www.ietdl.org

Published in IET Image Processing
Received on 7th July 2011
Revised on 9th May 2012
doi: 10.1049/iet-ipr.2011.0312

ISSN 1751-9659

Improved decision-based detail-preserving variational method for removal of random-valued impulse noise

Y.Y. Zhou Z.F. Ye J.J. Huang

Department of Electronic Engineering and Information Science, Institute of Statistical Signal Processing, University of Science and Technology of China, Hefei, Anhui 230027, People's Republic of China
E-mail: yezf@ustc.edu.cn

Abstract: The authors propose an improved decision-based detail-preserving variational method (DPVM) for removal of random-valued impulse noise. In the denoising scheme, adaptive centre weighted median filter (ACWMF) is first ameliorated by employing the variable window technique to improve its detection ability in highly corrupted images. Based on the improved ACWMF, a fast iteration strategy is used to classify the noise candidates and label them with different noise marks. Then, all the noise candidates are restored one-time by weight-adjustable detail-preserving variational method. The weights between the data-fidelity term and the smooth regularisation term of the convex cost-function in DPVM are decided by the noise marks. After minimisation, the restored image is obtained. Extensive simulation results show that the proposed method outperforms some existing algorithms, both in vision and quantitative measurements. Moreover, our method is faster than some decision-based DPVM. Therefore it can be ported into practical application easily.

1 Introduction

Images are often corrupted by impulse noise which is due to errors generated in noisy sensors and communication channels during image acquisition and transmission [1]. It is necessary to remove the impulse noise to guarantee good performance of subsequent image processes such as edge detection, image segmentation and object tracking. Impulse noise can be categorised into two types based on the noise values. One is the fixed-value impulse noise which is also called 'salt-and-pepper' noise with a large grey level like 255 and a small value 0 [2]. The other is the random-valued impulse noise with a random grey level uniformly distributed in the interval like [0, 255] [3]. It is much easier to remove salt-and-pepper noise because the differences in grey levels between a noisy pixel and its noise-free neighbours are significant most of the times. Some literatures [4–6] have obtained good restoration results in case of salt-and-pepper noise. However, for random-valued impulse noise, the differences are not so great most of the time hence it is much more difficult to remove them. In this paper, we focus on the restoration for random-valued impulse noise.

The median filter was once a very effective approach for random noise removal because of its denoising power and computational efficiency [1]. However, it processes every pixel in the image identically, which spoils a lot of details after denoising. To make matters worse, when the noise level is very high, the median filter cannot work well since there are too many corrupted pixels in the local area. Owing to this, a lot of researchers try to detect noise first and then only process the corrupted pixels while preserving the grey levels of the uncorrupted ones. It is called a decision-based scheme. In the past decades, some effective decision-based random-valued impulse noise removal methods have been proposed such as adaptive centre weighted median filter (ACWMF) [7], noise map-based median filter (NMMF) [8], thresholding noise-free ordered mean filter (TNOMF) [9], directional weighted median filter (DWMF) [10] and so on. ACWMF utilises the centre-weighted median filter [11] that varies centre weights to realise impulse detection by using the differences defined between the outputs of centre-weighted median filter and the current pixel of concern. In NMMF, the impulse detection algorithm uses a weighted filtering technique to obtain the noise map. This process is operated iteratively to locate the final positions of noisy pixels. TNOMF employs the Dempster–Shafer evidence theory to determine whether the pixel is noisy or not. In DWMF, four main directions in the filter window are considered separately. Its impulse detector makes use of the differences between the current pixel and its neighbours aligned with the four directions. There is no doubt that these methods have achieved good performance in impulse detection. However, when it comes to image restoration under noise, only median filter or modified median filter is adopted. Besides, some methods [7–9] fail to detect lots of noisy pixels in high noise levels.

In the late 1990s, a new class of methods, partial differential equation-based methods, had been developed broadly in image processing. In image denoising, the denoised image is the solution of an optimisation problem composed of a data-fidelity term and a smooth regularisation term. Two representative techniques belonging to the variational scheme are total variation

[12–16] and the detail-preserving variational method (DPVM) [17, 18]. Unlike median filter, these filters can preserve more edges or details during minimisation of the cost-function. If the decision-based scheme is employed it will undoubtedly outperform the median-based scheme. Therefore some researchers combine a 'good' noise detector with a variational approach-based noise restorer [19]. We call this method as ACWMF-DPVM for short. It first uses the ACWMF to identify the noisy pixels and then restores these noise candidates using the DPVM. The simulation results are indeed better than those by median-based filters. However, unfortunately, this process needs to be operated iteratively to get the final result. In each iteration, it uses different thresholds in the ACWMF to get noise candidates and then restores them by the DPVM which is a time-consuming minimisation problem. In practice, four iterations are needed so that the whole denoising process is slow. Besides, its denoising capability is limited by the accuracy of the noise detector ACWMF. When the noise level is higher than 40%, some noticeable noise patches are clearly visible in the restored image.

In order to improve the method proposed in [19], we do some work mainly in three aspects. First, the variable window technique is employed in the ACWMF to improve its noise detection ability for highly corrupted images. Second, we give different marks for different classes of noise candidates. The mark indicates the degree to which a noisy pixel is different from its local neighbours in the grey level. Third, instead of carrying out four DPVM in the whole denoising process, we employ DPVM only once to restore all the noise candidates, altering the value of trade-off between the data-fidelity term and the smooth regularisation term according to the noise mark. Besides, we offer a method to estimate the noise level of a corrupted image. Hence, our denoising method is not restricted to the simulated corrupted images alone. The experimental results show that the proposed method has a better noise removal capability than the ACWMF–DPVM and some representative denoising algorithms, especially in highly corrupted images. Moreover, its time complexity is much lower than the ACWMF–DPVM. Hence, our method extends the practical application of the idea of combining a good noise detector with the variational approach-based noise restorer.

The outline of the paper is as follows. In Section 2, the methods ACWMF, DPVM and ACWMF–DPVM are reviewed and analysed. In Section 3, our method is described in four parts. The simulation and results are shown in Section 4. The conclusion is given in Section 5.

2 Reviews and analysis

In this section, the basic methods ACWMF [7] and DPVM [17, 18] are reviewed in the first place. Then, we give an overview of the ACWMF–DPVM proposed in [19]. Meanwhile, some analysis is presented to lead to our idea.

Let $c(i,j)$, for $(i,j) \in \Omega \equiv \{1,\ldots,M\} \times \{1,\ldots,N\}$, be the grey level of a 'clean' M-by-N image $\boldsymbol{c}$ at pixel location (i,j) and $[g_{\min}, g_{\max}]$ (e.g. [0, 255] for an 8-bit image) be the dynamic range of $\boldsymbol{c}$. Denote a noisy image by $\boldsymbol{u}^0$, and in the random-valued impulse noise model, the observed grey level at pixel location (i,j) is given by

$$u^0(i,j) = \begin{cases} g, & \text{with the probability } p \\ c(i,j), & \text{with the probability } 1-p \end{cases}$$

where p defines the image corruption ratio which is also called noise level and g is uniformly distributed in the grey level range $[g_{\min}, g_{\max}]$.

2.1 Adaptive centre weighted median filter

Extensive denoising simulations show that the ACWMF [7] is an excellent method to remove random-valued impulse noise when the noise level is not high. The details of the method are as follows:

Let w denote half of the window size and $L = 2w \times (w+1)$. Therefore $2L+1$ is the total number of pixels in a window. Let

$$y_{ij}^{2k} = \text{median}\{u_{i-m,j-n}^0, (2k) \diamond u_{ij}^0 | -w \le m, n \le w\} \quad (1)$$

Here, $(2k) \diamond u_{ij}^0$ means duplicating the central pixel u_{ij}^0 for $2k$ times and median means the median value of the set. In fact, y_{ij}^0 is equivalent to the output of the standard median filter, whereas y_{ij}^{2k} $(k \ge L)$ has the same value with u_{ij}^0. Hence, k is limited in the range of $[0, L-1]$. The difference d_k is defined as $|y_{ij}^{2k} - u_{ij}^0|$. It can be seen that $d_k \le d_{k-1}$ for $k \ge 1$.

Then a set of thresholds T_ks are employed to distinguish the noisy pixel, where $T_{k-1} \ge T_k$ for $k = 1, 2, \ldots, L-1$. If any one of the inequalities $d_k > T_k$ is true, then u_{ij}^0 is regarded as a noise candidate and replaced by a median of the neighbourhood. Otherwise, it is regarded as an uncorrupted pixel and will not be changed. The choice of T_k is very crucial in this method. Usually, if a 3×3 window is used, w equals 1 and four thresholds T_k are needed. They are as follows

$$T_k = s \cdot \text{MAD} + \delta_k, \quad 0 \le k \le 3 \quad (2)$$

where

$$\text{MAD} = \text{median}\{|u_{i-m,j-n}^0 - y_{ij}^0| : -w \le m, n \le w\} \quad (3)$$

In practice, δ_k is chosen as $[\delta_0, \delta_1, \delta_2, \delta_3] = [40, 25, 10, 5]$ for random-valued impulse noise and s is in the range of $[0, 0.6]$.

When the noise level $p \le 30\%$, the ACWMF is effective. However, with the raising of p, it fails to detect quite a lot of noisy pixels. Fig. 1 shows the performance of noise

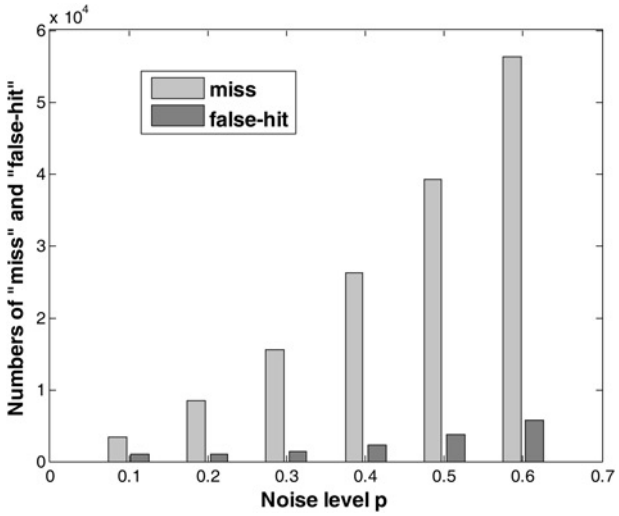

Fig. 1 *Bar graph about the noise detection performance of ACWMF for 'Lena' corrupted at different levels*

detection in the ACWMF for different corrupted 'Lena' images. The horizontal axis represents the noise level p and the vertical axis represents the number of undetected noise or false-hit noise. For simplicity, we denote the number of undetected noisy pixels and false-hit ones by 'miss' and 'false-hit', respectively. From the bars, we can see that there are little changes of 'false-fit' when p increases. However, the un-detection situation becomes sharply worse. If we want to use the ACWMF-based noise detector to obtain a good performance of noise detection, especially for a serious corrupted image, the ACWMF must be ameliorated. In Section 3, we will present the improved method.

2.2 DPVM to restore noisy image

Nikolova [17, 18] first advanced the DPVM to remove outliers and impulse noise. Given the corrupted image $\boldsymbol{u}^0$, the estimate $\hat{u}$ of a clean image is defined as the minimiser of a convex cost-function $F_{u^0}: R^2 \rightarrow R$ which combines a data-fidelity term and a regularisation term, weighted by a parameter $\beta > 0$

$$F_{u^0}(u) = \sum_{(i,j)\in\Omega} |u_{ij} - u^0_{ij}| + \frac{\beta}{2} \sum_{(i,j)\in\Omega} \sum_{(m,n)\in V_{ij}} \varphi(u_{ij} - u_{mn}) \quad (4)$$

where V_{ij} is the set of the four closest neighbours of (i, j), not including (i, j).

The first term $\sum_{(i,j)\in\Omega} |u_{ij} - u^0_{ij}|$ measures the fidelity of the observed data. ℓ_1-norm is designed to better measure impulse noise and outliers. Some literatures [12–14] employ ℓ_2-norm as the data-fidelity term with the assumption that the data are corrupted with the Gaussian noise. However, for impulse noise, ℓ_2-norm is not suitable. Chartrand and Staneva [16] analysed the data-fidelity term of impulse noise as a typical non-Gaussian noise in detail. The result shows that ℓ_1-norm data-fidelity term is best suited to the removal of impulse noise. Nikolova describes it as the non-smooth data-fidelity term. The second term is a smooth regularisation term that represents the local variations in the whole image region Ω. It requires the restored image to be smooth with the edge preserved. $\varphi(t)$ is a smooth and convex edge-preserving potential function. Examples of $\varphi(t)$ are in (5) and (6). β controls the trade-off between the two terms.

$$\varphi(t) = |t|^\alpha,\ 1 < \alpha \leq 2 \quad (5)$$

$$\varphi(t) = \sqrt{\alpha + t^2},\ \alpha > 0 \quad (6)$$

The DPVM provides a very effective framework for processing data contaminated by outliers and impulse noise. Compared with other median-based filters, the crucial advantage of the DPVM is that it considers features in the images such as the possible presence of edges. Some researchers introduce a 'good' noise detector ACWMF into the DPVM. Then, the DPVM is restricted to the noise set. This method is the ACWMF–DPVM proposed in [19].

2.3 Iterative procedure combining the ACWMF with the DPVM

Chan *et al.* [19] proposed a method to combine the ACWMF with the DPVM. Noisy pixels are detected using the ACWMF first and then these pixels are selectively restored by the DPVM. The noise-free pixels are unchanged. Hence, the cost-function is altered as (7)

$$F_{u^0}(u) = \sum_{(i,j)\in N} \left\{ |u_{ij} - u^0_{ij}| + \frac{\beta}{2} \left(\sum_{(m,n)\in V_{ij}\cap N} \varphi(u_{ij} - u_{mn}) + \sum_{(m,n)\in V_{ij}\setminus N} \varphi(u_{ij} - u^0_{mn}) \right) \right\} \quad (7)$$

where N denotes the noise candidates set. $V_{ij}\setminus N$ denotes those neighbours of (i, j) which have been detected uncorrupted.

The method is applied iteratively. At the early iteration, large thresholds in ACWMF are employed to select pixels that are most likely to be noisy. Then, they are restored by the DPVM. In the subsequent iterations, the thresholds are decreased to include more noise candidates. Then, they are also restored by the DPVM. In practice, four iterations are needed. Hence, the algorithm requires implementation of the ACWMF as well as the DPVM four times. The ACWMF can be used very quickly. However, the application of the DPVM is the most time-consuming because it requires minimisation of the functional (7). Therefore this disadvantage limits its application in practice greatly. In Section 3, we offer our strategy to overcome this drawback.

3 Improved decision-based DPVM

To obtain a better noise detection ability, we employ the variable window technique in the ACWMF and name it as improved adaptive center weighted median filter (IACWMF). Based on the method, we use a fast iteration strategy to give different marks for the different classes of noise candidates. Then, the DPVM is applied only once to restore all the selected noise with different weight between the data-fidelity term and the smooth regularisation term according to the noise marks. Besides, a method for estimating the noise level p is presented. In the following, we describe our method in detail.

3.1 Improved adaptive centre weighted median filter

As described in the previous analysis in Section 2, the un-detection situation of the ACWMF becomes worse when the noise level increases. The reason is that there are very few noise-free pixels in the 3×3 filtering window hence many 'bad' pixels destroy the effectiveness of the ACWMF. Therefore when the image is highly corrupted, more pixels should be added to guarantee enough correct local information. In other words, the window size must be enlarged. At the same time, it cannot be too large because the central pixel is similar only to its nearest neighbour. Otherwise, the false-hit will become more serious. Therefore the size of the window is a trade-off between the un-detection and the false-hit. It is crucial in noise detection. Usually, a square window like $(2w + 1) \times (2w + 1)$ is chosen. w denotes half of the window size. Some methods like DWMF [10] use a 5×5 window to enlarge the neighbour. However, in our method, we consider windows with other shapes. Within the scope of a 3×3 square window to a 5×5 one, there are other window types shown in Fig. 2. The best trade-off between the un-detection and the false-hit may be obtained by one

doi: 10.1049/iet-ipr.2011.0312

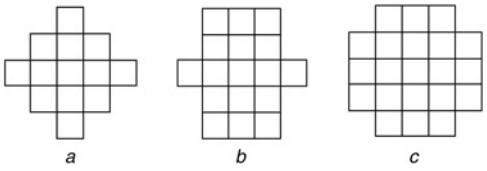

Fig. 2 *Three window types*

a 'Window 1'
b 'Window 2'
c 'Window 3'

of them. Lots of experiments have been conducted to count the total number of undetected noise and false-hit noise using these windows. The results show that the best window type depends on the noise level p (as shown in Table 1).

We describe our IACWMF as follows. When $p \leq 30\%$, the IACWMF is equivalent to the ACWMF. When $p > 30\%$, the calculation of y_{ij}^{2k} and MAD in the ACWMF are changed as (8) and (9), respectively. S is the set of coordinates included in the selected window. For example, if we choose 'window 2' as Fig. 2*b*, S is (10). Hence, the d_ks of every pixel is changed with y_{ij}^{2k} because d_k is defined as $|y_{ij}^{2k} - u_{ij}^0|$. In the fourth part of the section, we will give a simple and effective scheme for estimating p to guide the choice of window type

$$y_{ij}^{2k} = \text{median}\{u_{i-m,j-n}^0, (2k) \diamond u_{ij}^0 | (m, n) \in S\} \quad (8)$$

$$\text{MAD} = \text{median}\{|u_{i-m,j-n}^0 - y_{ij}^0| : (m, n) \in S\} \quad (9)$$

$$\begin{aligned} S = \{&(-2, -1), (-2, 0), (-2, 1), (1, -1), (1, 0), (1, 1) \\ &(0, -2), (0, -1), (0, 0), (0, 1), (0, 2) \\ &(1, -1), (1, 0), (1, 1), (2, -1), (2, 0), (2, 1)\} \end{aligned} \quad (10)$$

3.2 Mark noise using a fast iteration strategy

Random-valued impulse noise owns a random grey level uniformly distributed in the interval like [0, 255]. The differences in grey levels between a noisy pixel and its noise-free neighbours are different. Some are very significant, others are not. Hence, we can make use of the differences to classify the noisy pixels.

Let $NM(i, j)$, for $(i, j) \in \Omega$, be the noise mark of a corrupted M-by-N image $\boldsymbol{u}^0$ at pixel location (i, j) and $NM(i, j) \in \{0, 1/D, 2/D, \ldots, 1\}$. D is a positive integer which denotes the number of noise category. '0' represents $u^0(i, j)$ noise-free, whereas any other value means $u^0(i, j)$ noisy. Closer $NM(i, j)$ is to 1, greater the difference is between $u^0(i, j)$ and its neighbours. Based on the IACWMF and the idea of iteration, we propose a fast strategy to get $NM(i, j)$ for every pixel. At the first iteration, we take large thresholds in the IACWMF to select noisy pixels and render their NMs by '*1*'. Owing to the large thresholds, these noisy pixels are much different from their neighbours. At the same time, they are restored simply by the median of its neighbours to generate a new image 'temp'. In the subsequent iterations, the thresholds are decreased to select more noisy pixels in 'temp'. Their NMs are rendered by '$(q_{\max} - q + 1)/q_{\max}$'. q represents the qth iteration, $q \in \{1, 2, \ldots, q_{\max}\}$. $q_{\max}$ represents the total number of iteration. Hence, we see $q_{\max}$ is equal to D. If the pixels are not selected as noise until the end, their NMs are '0'. We offer the steps of calculating NM as follows:

Step 1: Set $q = 1$ and $\boldsymbol{u}^q = u^0$. Initial $NM(i, j) = 0$ for every $(i, j) \in \Omega$.
Step 2: Set the thresholds $T_k^{(q)}$ as (11) in the IACWMF and employ the IACWMF to process $\boldsymbol{u}^q$. For a selected noisy pixel, if its $NM(i, j)$ is zero, set $NM(i, j) = (q_{\max} - q + 1)/q_{\max}$. At the same time, $\boldsymbol{u}^q$ is renewed by the IACWMF to generate a 'temp' image.
Step 3: Set $\boldsymbol{u}^{q+1}$ = 'temp'.
Step 4: If $q < q_{\max}$, set $q = q + 1$ and go back to step 2. Otherwise, NM is the final noise mark matrix for the corrupted image $\boldsymbol{u}^0$

$$T_k^{(q)} = s \cdot \text{MAD} + \delta_k + 20 \cdot (q_{\max} - q), \quad 0 \leq k \leq 3 \quad (11)$$

In practice, $q_{\max}$ is set by four. So, $NM(i, j) \in \{0, 1/4, 2/4, 3/4, 1\}$. All the pixels with non-zero noise marks are included in the noise candidates set N and other pixels are included in the noise-free candidates set N^c. Although iterative, this marking procedure is not very time-consuming because each implementation of the IACWMF is fast.

3.3 Weight-adjustable DPVM to restore noisy image

After getting different marks for the different classes of noise candidates, we apply the DPVM only once to restore all the selected noise candidates in N with adjustable weight according to the noise marks. The noise-free pixels in N^c are unchanged.

Similar to the description in Section 2.2, the second term in (4) plays the role of edge-preserving smoothing. β is a weight that controls the trade-off between the two terms. If a noisy candidate is very different with its neighbours in the grey level, it is reasonable to increase the weight of the smooth term to make smooth stronger when restoring. Since the noise mark $NM(i, j)$ represents the degree of difference between the pixel in (i, j) and its neighbours, we can define an adjustable weight β_e as (12). Through the weight-adjustable DPVM, we do not need to perform DPVM four times such as in [19] when restoring different classes of noise candidates. This speeds up the whole denoising algorithm evidently.

$$\beta_e = \beta_0 \cdot f(NM(i, j)) \quad (12)$$

Here, β_0 is the upper bound of weight added to the smooth term. It is set to a fixed value. f is a monotone increasing function with the noise mark as its variable. Larger $NM(i, j)$ is, greater weight is added to the smooth term. We choose a linear function $f(t) = t$ owing to its simplicity and effectiveness. Hence, β_e can be calculated as (13).

$$\beta_e = \beta_0 \cdot NM(i, j) \quad (13)$$

Table 1 Best choices of window types

Noise level	Window type
$p \leq 30\%$	3×3 square window
$30\% < p < 50\%$	window 1 as Fig. 2*a*
$p \geq 50\%$	window 2 as Fig. 2*b*

Then, the cost-function in (7) is altered as (14)

$$F_{u^0}(u) = \sum_{(i,j)\in N} \left\{ |u_{ij} - u^0_{ij}| + \frac{\beta_0 \cdot NM(i,j)}{2} \cdot \left(\sum_{(m,n)\in V_{ij}\cap N} \varphi(u_{ij} - u_{mn}) + \sum_{(m,n)\in V_{i,j}\setminus N} \varphi(u_{ij} - u^0_{mn}) \right) \right\} \quad (14)$$

The choice of algorithm to minimise the functional $F_{u^0}(u)$ is crucial to speed. In order to operate as fast as possible, we employ Newton's method with a favourable initial guess to guarantee convergence (see [20] for more details). The convergence rate of the minimisation scheme is very fast. Then, we obtain the restored image $\hat{u}$.

3.4 Estimation of the noise level p

For the corrupted images in the simulation, we know the precise value of the noise level p before denoising. However, in practical application, p is unknown. In our method, we need to choose the best window type depending on p. Hence, here, we offer a possible method to estimate p. Note that we only need to estimate a rough p, not a precise value. The method is as follows: extract a sub-image which is nearly a homogeneous region from the noisy image (see Fig. 3). A homogeneous region means that there is not much grey variation in it. Usually, there are always some small regions which are smoothly varying. Hence, this extraction should not be much difficult based on vision.

Denote the sub-image by 'SI'. The size of the sub-image can be controlled by the users or be specified as a 15×15 square window to include a certain number of pixels in a homogeneous region. The number of pixels in SI should be enough for estimation. We then calculate the mean grey value m_1 and variance m_2 of SI. There are two sets in SI, SI1 and SI2. SI1 includes the corrupted pixels, whereas SI2 includes the clean ones. Owing to near uniformity in the sub-image, we can assume the pixels in SI2 own the same grey value SI2c. The total pixel number of SI is denoted by SN. Hence, two equations can be derived

$$m_1 \simeq p \cdot \text{mean}\{\text{SI1}\} + (1-p) \cdot \text{mean}\{\text{SI2}\} \quad (15)$$

$$m_2 \simeq \frac{\sum_{i=1}^{\text{SN}\cdot p} (\text{SI1}_i - m_1)^2 + \sum_{j=1}^{\text{SN}\cdot(1-p)} (\text{SI2}_j - m_1)^2}{\text{SN}} \quad (16)$$

Here, mean$\{\cdot\}$ denotes the mean grey level of one set. SI1$_i$ is the grey level of the ith pixel in SI1 and SI2$_j$ is that of the jth pixel in SI2. For (16), we make some changes as follows

$$m_2 \simeq \left(\sum_{i=1}^{\text{SN}\cdot p} ((\text{SI1}_i - \text{mean}\{\text{SI1}\}) + (\text{mean}\{\text{SI1}\} - m_1))^2 \right) ()/\text{SN} + (1-p)(\text{SI2}c - m_1)^2$$

Then we derive a simple formula as (17)

$$m_2 \simeq p \cdot \text{var}\{\text{SI1}\} + p \cdot (\text{mean}\{\text{SI1}\} - m_1)^2 + (1-p) \cdot (\text{SI2}c - m_1)^2 \quad (17)$$

var$\{$SI1$\}$ denotes the variance of grey level in SI1. Note that the pixels in SI1 are approximately subject to a random distribution with the grey level uniformly distributed in the interval [0, 255]. Hence, mean$\{$SI1$\}$ and var$\{$SI1$\}$ are $\sim$255/2 and 255^2/12, respectively.

Solve (15) and (17) simultaneously and p is obtained. The operation can be carried out on several sub-images and the average value is a rough estimation of p. We have conducted some tests and the results are substantially correct.

Next, we summarise our whole random-valued impulse noise removal algorithm as follows:

- Estimate the noise level p of a corrupted image and choose the best window type according to Table 1 for noise detection.
- Based on the IACWMF and the iteration scheme, the noise candidates are selected and rendered with different noise marks.
- Restore the selected noise candidates one-time by a weight-adjustable DPVM and get the final restored image.

Our algorithm is fast enough for being ported into practical applications. In Section 4, we will show its low time complexity.

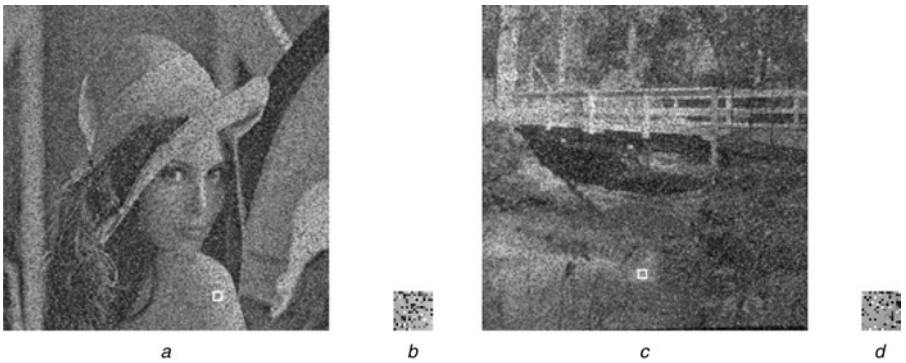

Fig. 3 *Extract a homogeneous region from the corrupted image*

a Corrupted 'Lena' image with 40% random-valued impulse noise
b Extracted region from *a*
c Corrupted 'Bridge' image with 40% random-valued impulse noise
d Extracted region from *c*

doi: 10.1049/iet-ipr.2011.0312

Table 2 PSNRs (MSSIMs) results of restored images with $p = 10$ and 30% random-valued impulse noise

Method	$p = 10\%$				$p = 30\%$			
	'Lena'	'Barbara'	'Bridge'	'Boat'	'Lena'	'Barbara'	'Bridge'	'Boat'
noisy image	19.19(0.3147)	18.84(0.4188)	18.83(0.5049)	19.27(0.3842)	14.46(0.1150)	14.05(0.1899)	14.03(0.2251)	14.56(0.1562)
SMF	33.81(0.9108)	24.91(0.7953)	26.01(0.7712)	29.84(0.8360)	28.08(0.7938)	22.91(0.6691)	23.27(0.6561)	26.18(0.7296)
ACWMF	37.53(0.9744)	26.94(0.9183)	28.70(0.9110)	32.70(0.9520)	27.99(0.7989)	23.41(0.7326)	23.88(0.7498)	26.47(0.7851)
NMMF	37.26(0.9723)	27.19(0.9224)	28.03(0.9014)	32.11(0.9495)	31.86(0.9045)	24.41(0.8138)	25.03(0.7944)	28.46(0.8613)
DWMF	36.91(0.9586)	26.14(0.8913)	27.91(0.8994)	32.49(0.9481)	32.67(0.9182)	24.11(0.7910)	24.73(0.7500)	28.30(0.8707)
ACWMF-DPVM	38.64(0.9794)	27.11(0.9225)	28.98(0.9131)	33.03(0.9552)	33.11(0.9285)	24.89(0.8452)	26.15(0.8315)	29.53(0.8881)
our method	38.66(0.9797)	27.32(0.9261)	29.25(0.9186)	33.26(0.9573)	33.22(0.9316)	24.96(0.8474)	26.21(0.8336)	29.60(0.8906)

Table 3 PSNRs (MSSIMs) results of restored images with $p = 50\%$ and 60% random-valued impulse noise

Method	$p = 50\%$				$p = 60\%$			
	'Lena'	'Barbara'	'Bridge'	'Boat'	'Lena'	'Barbara'	'Bridge'	'Boat'
noisy image	12.27(0.0622)	11.87(0.1057)	11.83(0.1209)	12.38(0.0866)	11.46(0.0463)	11.08(0.0782)	11.04(0.0892)	11.53(0.0614)
SMF	21.67(0.4922)	19.21(0.4060)	19.24(0.4432)	21.01(0.4830)	18.91(0.3321)	17.16(0.2892)	17.20(0.3253)	18.66(0.3474)
ACWMF	20.92(0.4388)	18.85(0.4116)	18.88(0.4682)	20.43(0.4664)	18.16(0.2805)	16.64(0.2784)	16.70(0.3305)	17.98(0.3183)
NMMF	26.78(0.7635)	22.02(0.6316)	21.77(0.6073)	24.51(0.6982)	23.04(0.6118)	19.81(0.4841)	19.63(0.4766)	21.76(0.5641)
DWMF	29.18(0.8395)	22.91(0.6924)	22.95(0.6503)	25.86(0.7517)	26.35(0.7412)	21.70(0.5971)	21.37(0.5486)	23.91(0.6660)
ACWMF-DPVM	27.27(0.7662)	22.53(0.6823)	22.82(0.6745)	25.28(0.7231)	23.45(0.5953)	20.37(0.5298)	20.44(0.5399)	22.33(0.5751)
our method	30.01(0.8818)	23.30(0.7261)	23.26(0.6920)	26.37(0.7869)	27.59(0.8198)	22.38(0.6544)	21.87(0.5721)	24.64(0.7134)

4 Simulation and results

In our experiments, four images are chosen as the test images corrupted by different levels of random-valued impulse noise. They are 'Lena', 'Barbara', 'Bridge' and 'Boat'. Every image is an 8-bit grey level image with 512 × 512 size. A range of noise levels varying from 10 to 60% with increments of 10% are tested. The peak signal-to-noise ratio (PSNR) [1] between the restored image and the original image is first selected as the performance index. The restored results are required to have high PSNRs. It is defined as follows

$$\mathrm{PSNR} = 10\log_{10}\frac{255^2}{(1/MN)\sum_{i,j}(\hat{u}_{ij} - c_{ij})^2} \tag{18}$$

where MN is the image size; $\hat{u}_{ij}$ and c_{ij} denote the pixel values of the restored image and the original image, respectively.

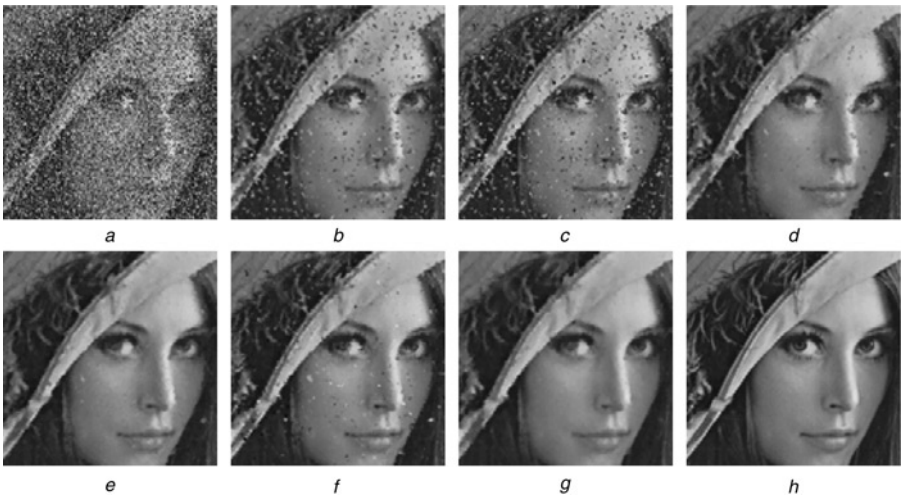

Fig. 4 *'Lena' restoration results of different filters*

a Corrupted Lena image with 50% random-valued impulse noise
b Restored image by SMF
c Restored image by ACWMF
d Restored image by NMMF
e Restored image by DWMF
f Restored image by ACWMF-DPVM
g Restored image by our method
h Original image

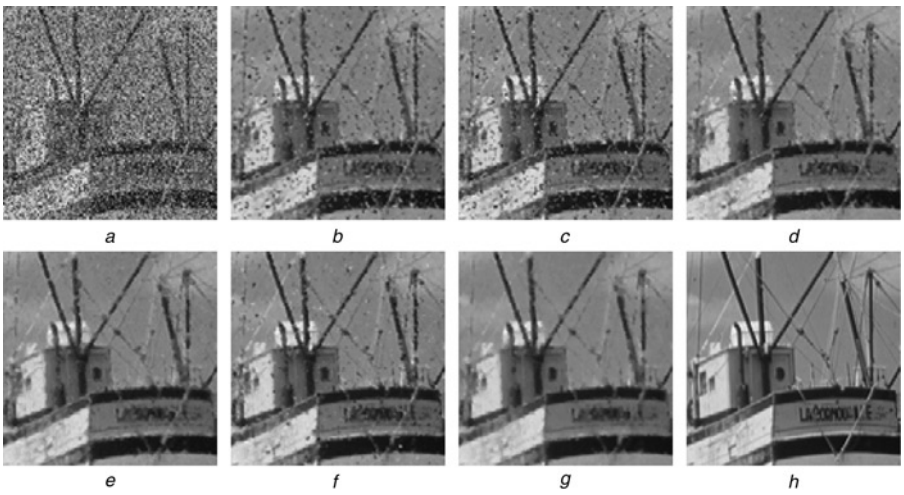

Fig. 5 *'Boat' restoration results of different filters*

a Corrupted boat image with 50% random-valued impulse noise
b Restored image by SMF
c Restored image by ACWMF
d Restored image by NMMF
e Restored image by DWMF
f Restored image by ACWMF-DPVM
g Restored image by our method
h Original image

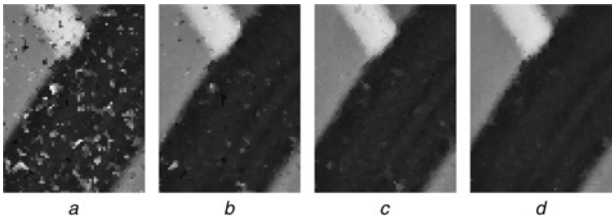

Fig. 6 *Mirror frame in the restored Lena images of 50% corrupted ratio*

a Restored image by ACWMF
b Restored image by ACWMF-DPVM
c Restored image by DWMF
d Restored image by our method

PSNR is based on pixel-wise signal differences, which ignores the underlying signal structure. Therefore we choose another index named structure similarity index measure (SSIM) proposed in [21] to compare the local patterns of pixel intensities between two images. SSIM is calculated within the local window and every pixel in $\hat{u}$ owns its SSIM value. In practice, a mean SSIM (MSSIM) index is often used to evaluate the overall image quality. If $\hat{u}$ is more similar to $\boldsymbol{c}$, MSSIM is closer to 1.

In order to obtain the best denoised images, we tune some parameters to the best values. s is set to 0.3 in the IACWMF. The edge-preserving potential function is chosen as $\varphi(t) = |t|^{\alpha}$ and α is set to 1.3. β_0 is set to 2 as the upper bound of the weight. Then, the adjustable weight is $\beta_e \in \{\beta_0/4, 2\beta_0/4, 3\beta_0/4, \beta_0\}$.

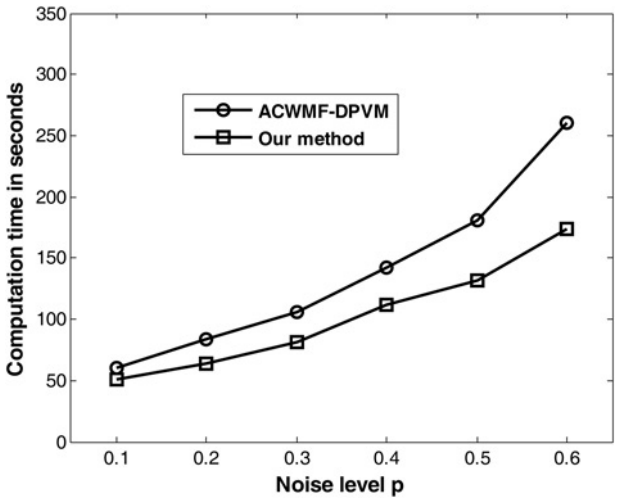

Fig. 7 *Computation time of restoring 10– 60% corrupted 'Lena' using ACWMF–DPVM and our method*

Table 4 Comparison of detection results for the image 'Lena' corrupted by random-valued impulse noise

Method	40%			50%			60%		
	Undetected	False-hit	Total error	Undetected	False-hit	Total error	Undetected	False-hit	Total error
ACWMF	26 194	2202	28 396	39 210	3668	42 878	56 189	5772	61 961
NMMF	15 734	9838	25 572	19 165	16 356	35 521	23 936	22 275	46 211
DWMF	14 177	6188	20 365	14 658	9113	23 771	18 875	9638	28 513
ACWMF-DPVM	16 408	1699	18 107	24 053	1942	25 995	36 250	2584	38 834
our method	10 912	5384	16 296	12 082	8870	20 952	16 361	9034	25 395

doi: 10.1049/iet-ipr.2011.0312

4.1 Denoising performance

To test the noise removal capability of our proposed method, some other random-valued impulse noise filters are also tested for comparison purpose. They are standard median filter (SMF) [2], ACWMF [7], noise map-based median filter (NMMF) [8], directional weighted median filter (DWMF) [10] and ACWMF–DPVM [19]. All the parameters in these methods are set as the values recommended in the original paper.

Tables 2 and 3 list the PSNR and MSSIM values of restored images with random-valued impulse noise level $p = 10$, 30, 50 and 60% for 'Lena', 'Barbara', 'Bridge' and 'Boat,' respectively. The bold numbers are the best values in corresponding columns. Clearly, our proposed method achieves the highest PSNRs and MSSIMs. We present the restored images of the 50% corrupted 'Lena' and 'Boat' in Figs. 4 and 5. From them, we see that in the case of a high noise level like 50%, there are apparent noise patches in the restored images of 'SMF', 'ACWMF', 'NMMF' and

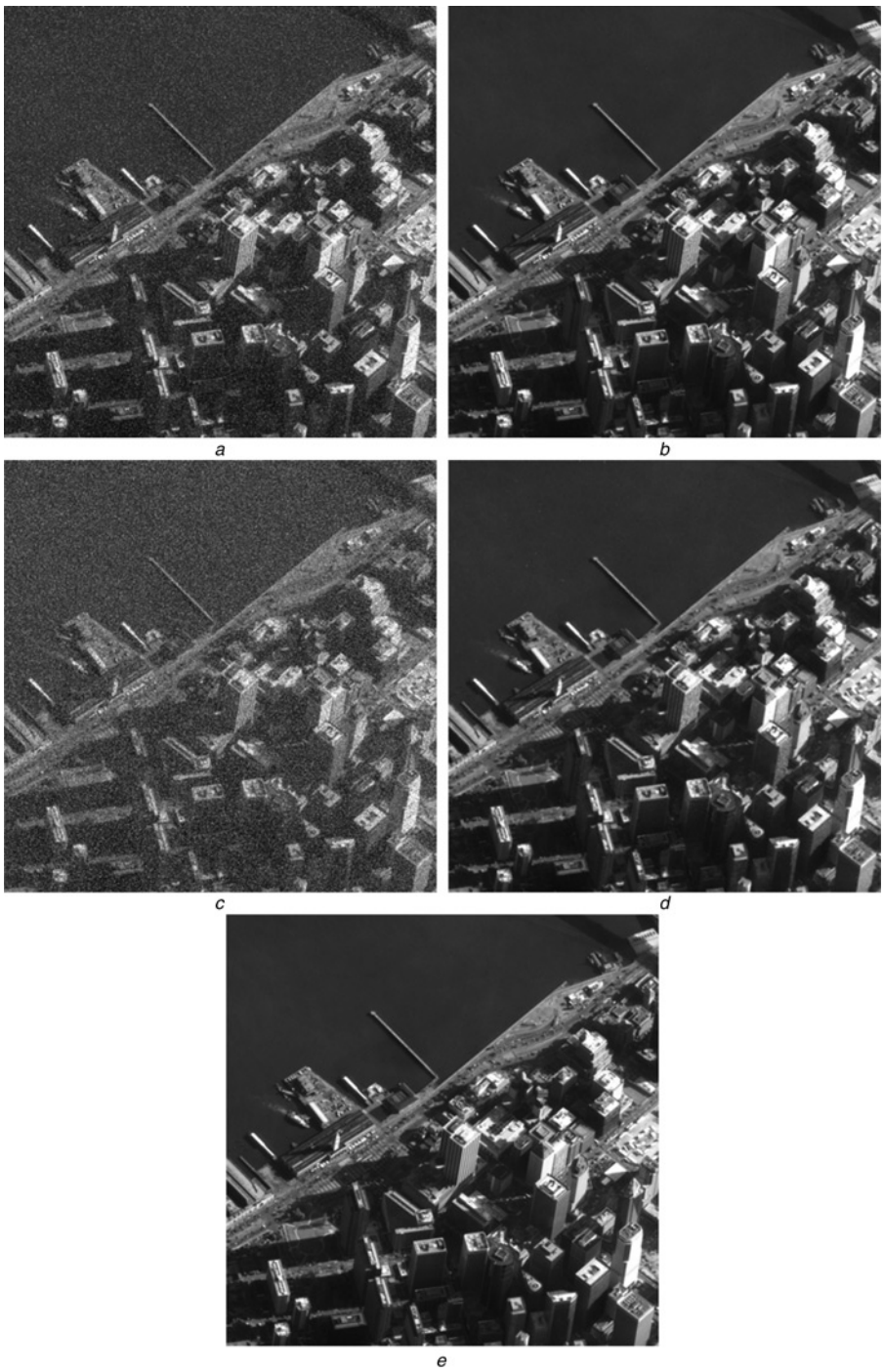

Fig. 8 *Potential application of our algorithm in the satellite image restoration*

a 20% corrupted satellite image 'San-francisco'
b Restored image of *a* by our algorithm
c 40% corrupted satellite image 'San-francisco'
d Restored image of *c* by our algorithm
e Original satellite image 'San-francisco'

'ACWMF–DPVM'. The reason is that these methods use the fixed square window with 3×3 size to provide a neighbourhood, which inevitably leads to quite a lot of noisy pixels being undetected. Although this disadvantage is overcome to some extent by the 'DWMF' and our method, that all have a larger window size; however, the DWMF uses a 5×5 window for all noise levels so that it causes a little bit more blur. The ACWMF–DPVM preserves details well but the undetected pixels seriously destroy the visual performance of the restored images. Our method not only removes lots of noise, but also preserves more details. In Fig. 6, the details of the mirror frame in the restored images of 50% corrupted 'Lena' are magnified. We find that our method obtains a best compromise between noise-removal and detail-preservation compared with the ACWMF, ACWMF–DPVM and DWMF.

4.2 Noise detection performance

In addition to comparing the denoising performances, the capability of noise detection is also compared. In Table 4, we list the number of undetected noisy pixels, false-hit pixels and total error-detected pixels about the ACWMF, NMMF, DWMF, ACWMF–DPVM and our method in 40–60% corrupted situations. Since we employ the variable window technique to detect noise, the number of undetected noisy pixels of our method is the smallest. Although the DWMF uses a larger window, the un-detection is more serious. It means that the ACWMF-based noise detector is better than the direction-based one. Of course, the false-hit numbers are a little bit larger than that of the ACWMF–DPVM, but the number of total error-detected pixels of our method is smallest. It means that our algorithm obtains the best trade-off between the un-detection and the false-hit. The variable window technique is very successful in noise detection.

4.3 Computational complexity

Our method improves the ACWMF–DPVM in denoising performance and time performance. Here, we compare the CPU time of restoring 10–60% corrupted 'Lena' in seconds using the two methods in the platform of MATLAB 2009. The computer is equipped with a 2.80-GHZ Intel(R) core(TM) i5 CPU and a 4.0-GB memory. The result is shown in Fig. 7. The ACWMF–DPVM is obviously slower than our method. Moreover, with the increase of the noise level p, our method is much faster than the ACWMF–DPVM. The reason is that the ACWMF–DPVM requires four applications of the ACWMF and the DPVM, which is very time consuming. However, in our method, the DPVM is employed only once to restore all the selected noise candidates. Hence, a big advantage of our method over ACWMF–DPVM is low time complexity.

4.4 A potential application of our work in satellite image restoration

Satellite images usually suffer from a series of degradation processes in the imaging and transmission chain including atmospheric turbulence, optical aberration, transmission errors and so on. In particular, in the long-distance communication channels, there may be various adverse environmental factors to threaten the quality of satellite images. A certain amount of impulse noise tends to contaminate images. If the users obtain a noisy satellite image, it is very necessary to retrieve the wrong information in some manner. Otherwise, the subsequent processes for the image will be destroyed to some extent. Our restoration algorithm can be applied in coping with this situation. We simulate a degradation process which may happen in the transmission of satellite images. The noise ratio is considered in the range of 15–45% and the noise type is chosen as random-valued impulse noise. Some typical satellite images are selected [These satellite images can be downloaded from the website http://www.satimagingcorp.com/.] and degraded by the corruption process. Then, we employ our proposed algorithm to restore them and an example is presented in Fig. 8. From the five images, we see that the restored images are very close to the original one. The noise is almost completely removed and the details are mostly well preserved. Therefore our proposed algorithm has a potential application in satellite image restoration.

5 Conclusions

In this paper, we propose an improved decision-based DPVM for removal of random-valued impulse noise. The ability of noise removal and the restoring speed are the main improvements in our algorithm. We choose the more suitable window type rather than the fixed square window to detect noise. This modification makes the number of undetected noisy pixels decrease greatly. Hence, it offers a good foundation for noise restoration. Considering the distribution of random-valued impulse noise and the difference between a noisy pixel and its neighbour, we classify noise candidates and mark them with different values using a fast iteration scheme based on the IACWMF. Then, these noise marks guide the restoration algorithm DPVM to change the weight adaptively. More importantly, the weight-adjustable DPVM is implemented only once to restore all the noise candidates. Such a strategy makes the whole restoration faster than the ACWMF–DPVM. Besides, the restored images are better in terms of PSNR, MSSIM and visual performance than those obtained by some existing methods. In view of the advantages of speed and restoration performance, our method can be ported into practical application.

6 Acknowledgments

The authors thank the anonymous reviewers for their valuable comments, which help to improve the quality of this paper.

7 References

1 Bovik, A.: 'Handbook of image and video processing' (Academic Press, New York, 2000)
2 Gonzalez, R.C., Woods, R.E.: 'Digital Image Processing' (Addison Wesley, New York, 1992)
3 Chen, T., Wu, H.R.: 'Space variant median filters for the restoration of impulse noise corrupted images', *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, 2001, **48**, (8), pp. 784–789
4 Hwang, H., Haddad, R.A.: 'Adaptive median filters: new algorithms and results', *IEEE Trans. Image Process*, 1995, **4**, (4), pp. 499–502
5 Wang, Z., Zhang, D.: 'Progressive switching median filter for the removal of impulse noise from highly corrupted images', *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, 1999, **46**, (1), pp. 78–80
6 Chan, R.H., Ho, C.-W., Nikolova, M.: 'Salt-and-Pepper noise removal by median-type noise detectors and detail-preserving regularization', *IEEE Trans. Image Process.*, 2005, **14**, (10), pp. 1479–1485
7 Chen, T., Ren Wu, H.: 'Adaptive impulse detection using center-weighted median filters', *IEEE Signal Process. Lett.*, 2001, **8**, (1), pp. 1–3

doi: 10.1049/iet-ipr.2011.0312

8 Luo, W.: 'A new efficient impulse detection algorithm for the removal of impulse noise', *IEICE Trans. Fundam. Electron. Commun. Comput.*, 2005, **E88-A**, (10), pp. 2579–2586

9 Lin, T.-C., Yu, P.-T.: 'Thresholding noise-free ordered mean filter based on Dempster–Shafer theory for image restoration', *IEEE Trans. Circuits Syst.-I, Regul. Pap.*, 2006, **53**, (5), pp. 1057–1064

10 Dong, Y., Xu, S.: 'A new directional weighted median filter for removal of random-valued impulse noise', *IEEE Signal Process Lett.*, 2007, **14**, (3), pp. 193–196

11 Ko, S.-J., Lee, Y.-H.: 'Center weighted median filters and their applications to image enhancement', *IEEE Trans. Circuits Syst.*, 1991, **38**, pp. 984–993

12 Rudin, L.I., Osher, S., Fatemi, E.: 'Nonlinear total variation based noise removal algorithms', *Physica D*, 1992, **60**, pp. 259–268

13 Chambolle, A., Lions, P.-L.: 'Image recovery via total variation minimization and related problems', *Numer. Math.*, 1997, **76**, pp. 167–188

14 Chan, T.-F., Osher, S., Shen, J.: 'The digital TV filter and nonlinear denoising', *IEEE Trans. Image Process.*, 2001, **10**, (2), pp. 231–241

15 Chan, T.F., Shen, J.: 'Mathematical models for local nontexture inpaintings', *SIAM J. Appl. Math*, 2002, **62**, (3), pp. 1019–1043

16 Chartrand, R., Staneva, V.: 'Total variation regularisation of images corrupted by non-Gaussian noise using a quasi-Newton method', *IET Image Process.*, 2008, **2**, pp. 295–303

17 Nikolova, M.: 'Minimizers of cost-functions involving nonsmooth data-fidelity terms. Application to the processing of outliers', *SIAM J. Numer. Anal.*, 2002, **40**, pp. 965–994

18 Nikolova, M.: 'A variational approach to remove outliers and impulse noise', *J. Math. Imaging Vis.*, 2004, **20**, pp. 99–120

19 Chan, R.H., Hu, C., Nikolova, M.: 'An iterative procedure for removing random-valued impulse noise', *IEEE Signal Process. Lett.*, 2004, **11**, (12), pp. 921–924

20 Chan, R.-H., Ho, C.-w.: 'Convergence of Newton's method for a minimization problem in impulse noise removal', *J. Comput. Math.*, 2004, **22**, (2), pp. 168–177

21 Wang, Z., Conrad Bovik, A., Sheikh, H.R., Simoncelli, E.P.: 'Image quality assessment: from error visibility to structural similarity', *IEEE Trans. Image Process.*, 2004, **13**, (4), pp. 600–612