in N_{x_0,0}} w_D(x_0, x) v(x) & \text{if } w_{H_1}(x_0) < 1, \\ v(x_0) & \text{if } w_{H_1}(x_0) = 1, \end{cases}$$

where

$$w_D(x_0, x) = \frac{w_{H_1}(x) \kappa_{tr} \left( \frac{\rho_{x_0}(x)}{a} \right)}{\sum_{x \in N_{x_0,0}} w_{H_1}(x) \kappa_{tr} \left( \frac{\rho_{x_0}(x)}{a} \right)}$$

with $a > 0$ defined by

$$\sum_{x \in N_{x_0,D}} \frac{1}{\sigma^2} \rho_{x_0}(x) (a - \rho_{x_0}(x))_+ = 1.$$

The following theorem shows that the oracle filter $u_D(x_0)$ converges to $u(x_0)$ under suitable conditions. By Theorem 3.2, we know that $\hat{\rho}_{H,\kappa,x_0}(x)$ is a convergent estimator of $\rho_{x_0}(x)$. This justifies that the filter RRWF $\hat{u}_D(x_0)$ is also a reasonable estimator of $u(x_0)$.

**Theorem 3.3.** Assume Hypothesis 1 with $H$ replaced by $H_1$ from (28). Assume also the Hölder condition (20). Then, for every $x_0 \in I_0$,

$$u_D(x_0) \rightarrow u(x_0)$$

in probability, as $D \to \infty$ and $D/N \to 0$.

The proof of this theorem is deferred to Section 5.2.

In Theorems 3.1 and 3.3 have assumptions $N \to +\infty$, $d/N \to 0$ and $D/N \to 0$. It is expected that the restoration quality increases when $N$ increases and $d/N \to 0$, $D/N \to 0$. This is confirmed by our simulation results presented in Fig. 5. Actually in Fig. 5, we take the image Male with different resolutions $N$ and we can see that the PSNR value is a increasing function of $N$. As the resolution increases, the PSNR value rises significantly. We have chosen the size $d$ of the similarity patch and the size $D$ of the research window as constant, so that the ratios $d/N$ and $D/N$
become smaller with large $N$. However, as shown by the simulation results in Figs. 4, 5 and 6, when we increase the values of $d$ the quality of restoration increases. We avoided to increase the values of $D$ because of the very high computational cost. So our simulations results are coherent with the theoretical results of Theorems 3.2 and 3.3: the PNR values increase when $N$ increases and the ratios $d/N$ and $D/N$ decrease.

4. Simulation. In this section, the proposed algorithm is evaluated and compared with several other existing techniques for removing random-valued impulse noise. Extensive experiments are conducted on standard $512 \times 512$, 8-bit gray-level images with different features, including Baboon, Bridge, Lena and Pentagon (see Fig. 9 (a) - (d)). Our experiments are done in the same way as in [17] in order to produce comparable results. The authors of [17] kindly provided us with their set of noisy images, restored images and PNSR values.

4.1. Algorithm and parameters. The computational algorithm is given in Algorithm 1.

The parameters we give here are suited to 8-bit gray-level images with which we do simulations. In section 3.1 we have already discussed on the choice of the parameter $b$ (appearing in the definition of the reliable weight (17)) : a good choice is $b = 4$ starting from the fact that differences of gray values less than 8 are not visually noticeable (see [7]). For the parameter $\eta$ introduced in Eq. (22) the same consideration leads to the choice $\eta = \sqrt{2b}$, which is confirmed to be good by simulation results.

We next discuss on the choice of the other parameters $d, D, \sigma, R, K, H$ and $H_1$ based on experimental results. In the series of experiments that we did, we assume that the parameters $\{p, d, D, \sigma, R, K, H, H_1\}$ take values in the set $\Psi = A_1 \times A_2 \times A_3 \times A_4 \times A_5 \times A_6 \times A_7 \times A_8$, where

$$
A_1 = \{0.1, 0.2, 0.3, \ldots, 0.6\}, \\
A_2 = \{5.7, 9, \ldots, 31\}, \\
A_3 = \{3, 5, 7, \ldots, 11\}, \\
A_4 = \{1, 2, 3, \ldots, 40\}, \\
A_5 = \{3, 5, 7, 9, 11\}, \\
A_6 = \{2, 3, \ldots, ((2d + 1)^2 - 1)/2\}, \\
A_7 = \{1, 2, 3, \ldots, 40\}, \\
A_8 = \{1, 2, 3, \ldots, 40\}.
$$

For each $p \in A_1$, we maximize the PSNR as a function of $d, D, \sigma, R, K, H$ and $H_1$, and we obtain simple formulas for the $d, D, \sigma, R, K, H$ and $H_1$ (as function of $p$) where the maximum is attained. To do this, for each $p$, we let $d, D, \sigma, R, K, H$ and $H_1$ ran over the set $A_2 \times A_3 \times A_4 \times A_5 \times A_6 \times A_7 \times A_8$ and record the maximal values of PSNR and the maximizers $d, D, \sigma, R, K, H$ and $H_1$. In our simulations, the probability $p$ is assumed to be known.

In the following, we find expressions to approximate the maximizers $d, D, \sigma, R, K, H$ and $H_1$ in function of $p$ using the Lena image; the validity of the expressions is then confirmed by experiments on other images (cf. Tables 1). For example, the

---

1The PSNR values marked with * come from [17], the others are calculated with the code provided by the authors.
Removing random-valued impulse noise with Reliable Weight

dependence of $d$ on $p$ is tested on the Lena image with different values of $p$. In Figs. 6-8 we display the PSNR values in function of $d$ with different levels of the impulse noise $p = 20\%, 40\%, 60\%$, with search window size $D = 3, 5, 7$ respectively, and $\sigma, R, K, H$ and $H_1$ given by (32)-(36) respectively. These values of $D, \sigma, R, K, H$ and $H_1$ are maximizers of the PSNR. For $p = 20\%$, $p = 40\%$ and $p = 60\%$, graphs in Figs. 6-8 illustrate that the PSNR value increases with $d$. We note that the growth of the PSNR is not significant for $d \geq 19$ when $p = 20\%$, for $d \geq 23$ when $p = 40\%$, and for $d \geq 27$ when $p = 60\%$. Since a larger width of similarity patches increases the cost of computation, we take the similarity patch size $d \geq 19$ for $p = 20\%$, $d \geq 23$ for $p = 40\%$, $d \geq 27$ for $p = 60\%$. The above mentioned analysis leads us to the choice as a function of $p$ given by

$$d = 2 \times [10p + 7] + 1,$$

where $[x]$ denotes the integral part of $x$. Similar consideration lead to the following formulas for the choice of other parameters (as functions of $p$):

$$D = 2 \times [4p + 1] + 1;$$

$$\sigma = \begin{cases} 10 & \text{if } p \leq 50\%, \\ 10 + (100p - 50)/2 & \text{if } p > 50\%; \end{cases}$$

$$R = 2 \times [4p + 1] + 1;$$

$$K = [(2d + 1)^2 - 1] \times \min(0.5, -p/4 + 0.55);$$

$$H = 5 + \frac{30}{1 + 20p};$$

$$H_1 = 27 - 20p.$$

Note that our choice of parameters is different from that proposed in [18]: here we find suitable to use detection window size $R$ given by (33), while [18] proposes $R = 3$ in most cases; for impulse noise with probability $p = 40\%$ our formula gives $R = 5$ for the calculation of SROAD, while [18] proposes to use always $R = 3$. Simulation results show that our choice of parameters give better restoration results. Moreover, our filter can remove efficiently impulse noise until the noise level $p = 60\%$ (using always the above formulas for the choice of parameters), while the filters proposed in [18] and [21] do not work well for $p = 60\%$. An important point is that the choice of parameters proposed in [18] and [21] are no longer suitable for $p = 60\%$, mainly due to the non-stability of the ROAD statistic used therein.

All the formulas above are initially obtained from experiments using the Lena image, but they work also well for other images like Baboon, Bridge, Pentagon, etc.

4.2. Comparison of PSNR values. We first concentrate on directly comparable and quantitative measures of restoration quality. We evaluate the performance by using the Peak Signal-to-Noise Ratio (PSNR) [7]. If $u$ is the original image and $\hat{u}$ is a restored image of $u$, the PSNR of $\hat{u}$ is given by

$$\text{PSNR}(\hat{u}_h) = 10 \log_{10} \frac{255^2}{\text{MSE}},$$

$$\text{MSE} = \frac{1}{\text{card}I} \sum_{x \in I} (\hat{u}(x) - u(x))^2.$$

Larger PSNR value signifies better restoration result.
Figure 9. The set of 20 tested images. The commonly-used images (a)-(d) are provided by the authors of [17]; the images (e)-(m) are taken from USC-SIPI Image Database: http://sipi.usc.edu/database/; the images (n)-(t) are from: http://www.cs.tut.fi/~foi/GCF-BM3D/.
Algorithm 1 Reliable Radiometric weight Filter

Require: Noisy image \( v(x), x \in I \)
1: Parameters: \( \{d, D, \sigma, R, K, H, H_1\} \) are depended on \( p \). The Eqs (30) - (36) show the relations between \( \{d, D, \sigma, R, K, H, H_1\} \) and \( p \).

Ensure: Estimated image \( \hat{u}_D(x), x \in I \)
2: for all \( x \in I \) do
3: calculate \( w_H(x) \) and \( w_{H_1}(x) \) defined by (17)
4: end for
5: for each \( x \in I \) do
6: if \( w_{H_1}(x) = 1 \) then
7: \( \hat{u}_D(x_0) = u(x_0) \)
8: else
9: compute \( \{\hat{\rho}_{H,\kappa,x_0}(x) : x \in \mathcal{N}_{x_0,D}\} \) by (22)
10: reorder \( \{\hat{\rho}_{H,\kappa,x_0}(x) : x \in \mathcal{N}_{x_0,D}\} \) as an increasing sequence, say \( \hat{\rho}_{H,\kappa,x_0}(x_1) \leq \hat{\rho}_{H,\kappa,x_0}(x_2) \leq \cdots \leq \hat{\rho}_{H,\kappa,x_0}(x_M) \)
11: for \( k = 1 \) to \( M \) do
12: if \( \sigma^2 + \sum_{i=1}^{k} \hat{\rho}_{H,\kappa,x_0}(x_i) \geq \hat{\rho}_{H,\kappa,x_0}(x_k) \) then
13: compute \( \hat{a}_H = \frac{\sigma^2 + \sum_{i=1}^{k} \hat{\rho}_{H,\kappa,x_0}(x_i)}{\sum_{i=1}^{k} \hat{\rho}_{H,\kappa,x_0}(x_i)} \)
14: else
15: quit loop
16: end if
17: for all \( x \in \mathcal{N}_{x_0,D} \) do
18: \( \hat{w}(x_0, x_i) = \frac{w_{H_1}(x_i)K_t\left(\frac{\hat{\rho}_{H,\kappa,x_0}(x_i)}{\hat{a}_H}\right)}{\sum_{j=1}^{M} w_{H_1}(x_j)K_t\left(\frac{\hat{\rho}_{H,\kappa,x_0}(x_j)}{\hat{a}_H}\right)} \)
19: end for
20: compute \( \hat{u}_D \) at \( x_0 : \hat{u}_D(x_0) = \sum_{i=1}^{M} \hat{w}(x_0, x_i)v(x_i) \).
21: end for
22: end if
23: end for
24: To avoid the undesirable border effects, in our simulations we mirror the image outside the image limits symmetrically with respect to the border. At the corners, the image is extended symmetrically with respect to the corner pixels.
Table 1. Comparison of restoration in PSNR (db) for images corrupted with random-valued impulse noise.

| Images | Baboon | Bridge | Lena | Pentagon | Average |
|--------|--------|--------|------|----------|---------|
|        | 20%    | 40%    | 60%  | 20%      | 40%     | 60%     | 20%     | 40%     | 60%     |
| p      |        |        |      |          |         |         |         |         |         |
| Method | PSNR   | PSNR   | PSNR | PSNR     | PSNR    | PSNR    | PSNR    | PSNR    | PSNR    |
| MF     | 22.52db| 20.65db| 19.36db| 25.044db | 22.17db | 19.364db| 25.044db| 22.17db | 19.364db|
| SS-II  | 21.67db| 20.85db| 21.97db| 26.264db | 22.96db | 19.134db| 23.26db | 22.96db | 19.134db|
| SD-ROM | 22.46db| 21.35db| 19.42db| 25.996db | 22.85db | 19.04db | 27.53db| 22.85db | 19.04db |
| PWMAD  | 21.81db| 21.49db| 19.45db| 26.565db | 23.86db | 20.064db| 25.71db | 23.86db | 20.064db|
| Luo-IMF | 23.43db| 21.07db| 19.56db| 26.333db | 22.75db | 19.734db| 28.92db| 22.75db | 19.734db|
| FWNLM  | 23.73db| 21.38db| 19.44db| 26.524db | 22.89db | 19.04db | 28.36db| 22.89db | 19.04db |
| MSM    | 24.02db| 21.52db| 19.83db| 27.57db  | 23.55db | 20.075db| 35.44db| 29.26db | 22.14db |
| ACWM   | 24.17db| 21.58db| 19.56db| 27.08db  | 23.23db | 19.25db | 36.07db| 28.79db | 21.19db |
| PWMAD  | 23.78db| 21.56db| 19.84db| 26.904db | 23.83db | 20.83db | 36.50db| 31.41db | 24.30db |
| Luo-IMF| 24.18db| 21.41db| 19.08db| 27.05db  | 23.88db | 19.74db | 36.90db| 30.25db | 22.96db |
| Trf    | 24.18db| 21.60db| 19.52db| 27.60db  | 24.01db | 20.84db | 36.70db| 31.12db | 26.08db |
| ACWM-EPR | 23.97db| 21.62db| 19.87db| 27.314db | 24.06db | 20.894db| 36.57db| 32.21db | 24.62db |
| ROAD-EPR | 23.12db| 21.56db| 19.96db| 27.42db  | 24.52db | 22.04db | 36.79db| 32.32db | 28.37db |
| OWL-EPR | 23.46db| 21.92db| 20.38db| 27.86db  | 24.79db | 22.59db | 37.45db| 32.76db | 29.03db |
| FWNLM  | 24.17db| 21.71db| 20.45db| 26.824db | 24.23db | 22.23db | 34.95db| 32.12db | 28.03db |
| PWMF   | 24.85db| 22.00db| 14.72db| 28.124db | 24.71db | 11.58db | 37.06db| 33.54db | 30.41db |
| PARIGI | 24.46db| 21.85db| 19.79db| 26.538db | 24.06db | 21.94db | 36.62db| 31.94db | 27.61db |
| RRVF   | 25.01db| 22.41db| 20.46db| 27.86db  | 24.91db | 22.98db | 37.56db| 33.07db | 29.05db |

190
Table 2. Comparison of restoration in PSNR(db) for images corrupted with random-valued impulse noise with $p = 20$.

| Images   | Aerial | Airplane | B. wall | Cam | Clock | Male512 | Male1024 | M. surface | P. bubbles | R. chart | Barbara | boat | couple | F. print | hill | house | Average |
|----------|--------|----------|---------|-----|-------|---------|----------|------------|------------|-----------|---------|--------|-------|--------|----------|------|-------|---------|
| Method   | PSNR   | PSNR     | PSNR    | PSNR| PSNR  | PSNR    | PSNR     | PSNR       | PSNR       | PSNR      | PSNR    | PSNR   | PSNR  | PSNR   | PSNR   | PSNR | PSNR | PSNR   |
| FWNLM [49] | 36.50db | 40.78db  | 36.47db | 38.63db | 40.02db | 38.74db | 39.59db | 37.08db | 38.56db | 38.51db | 38.09db | 36.56db | 38.85db | 40.51db | 38.37db |
| PWML [21]| 36.04db | 39.77db  | 36.15db | 38.52db | 39.52db | 38.38db | 38.78db | 36.03db | 37.77db | 36.62db | 37.51db | 37.86db | 38.54db | 39.85db | 37.83db |
| PARIGI [15, 16] | 33.13db | 40.46db  | 32.64db | 36.63db | 38.95db | 35.71db | 37.09db | 34.79db | 34.85db | 38.35db | 37.06db | 35.90db | 35.67db | 32.07db | 36.54db | 39.89db |
| RRWF     | 36.40db | 41.73db  | 37.03db | 38.36db | 40.27db | 36.46db | 30.56db | 37.61db | 37.57db | 37.80db | 38.87db | 38.41db | 36.24db | 39.00db | 41.20db | 38.54db |

Table 3. Comparison of restoration in PSNR(db) for images corrupted with random-valued impulse noise with $p = 40$.

| Images   | Aerial | Airplane | B. wall | Cam | Clock | Male512 | Male1024 | M. surface | P. bubbles | R. chart | Barbara | boat | couple | F. print | hill | house | Average |
|----------|--------|----------|---------|-----|-------|---------|----------|------------|------------|-----------|---------|--------|-------|--------|----------|------|-------|---------|
| Method   | PSNR   | PSNR     | PSNR    | PSNR| PSNR  | PSNR    | PSNR     | PSNR       | PSNR       | PSNR      | PSNR    | PSNR   | PSNR  | PSNR   | PSNR   | PSNR | PSNR | PSNR   |
| FWNLM [49] | 33.17db | 37.95db  | 33.32db | 33.56db | 36.91db | 35.06db | 35.26db | 35.09db | 34.47db | 35.80db | 34.80db | 34.96db | 34.86db | 32.35db | 35.56db | 37.54db |
| PWML [21]| 33.74db | 38.14db  | 33.47db | 36.09db | 37.50db | 35.44db | 34.77db | 34.88db | 35.98db | 34.64db | 34.26db | 34.95db | 35.02db | 35.96db | 37.94db | 35.51db |
| PARIGI [15, 16] | 30.80db | 36.34db  | 30.93db | 34.56db | 34.72db | 33.01db | 34.15db | 33.22db | 32.75db | 33.66db | 33.08db | 32.90db | 32.45db | 28.99db | 33.69db | 36.29db |
| RRWF     | 33.43db | 38.37db  | 34.21db | 35.79db | 36.94db | 35.41db | 36.70db | 35.56db | 35.04db | 35.74db | 34.78db | 35.54db | 35.19db | 32.55db | 36.25db | 38.41db | 35.62db |

Table 4. Comparison of restoration in PSNR(db) for images corrupted with random-valued impulse noise with $p = 60$.

| Images   | Aerial | Airplane | B. wall | Cam | Clock | Male512 | Male1024 | M. surface | P. bubbles | R. chart | Barbara | boat | couple | F. print | hill | house | Average |
|----------|--------|----------|---------|-----|-------|---------|----------|------------|------------|-----------|---------|--------|-------|--------|----------|------|-------|---------|
| Method   | PSNR   | PSNR     | PSNR    | PSNR| PSNR  | PSNR    | PSNR     | PSNR       | PSNR       | PSNR      | PSNR    | PSNR   | PSNR  | PSNR   | PSNR   | PSNR | PSNR | PSNR   |
| FWNLM [49] | 31.84db | 37.39db  | 32.28db | 33.54db | 35.43db | 32.64db | 33.63db | 33.64db | 32.79db | 34.60db | 32.55db | 33.16db | 32.67db | 30.12db | 33.55db | 35.60db |
| PWML [21]| 30.13db | 35.83db  | 30.19db | 33.44db | 33.71db | 31.50db | 32.27db | 32.14db | 30.55db | 34.56db | 31.18db | 31.62db | 31.04db | 28.21db | 31.75db | 34.56db | 32.04db |
| PARIGI [15, 16] | 32.07db | 37.75db  | 32.73db | 33.59db | 35.35db | 32.61db | 33.70db | 33.81db | 33.21db | 34.10db | 33.06db | 33.32db | 33.00db | 30.18db | 33.74db | 36.00db | 33.64db |
In Table 1, we list the best PSNR values from all considered methods for the four images with \( p \in \{20\%, 40\%, 60\%\} \). The best values are in bold so that they can be compared easily with the values from our method. From Table 1, it is clear that our method RRWF provides significant improvement over all other algorithms for the cases of Baboon, Lena and Pentagon. In the case of Bridge, PWMF and our algorithm both provide satisfactory denoising performance.

For a large set of images (see Fig. 9(e)-(t)), we compare our algorithm with the resent methods PWNLM, PWMF and PARIGI, see Tables 2-4. PWMF doesn’t work for the case \( p = 60 \), so we don’t put the results of PWMF in Table 4. From Tables 2-4 we can see that our method performs better for most of the images.

In Fig. 5 we give simulation results for different sizes of \( N \), which are in line with our theoretical results, see comments after Theorem 3.3.

4.3. Image quality. Our main goal is to ensure that our approach provides improved denoising and visually better results. To compare the results subjectively, we enlarge portion of the images restored by some methods listed in Table 1. Figs. 10 and 11 show the results in restoring 40\% corrupted images of Baboon and Lena respectively. In the images restored by ACWM-EPR [10] and TriF [18], we can see that there are still some noticeable noise remained in the face of the Lena, and some loss of details in the hair around the mouth of the baboon. The visual qualities of images restored by ROLD-EPR [17] are improved obviously, but we can still find a few noise around the nose of baboon and the face of Lena is not smooth enough. The restored images of PARIGI [15, 16] are too smooth and some details are removed. Our restored images are quite good, which not only remain the abundance of image details, but also keep continuity of the details. To further compare the capability of ROLD-EPR and our method, we provide the square error images which are calculated as

\[
\text{Difference}(x) = |\hat{u}(x) - u(x)|^2,
\]

where \( \hat{u}(x) \) is restored gray value, \( u(x) \) is original gray value, Difference\((x)\) is the square error value and \( \Gamma \) is a parameter to control the value range of square error image for better results. The blue image regions correspond to high-confidence estimates. As show in Fig. 14, the Difference images show that our results are better than others. For example, the hot points in our Difference image is significantly less than that of ROLD-EPR and PARIGI difference images. Figs. 12 and 13 show the results in restoring 60\% corrupted images of Bridge and Pentagon. There are still many noticeable noise patches in the images restored by ACWM-EPR [10]. In contrast, the images restored by TriF [18] are blurred significantly and lose so much details. ROLD-EPR [17] performs better, but it still leaves a little noise and remove some important details (see Fig. 12). Our method can suppress the noise successfully while preserving more details.

4.4. Computational complexity. Recall that the TriF [18] is defined by the weighted average estimate

\[
\text{TriF}(x_0) = \frac{\sum_{x \in N_{x_0, D}} w_S(x_0, x)w_r(x_0, x)^{1-J(x_0,x)}w_I(x)^{J(x_0,x)}v(x)}{\sum_{x \in N_{x_0, D}} w_S(x_0, x)w_r(x_0, x)^{1-J(x_0,x)}w_I(x)^{J(x_0,x)}},
\]
Figure 10. Results of different methods in restoring 40% corrupted Baboon image.
Figure 11. Results of different methods in restoring 40% corrupted Lena image.
Figure 12. Results of different methods in restoring 60% corrupted Bridge image.
Figure 13. Results of different methods in restoring 60% corrupted Pentagon image.
Removing random-valued impulse noise with Reliable Weight

Figure 14. The Difference between the denoised image with impulse noise level $p = 40\%$ and original image.

in which the trilateral weight $w_{tr}(x_0, x) = w_S(x_0, x)w_I(x_0, x)^{1-J(x_0, x)}w_{IR}(x)^{J(x_0, x)}$ is calculated $n \times D^2$ times, where $J(x_0, x) = 1 - e^{-\frac{1}{2\sigma^2} \left(\frac{\text{ROAD}(x_0)+\text{ROAD}(x)}{2}\right)^2}$ is the joint impulsivity weight. Instead, in our method, we calculate $n \times D^2$ times the reliable weight $w_r(x_0, x) = w_{H_1}(x)\kappa_{ir}\left(\frac{\hat{\rho}_{H,\kappa,x_0}(x)}{\hat{a}_{H}}\right)$. The main computational advantage in our approach is that we do not need to calculate the joint impulsivity weights, whose computational complexity is of order $O(D^2 \times n)$. Moreover, the calculation of the trilateral weight $w_{tr}(x_0, x)$ is more complicated than that of the reliable weight $w_r(x_0, x)$, because of the presence of the power operation in the definition of $w_{tr}(x_0, x)$. This explains why our filter RRWF (26) is much faster than the TriF [18]. See Table 5 for a comparison of simulation results of the average runtime with a set of $512 \times 512$ images.

In Table 5 we also do a comparison of speed with the recently proposed algorithms FWNLM [49], PWMF [21] and PARIGI [15, 16]. From the table, we see that our filter RRWF (26) is also much faster than FWNLM [49] and PARIGI [15, 16], and significantly faster than PWMF [21].
5. Proof of Theorems 3.2 and 3.3.

5.1. Proof of Theorem 3.2.

Proof. Fix \( x_0, x \in \mathbb{I}_0 \). Let \( \Delta v(t) = v(x_0' + t) - v(x' + t) \). Then

\[
\|v(N_{x_0,d}) - v(N_{x,d})\|^2_H = \|v(N_{x_0,d}) - v(N_{x,d})\|^2_H + \sum_{t \in N_{0,d}} \kappa(t)w_H(x_0' + t)w_H(x' + t)\Delta v(t)^2
\]

where

\[
J_1 = \sum_{t \in N_{0,d}} 1_{\{x_0' + t \in \mathbb{B} \text{ and } x' + t \in \mathbb{B} \}} \kappa(t)w_H(x_0' + t)w_H(x' + t),
\]

\[
J_2 = \sum_{t \in N_{0,d}} 1_{\{x_0' + t \in \mathbb{B} \text{ or } x' + t \in \mathbb{B} \}} \kappa(t)w_H(x_0' + t)w_H(x' + t),
\]

\[
R_1 = \sum_{t \in N_{0,d}} 1_{\{x_0' + t \in \mathbb{B} \text{ and } x' + t \in \mathbb{B} \}} \kappa(t)w_H(x_0' + t)w_H(x' + t)\Delta v(x')^2,
\]

\[
R_2 = \sum_{t \in N_{0,d}} 1_{\{x_0' + t \in \mathbb{B} \text{ or } x' + t \in \mathbb{B} \}} \kappa(t)w_H(x_0' + t)w_H(x' + t)\Delta v(x')^2.
\]

By the definition of \( \overline{\mathbb{B}} \) we have \( u(y) = v(y) \) when \( y \in \overline{\mathbb{B}} \). So when \( x_0' + t, x' + t \in \overline{\mathbb{B}} \), \( \Delta v(t) = u(x_0' + t) - u(x' + t) \). Since the image \( u \) satisfies the Hölder condition and \( \|x_0' + t - x_0\| \leq \|x_0' + t - x_0\| + \|x_0' - x_0\| \leq \frac{d}{2N} + \frac{1}{N} \leq \frac{d}{N} \), we have \( u(x_0' + t) = u(x_0) + O\left(\left(\frac{d}{N}\right)^\alpha\right)\) and \( u(x' + t) = u(x) + O\left(\left(\frac{d}{N}\right)^\alpha\right)\). Therefore, for any \( t \in N_{0,d} \) such that \( x_0' + t \in \overline{\mathbb{B}} \) and \( x' + t \in \overline{\mathbb{B}} \), we get

\[
\Delta v(t)^2 = (u(x_0' + t) - u(x' + t))^2
\]

\[
= \left(u(x_0) - u(x) + O\left(\left(\frac{d}{N}\right)^\alpha\right)\right)^2
\]

\[
= (u(x_0) - u(x))^2 + O\left(\left(\frac{d}{N}\right)^\alpha\right).
\]
Hence
\[
\|v(N_{x,0}) - v(N_{x_0,0})\|_{H,\kappa}^2 = \left( \frac{\left( (u(x_0) - u(x))^2 + O \left( \frac{\alpha}{N^\alpha} \right) \right)}{J_1 + J_2} \right) \frac{J_1}{J_1 + J_2} \frac{R_2}{J_1 + J_2} + \frac{(u(x_0) - u(x))^2 + O \left( \frac{\alpha}{N^\alpha} \right)}{1 + J_2/J_1} + \frac{R_2}{J_1 + J_2}.
\]

(39)

Notice that when \( x_0 + t \in B \) or \( x' + t \in B \), we have \( w_H(x_0 + t)w_H(x' + t) \leq \sup_{z \in B} w_H(z) \) (since \( w_H(z) \leq 1 \) for every \( z \)). Therefore
\[
\frac{J_2}{J_1} \leq \frac{\sum_{t \in N_{0,d}} 1_{\{x_0 + t \in B \text{ or } x' + t \in B\} \kappa(t)}}{\sum_{t \in N_{0,d}} \kappa(t)}.
\]

where \( \lambda = \sup_{z \in B} w_H(z) / \inf_{z \in B} w_H(z)^2 \). According to Theorem 1 of Pruitt [38], for any triangular array of positive weights \( a_{n,j} \geq 0 \), \( j = 1, \ldots, k_n \) and \( n \geq 1 \) such that \( \sum_{j=1}^{k_n} a_{n,j} = 1 \) and independent identically distributed random variables \( X_{n,j} \) with finite mean \( \mu \), we have \( \sum_{j=1}^{k_n} a_{n,j} X_{n,j} \to \mu \) in probability if and only if \( \max_j a_{n,j} \to 0 \). Using this result for the weights \( \kappa(t)/\sum s_{N_{0,d}} \kappa(s) \) and the fact that \( \max_{t \in N_{0,d}} \kappa(t)/\sum s_{N_{0,d}} \kappa(s) \to 0 \) (in fact if in our case the kernel \( \kappa \) is bounded and the sum \( \sum_{t \in N_{0,d}} \kappa(t) \to \infty \) and \( \mathcal{P}\{x_0 + t \in B, x + t \in B\} = (1 - p)^2 \), we have the following convergences in probability:
\[
\frac{\sum_{t \in N_{0,d}} 1_{\{x_0 + t \in B \text{ or } x' + t \in B\} \kappa(t)}}{\sum_{t \in N_{0,d}} \kappa(t)} \to 1 - (1 - p)^2
\]

and
\[
\frac{\sum_{t \in N_{0,d}} 1_{\{x_0 + t \in B, x' + t \in B\} \kappa(t)}}{\sum_{t \in N_{0,d}} \kappa(t)} \to (1 - p)^2,
\]

so that
\[
\frac{\sum_{t \in N_{0,d}} 1_{\{x_0 + t \in B \text{ or } x' + t \in B\} \kappa(t)}}{\sum_{t \in N_{0,d}} 1_{\{x_0 + t \in B, x' + t \in B\} \kappa(t)}} \to \frac{1 - (1 - p)^2}{(1 - p)^2}.
\]

By Hypothesis 1, \( \lambda \to 0 \) in probability, so that
\[
\frac{J_2}{J_1} \to 0
\]
in probability. On the other hand, since the observed image \( v \) is bounded, say by \( V_{\max} \), we have \( \Delta v(t)^2 = (v(x_0 + t) - v(x + t))^2 \leq V_{\max}^2 \) for each \( t \), so that \( R_2 \leq V_{\max}^2 J_2 \). Hence
\[
\frac{R_2}{J_1 + J_2} \leq V_{\max}^2 \frac{J_2}{J_1} \to 0
\]
in probability. Thus, taking into account that \(d/N \to 0\) as \(d \to \infty\) and \(N \to \infty\), from (39) we see that
\[
\|v(N_{x_0,d}) - v(N_{x,d})\|_{H,\kappa}^2 \to (u(x_0) - u(x))^2
\]
in probability, which gives the desired result.

5.2. Proof of Theorem 3.3.

Proof. The proof is similar to that of Theorem 3.2. Fix \(x_0 \in I_0\). Let
\[
\begin{align*}
J'_1 &= \sum_{x \in N_{x_0,D}} 1_{\{x \in B\}} w_{H_1}(x) \kappa_{tr} \left( \frac{\rho_{x_0}(x)}{a} \right), \\
J'_2 &= \sum_{x \in N_{x_0,D}} 1_{\{x \in B\}} w_{H_1}(x) \kappa_{tr} \left( \frac{\rho_{x_0}(x)}{a} \right), \\
R'_1 &= \sum_{x \in N_{x_0,D}} 1_{\{x \in B\}} w_{H_1}(x) \kappa_{tr} \left( \frac{\rho_{x_0}(x)}{a} \right) v(x), \\
R'_2 &= \sum_{x \in N_{x_0,D}} 1_{\{x \in B\}} w_{H_1}(x) \kappa_{tr} \left( \frac{\rho_{x_0}(x)}{a} \right) v(x).
\end{align*}
\]

Then
\[
u_D(x_0) = \frac{\sum_{x \in N_{x_0,D}} w_{H_1}(x) \kappa_{tr} \left( \frac{\rho_{x_0}(x)}{a} \right) v(x)}{\sum_{x \in N_{x_0,D}} w_{H_1}(x) \kappa_{tr} \left( \frac{\rho_{x_0}(x)}{a} \right)} = \frac{R'_1}{J'_1 + J'_2} + \frac{R'_2}{J'_1 + J'_2}.
\]

As in the previous proof, for any \(x \in N_{x_0,D}\) satisfying \(x \in \overline{B}\), we have \(v(x) = u(x) = u(x_0) + O \left( \left( \frac{d}{N} \right)^{\alpha} \right) \). Therefore,
\[
u_D(x_0) = \frac{\left( u(x_0) + O \left( \left( \frac{d}{N} \right)^{\alpha} \right) \right) J'_1}{J'_1 + J'_2} + \frac{R'_2}{J'_1 + J'_2}
\]
\[
= \frac{u(x_0) + O \left( \left( \frac{d}{N} \right)^{\alpha} \right)}{1 + J'_2 / J'_1} + \frac{R'_2}{J'_1 + J'_2}.
\]

Using the fact that \(w_{H_1}(x) \leq \sup_{x \in B} w_{H_1}(z)\) for \(x \in B\), and \(w_{H_1}(x) \geq \inf_{z \in B} w_{H_1}(z)\) for \(x \in \overline{B}\), we have
\[
J'_2 / J'_1 \leq \lambda' \frac{\sum_{x \in N_{x_0,D}} 1_{\{x \in B\}} \kappa_{tr} \left( \frac{\rho_{x_0}(x)}{a} \right)}{\sum_{x \in N_{x_0,D}} 1_{\{x \in B\}} \kappa_{tr} \left( \frac{\rho_{x_0}(x)}{a} \right)},
\]
where \(\lambda' := \sup_{x \in B} w_{H_1}(z) / \inf_{z \in B} w_{H_1}(z)\). By Hypothesis 1, \(\lambda' \to 0\) in probability. Applying Theorem 1 of Pruitt [38] to the numerator and the denominator, and using the fact that the kernel \(\kappa_{tr}\) is bounded and the sum \(\sum_{x \in N_{x_0,D}} \kappa_{tr}(x) \to \infty\), we
see that
\[
\sum_{x \in N_{x_0, D}} 1_{\{x \in B\}} \kappa_{\text{tr}} \left( \frac{\rho(x)}{a} \right) \rightarrow p \quad \frac{1}{1 - p}
\]
in probability. Therefore,
\[
\frac{J'_2}{J'_1} \rightarrow 0
\]
in probability. On the other hand, since \(v(x)\) are bounded by a constant \(V_{\text{max}}\) we have \(R'_2 \leq V_{\text{max}}J'_2\). This implies that
\[
\frac{R'_2}{J'_1 + J'_2} \leq V_{\text{max}} \frac{J'_2}{J'_1} \rightarrow 0
\]
in probability. Hence, from (40) and the condition that \(D/N \to 0\) and \(D \to \infty\), it follows that
\[
u_D(x_0) \to u(x_0)
\]
in probability, which proves the desired result. \(\square\)

6. Conclusions. In this paper, we introduce the notion of Reliable Weight to measure the probability for a pixel to be noise-free, in an image with the presence of a random impulse noise. Combining Reliable Weight with the technique of the Optimal Weights Filter [25, 26], we get an efficient filter for removing random-valued impulse noise. Under suitable assumptions we prove the convergence of the filter. Simulation results show that our filter is competitive, both visually and quantitatively, compared with a number of known filters.

REFERENCES

[1] 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 Transactions on Image Processing, 5 (1996), 1012–1025.
[2] I. Aizenberg, C. Butakoff and D. Paliy, Impulsive noise removal using threshold boolean filtering based on the impulse detecting functions, IEEE Signal Processing Letters 12 (2005), 63–66.
[3] T. Y. Al-Naffouri, A. A. Quaddear and G. Caire, Impulse noise estimation and removal for ofdm systems, IEEE Transactions on communications, 62 (2014), 976–989.
[4] N. Alajlan, M. Kamel and E. Jernigan, Detail preserving impulsive noise removal, Signal Processing: Image Communication, 19 (2004), 993–1003.
[5] A. S. Awad, Standard deviation for obtaining the optimal direction in the removal of impulse noise, IEEE Signal Processing Letters, 18 (2011), 407–410.
[6] E. Beşök and M. Emin Yüksel, Impulsive noise suppression from images with jarque-bera test based median filter, AEU-International Journal of Electronics and Communications, 59 (2005), 105–110.
[7] A. C. Bovik, Handbook of Image and Video Processing, Access Online via Elsevier, 2010.
[8] D. R. K. Brownrigg, The weighted median filter, Communications of the ACM, 27 (1984), 807–818.
[9] R. H. Chan, C.-W. Ho and M. Nikolova, Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization, IEEE Transactions on Image Processing, 14 (2005), 1479–1485.
[10] R. H. Chan, C. Hu and M. Nikolova, An iterative procedure for removing random-valued impulse noise, IEEE Signal Processing Letters, 11 (2004), 921–924.
[11] T. Chen, K.-K. Ma and L.-H. Chen, Tri-state median filter for image denoising, IEEE Transactions on Image Processing, 8 (1999), 1834–1838.
[12] T. Chen and H. R. Wu, Adaptive impulse detection using center-weighted median filters, IEEE Signal Processing Letters, 8 (2001), 1–3.
[13] Qiyu Jin, Li Bai, Ion Grama, Quansheng Liu and Jie Yang, Space variant median filters for the restoration of impulse noise corrupted images, IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing, 48 (2001), 784–789.

[14] V. Crnojevic, V. Senk and Z. Trpovski, Advanced impulse detection based on pixel-wise mad, IEEE Signal Processing Letters, 11 (2004), 589–592.

[15] J. Delon and A. Desolneux, A patch-based approach for removing impulse or mixed gaussian-impulse noise, SIAM Journal on Imaging Sciences, 6 (2013), 1140–1174.

[16] J. Delon, A. Desolneux and T. Guillemot, Parigi: A patch-based approach to remove impulse-gaussian noise from images, Image Processing On Line, 6 (2016), 130–154.

[17] Y. Dong, R. H. Chan and S. Xu, A detection statistic for random-valued impulse noise, IEEE Transactions on Image Processing, 16 (2007), 1112–1120.

[18] R. Garnett, T. Huegerich, C. Chui and W. He, A universal noise removal algorithm with an impulse detector, IEEE Transactions on Image Processing, 14 (2005), 1747–1754.

[19] R. C. Gonzalez and R. E. Woods, Digital Image Processing, Englewood Cliffs, NJ: Prentice-Hall, 2002.

[20] M.-H. Hsieh, F.-C. Cheng, M.-C. Shie and S.-J. Ruan, Fast and efficient median filter for removing 1–99% levels of salt-and-pepper noise in images, Engineering Applications of Artificial Intelligence, 26 (2013), 1333–1338.

[21] H. Hu, B. Li and Q. Liu, Removing mixture of gaussian and impulse noise by patch-based weighted means, Journal of Scientific Computing, 67 (2016), 103–129.

[22] Y. Dong, R. H. Chan and S. Xu, A detection statistic for random-valued impulse noise, IEEE Transactions on Image Processing, 16 (2007), 1112–1120.
[39] F. Russo, Impulse noise cancellation in image data using a two-output nonlinear filter, *Measurement*, 36 (2004), 205–213.

[40] K. S. Srinivasaan and D. Ebenezer, A new fast and efficient decision-based algorithm for removal of high-density impulse noises, *IEEE Signal Processing Letters*, 14 (2007), 189–192.

[41] T. Sun and Y. Neuvo, Detail-preserving median based filters in image processing, *Pattern Recognition Letters*, 15 (1994), 341–347.

[42] C. Wang, T. Chen and Z. Qu, A novel improved median filter for salt-and-pepper noise from highly corrupted images, *Systems and Control in Aeronautics and Astronautics (ISSCAA), 2010 3rd International Symposium on*, IEEE, 2010, 718–722.

[43] S.-S. Wang and C.-H. Wu, A new impulse detection and filtering method for removal of wide range impulse noises, *Pattern Recognition*, 42 (2009), 2194–2202.

[44] L. Wenbin, A new efficient impulse detection algorithm for the removal of impulse noise, *IEEE Transactions on Fundamentals of Electronics, Communications and Computer Sciences*, 88 (2005), 2579–2586.

[45] P. S. Windyga, Fast impulsive noise removal, *IEEE Transactions on Image Processing*, 10 (2001), 173–179.

[46] Z. Zhou, Cognition and removal of impulse noise with uncertainty, *IEEE Transactions on Image Processing*, 21 (2012), 3157–3167.

[47] Z. Zhu, X. Zhang, Q. Wang, A random-valued impulse noise removal algorithm with local deviation index and edge-preserving regularization, *Signal, Image and Video Processing*, 9 (2015), 221–228.