International Journal of Innovative
Computing, Information and Control
Volume **11**, Number **5**, October **2015**

ICIC International ©2015 ISSN 1349-4198
pp. 1639–1653

A MULTI-CLASS DEMPSTER CLASSIFIER WITH SUPPORT VECTOR MACHINE FOR IMAGE ENHANCEMENT

Tzu-Chao Lin

Department of Computer Science and Information Engineering
WuFeng Institute of Technology
Chiayi 62107, Taiwan
tclin@wfu.edu.tw

Received March 2015; revised June 2015

Abstract. *In this paper, a novel approach for image noise removal is proposed. The proposed technique employs a switching scheme based on a noise detection mechanism that uses support vector machines (SVMs). The SVM noise detector determines whether a pixel is contaminated by noise. The fusion of sources of evidence based on the Dempster-Shafer evidence theory is used to determine to what extent noise is considered. An efficient step-like function is used to partition the belief space, and then a multi-class noise classifier is employed. The least mean squares learning algorithm is applied to obtain the optimal weight for the adaptive lower-upper-middle filter. To improve the final filtering performance, rank-ordered mean filtering is also adopted. Experimental results show the effectiveness of the proposed filter in terms of noise suppression and perceived image quality.*

Keywords: Support vector machine, Evidence theory, Impulse noise, Least mean square

1. **Introduction.** Noisy sensors, inappropriate shutter speed, camera mis-focus, and atmospheric turbulence may introduce impulse noise to images, degrading image quality. Therefore, the effective removal of noise without degrading quality (i.e., preserving image features) is a major step in computer vision and image processing [1-6]. Many noise removal methods based on order statistical filters have been proposed. The basic vector directional filter (BVDF) [7], vector median filter (VMF) [8], and directional distance filter (DDF) [9], which replace the center pixel with the median among all pixels in the window, are well-known classical nonlinear filters. These filters are commonly used for suppressing impulse noise due to their simple implementation and efficiency. However, these filters cause smoothing, which blurs fine image details and affects texture even at low noise density.

The center-weighted median (CWM) filter, weighted median (WM) filter, and adaptive center-weighted median (ACWM) filter, which give more emphasis to the center pixel in the filtering window, and achieve good smoothing for slightly corrupted images [10-12]. These filters improve the disadvantages of typical median filters but still blur some details and often damage edges [16]. To balance noise attenuation and image detail preservation, Lin proposed a neural-based CWM filter with an adjustable center weight [13]. In addition, Arakawa *et al.* proposed median-type filters based on the partition concept and fuzzy rules [14,15]. Although these adaptive schemes give satisfactory filtering results, their performance depends on the training reference image.

The above-mentioned filters make no distinction between noisy and noise-free pixels, causing valuable information in the image to be lost since spatially invariant operators are applied. In order to avoid the removal of good pixels, a switching strategy that detects

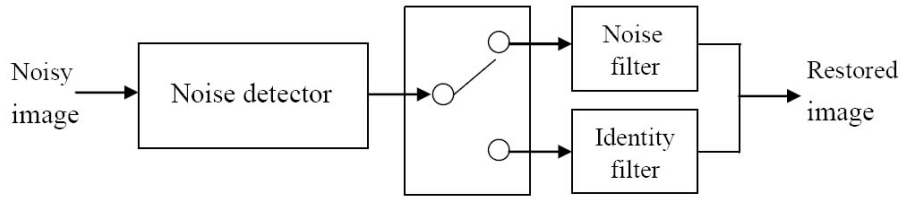

FIGURE 1. The switching-based filter with a noise detector

noisy pixels can be used prior to filtering, as shown in Figure 1. If the center pixel is identified by the noise detector as a corrupted pixel, its restored value is the output of the filtering mechanism. Otherwise, it remains unchanged. The switching median filter (SWM-I and SWM-II) is the most straightforward switching-based filter [16]. The fast peer group filter (FPGF) and adaptive vector median filter (AVMF) are used for impulse noise removal by adaptive switching schemes [17-19]. The signal-dependent rank-ordered mean (SD-ROM) filter is a switching filter that is conditional on the difference between the input pixel and remaining rank-ordered pixels in the sliding window [20]. The tri-state median (TSM) filter [21] is composed of the MED filter, identity filter, and CWM filter. Depending on the threshold value controlling the switching logic, the input pixel value is replaced by the output of either the identity filter, MED filter, or CWM filter. Another switching scheme used by an impulse detector is based on rank-ordered absolute difference (ROAD) statistics [22]. A high ROAD factor implies a noisy pixel. Vectorial lower-upper-middle (LUM) filters control the impulse detection process [23,43]. These switching-based filters, which rely on a proper threshold and are based on order statistics, have been proven to efficiently suppress impulse noise while preserving image details. Pankaj *et al.* proposed an improved adaptive impulse noise suppression filter based on a neuro-fuzzy network [24,44-46]. An artificial neural network is used in the detection stage. Unfortunately, the training strategy may work well at a pre-assumed noise density level but poorly at outside training. The main drawback of this network is thus the selection of training images.

These switching-based filters can well preserve details if the accuracy of noise detection is sufficient [47,48]. Recently, support vector machines (SVM) with a high generalization capability without a priori knowledge have been applied to image filtering [42]. Lin and Yu proposed an adaptive two-pass median (ATM) filter and Liu *et al.* proposed an SVM-EPR filter for impulse noise removal [25,26]. An SVM impulse detector is used to identify whether an input pixel is noisy. SVM-based filters can achieve satisfactory results, but they sometimes misdetect uncorrupted pixels as noisy or leave noisy pixels undetected.

To overcome the above-mentioned drawbacks, the present study proposes a novel multi-class LUM (MCL) filter based on a switching scheme that uses SVMs and Dempster-Shafer (D-S) evidence theory. The proposed filter improves the performance of median-based filters, preserving image details while effectively suppressing noise. The proposed filter comprises an efficient SVM noise detector and an MCL filter. According to the belief obtained using Dempster's combination rule, the proposed MCL filter partitions the space of belief using a simple step-like function to establish an MCL filter. The optimal weight of each class for the adaptive LUM filter is derived using a least mean squares (LMS) learning algorithm. To improve the final filtering performance, ROM filtering is adopted. Experimental results show that the proposed filter substantially outperforms many existing filters in terms of image restoration.

The rest of this paper is organized as follows. Section 2 covers the basic principles of SVMs and D-S evidence theory. The design of the proposed MCL filter is presented in

Section 3. In Section 4, experimental results are provided to demonstrate the performance of the proposed scheme. Finally, the conclusion is given in Section 5.

2. Basic Principles.

2.1. **Support vector machines.** SVM is a set of related supervised learning methods based on the statistical learning theory and the structural risk minimization principal. The original idea of the binary SVM is to use a linear separating hyperplane to maximize the distance between two classes [40].

The decision function of the binary SVM is defined as:

$$f(x) = W^T X + c \tag{1}$$

where $W = [w_1, w_2, \ldots, w_k]^T$ is the weight vector and c is a scalar. The decision function is obtained by solving the optimization problem.

$$\begin{aligned} &\text{Minimize } P(w,\xi) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}\xi_i, \\ &\text{Subject to } y_i\left[w^T x_i + b\right] \geq 1 - \xi_i, \quad y_i \in \{-1, 1\},\ \xi \geq 0,\ i = 1, 2, \ldots, n, \end{aligned} \tag{2}$$

where parameter C is a trade-off between generalization and accuracy. The solution to the problem is obtained by solving the corresponding Wolfe dual problem, a quadratic programming problem with n variables [27-29]. More details description can be found in [24]. The present work uses the LIBSVM package developed by Chang and Lin [30].

2.2. **Dempster-Shafer evidence theory.** The D-S evidence theory is based on Dempster's work, as extended by Shafer [31,32]. The concept is to decrease uncertainty in information fusion by means of a combination rule applied to evidence sources. Assume that D is a variable whose domain is a finite set X. The power set of X, 2^X, is called a frame of discernment. A D-S belief structure is associated with a mapping m, called the basic assignment function, $m : 2^X \to [0, 1]$, such that $\sum_{A \subseteq X} m(A) = 1$ and $m(\phi) = 0$, where ϕ is the empty set and A is any subset of X. A subset A with a non-zero mass $m(A) > 0$ is called the focal element of m. The D-S theory provides a way of combining independent sources of evidence to increase confidence in the overall hypothesis. The combined sources of evidence can be calculated as an orthogonal sum $m = m_1 \oplus m_2 \oplus \cdots \oplus m_n$ for fusing independent information sources m_i, $i = 1, 2, \ldots, n$. The orthogonal sum is defined by Dempster's combination rule:

$$\begin{cases} m(\phi) = 0 \\ m(A) = K \displaystyle\sum_{A_1 \cap \cdots \cap A_n = A} \prod_{i=1}^{n} m_i(A_i) \end{cases} \tag{3}$$

where $A \subseteq X$ and $K^{-1} = 1 - \displaystyle\sum_{A_1 \cap \cdots \cap A_n = \phi} \prod_{i=1}^{n} m_i(A_i)$.

Some common evidence measures are belief (Bel) and plausibility (Pl), which are derived from the mass function m. $\mathrm{Bel}(A)$ measures the total belief that the hypothesis is true and $\mathrm{Pl}(A)$ measures the total belief that can move into A. Bel and Pl are defined as:

$$\begin{cases} \mathrm{Bel}(\phi) = 0 \\ \mathrm{Bel}(A) = \displaystyle\sum_{B \subseteq A} m(B) \end{cases} \tag{4}$$

$$\begin{cases} \mathrm{Pl}(\phi) = 0 \\ \mathrm{Pl}(A) = \displaystyle\sum_{B \cap A \neq \phi} m(B) \end{cases} \tag{5}$$

where $B \subseteq X$, $A \subseteq X$. From Equations (4) and (5), $\mathrm{Pl}(A) \geq \mathrm{Bel}(A)$, and the following relationship holds: $\mathrm{Pl}(A) = 1 - \mathrm{Bel}(\bar{A})$, where $\bar{A} = X - A$ is the complementary set of A.

2.3. **Lower-upper-middle filter.** Let $K = \{k = (k_1, k_2) | 1 \leq k_1 \leq H, 1 \leq k_2 \leq W\}$ denote the pixel coordinates of the noisy image corrupted by impulsive noise, where H and W are the image height and width, respectively. Let $x(k)$ represent the input pixel value of the noisy image at location $k \in K$. At each location k, the observed filter window $w\{k\}$, whose size is $N = 2n + 1$ (n is a non-negative integer), is defined in terms of the coordinates symmetrically surrounding the input pixel $x(k)$.

$$w\{k\} = \{x_f(k) : f = 1, 2, \ldots, n, n+1, \ldots, N\}, \tag{6}$$

where the input pixel $x(k) = x_{n+1}(k)$ is the center pixel.

LUM filters, which have been shown to be equivalent to CWM filters [33,34], are defined as:

$$y(k) = MED\left\{x_{(l)}(k), x(k), x_{(N-l+1)}(k)\right\}, \tag{7}$$

where MED denotes the median operation, $1 \leq l \leq (N+1)/2$, $x(k)$ is the central sample from the filter window $w\{k\}$, and $x_{(1)}(k) \leq x_{(2)}(k) \leq \cdots \leq x_{(N)}(k)$ is the rank-ordered set of $w\{k\}$. Here, $x_{(l)}(k)$ and $x_{(N-l+1)}(k)$ are the lower- and upper-order statistics, respectively, and l is the control weight for smoothing. The weight l can range from $y(k) = x(k)$ (for $l = 1$) to the median (for $l = (N+1)/2$). LUM filters uniformly process the whole noisy image, which can lead to excessive or insufficient smoothing. In this work, to best balance the tradeoff between impulse noise suppression and image detail preservation, the weight l of the LUM filter is made adjustable according to the weight of the multi-class LMS algorithm.

3. Design of MCL Filter.

3.1. **Structure of MCL filter.** As a switching-based filter, the proposed MCL filter consists of a two-pass filtering mechanism to remove impulse noise, as shown in Figure 2. The input pixels are first classified by the proposed SVM noise detector as either noisy or noise-free. If the input pixel $x(k)$ is identified as noisy, then the output of the adaptive LUM filter replaces the pixel. Otherwise, the pixel is kept unchanged. The following subsection discusses the multi-class Dempster classifier and adaptive LUM filter shown in Figure 2. In addition, to improve filtering performance, second-pass filtering is performed by a simple ROM decision filter to remove any noisy pixels that may have been mistakenly left undetected by the SVM noise detector and to restore misdetected uncorrupted pixels.

3.2. **SVM noise detector.** The decision-making process is key to a successful switching-based filter. In this work, an SVM noise detector based on the SVM approach is proposed to identify noise. The proposed decision-making approach consists of two steps: (1) feature extraction and (2) training of the SVM noise detector.

3.2.1. *Feature extraction.* Since noise filtering effectiveness heavily depends on the correctness of noise detection in designing switching-based filters, the SVM noise detector should be made as precise as possible. Before noise filtering begins, the local features of the filter window $w\{k\}$ must be extracted to identify noisy pixels. The local features in the filter window, such as prominent signals and the possible presence of details and edges, are taken into account. The following four variables are defined to create feature vector $V\{k\}$ as the input data of the SVM noise detector.

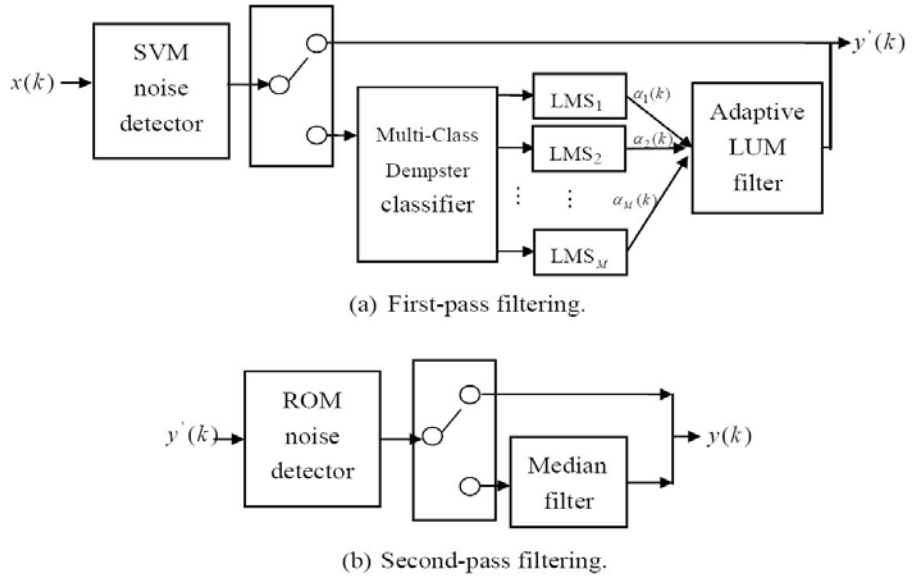

FIGURE 2. Structure of MCL filter

Definition 3.1. *The variable $c(k)$ indicates the absolute difference between the input $x(k)$ and the median value of $w\{k\}$ as follows [13,35]:*

$$c(k) = |x(k) - MED\,(w\{k\})|\,. \tag{8}$$

A large $c(k)$ value reveals that the central pixel $x(k)$ juts out among its neighboring pixels; that is, the input $x(k)$ may be corrupted by impulse noise. Most impulse noise can be detected by using the variable $c(k)$ as an indicator. However, if only the $c(k)$ value is used to consider whether impulse noise exists, it would be difficult to fully separate impulse noise. For example, line components and edges are usually present in an image; therefore, if $x(k)$ is located on a line or an edge, it may be mistakenly identified as impulse noise and removed. To avoid misjudgments, it is necessary to add other observations. Pixel-wise median of absolute deviation (MAD) (PWMAD) is a local feature based on a modified MAD [36]. MAD is used to consider the presence of image details. The PWMAD statistic helps in separating image details from noisy pixels. Let $X(k)$, $M(k)$, and $D(k)$ denote matrices centered around $x(k)$, $MED\{w\{k\}\}$, and $c(k)$, respectively. The median image and absolute deviation image can be defined as follows.

Definition 3.2.

$$\begin{aligned} b(k) = \mathrm{PWMAD}(k) &= c(k) - \mathrm{median}(D(k)) \\ &= c(k) - \mathrm{median}(|X(k) - M(k)|). \end{aligned} \tag{9}$$

Garnett *et al.* proposed a rank-ordered absolute differences (ROAD) statistic as a measure to quantify how different in intensity particular pixels are from their most similar neighbors [22]. In general, unwanted impulses differ greatly in intensity from most or all of their neighboring pixels, whereas most real image pixels have intensities similar to those of at least half of their neighboring pixels, even pixels on an edge. Excluding $x(k)$, for each pixel $x_i(k) \in w\{k\}$, d_i is defined as the absolute difference in intensity between $x(k)$ and $x_i(k)$; i.e., $d_i = |x(k) - x_i(k)|$. Then, d_i values are sorted in increasing order into the sequence $\{d_i\}$.

Definition 3.3.

$$h(k) = \frac{\sum_{i=1}^{4} c_i}{4} \tag{10}$$

where $c_i = ith\ smallest\ \{d_i\}$ *[5].*

The $h(k)$ value is the mean ROAD statistic. Noisy pixels have higher mean ROAD values than those of uncorrupted pixels. The mean ROAD value is a measure of how close a pixel value is to its four most similar neighbors.

Definition 3.4.

$$c^{w_0}(k) = MED\{x_1(k), \cdots, x_n(k), w_0 \diamondsuit x_{n+1}(k), \ldots, x_N(k)\}, \tag{11}$$

where

$$\begin{aligned} &MED\left\{x_1(k), \ldots, x_n(k), w_0 \diamondsuit x_{n+1}(k), \ldots, x_N(k)\right\} \\ &= MED\left\{x_1(k), \ldots, x_n(k), \underbrace{x_{n+1}(k), \ldots, x_{n+1}(k)}_{w_0\ times}, \ldots, x_N(k)\right\}. \end{aligned}$$

Here, w_0 *denotes a non-negative integer weight and* $w_0 \diamondsuit x_{n+1}(k)$ *means that there are* w_0 *copies of input pixel* $x(k) = x_{n+1}(k)$ *[11].*

Definition 3.5.

$$g(k) = \left|x(k) - c^5(k)\right|. \tag{12}$$

In the present study, the feature vector is given by:

$$V\{k\} = \{c(k), b(k), h(k), g(k)\}. \tag{13}$$

The feature vector $V\{k\}$ serves as the input data set to the SVM noise detector.

3.2.2. *Training of SVM noise detector.* The optimal separating hyperplane can be obtained through a training process for the training image. The input in the training process is the set of feature $V\{k\}$ and desired binary flag map F. Pixels with 1's on F are considered corrupted. Pixels with 0's on F are considered uncorrupted. Therefore, the discrimination function separates the training data into two classes. That is, the SVM noise detector classifies the input pixel as noisy or noise-free. The training design of the SVM noise detector is shown in Figure 3. This efficient design minimizes the risk of misclassification not only in the training set (i.e., training errors) but also in the test set (i.e., generalization errors).

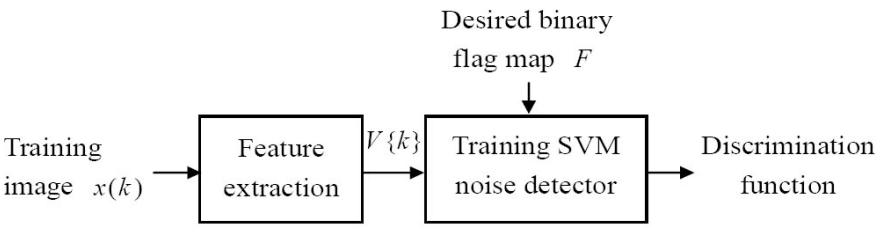

FIGURE 3. Training design of the SVM noise detector

3.3. **Adaptive weight of LUM filter.** Adaptive weight l allows the LUM filter to perform various degrees of noise suppression and image detail preservation [37]. To determine adaptive weight l, a weight controller (the switching scheme shown in Figure 2) is proposed in this work. Note that the multi-class Dempster classifier shown in Figure 2, which is defined as a function of the feature vector, is used to determine the partitioning, and $\alpha_i(k)$, $i \in \{1, 2, \ldots, M\}$ serves as adaptive weight l for the LUM filter. The partition belief value method is used to partition the feature vector space into M blocks, and the LMS learning algorithm is used to set the weight for each block to minimize the mean square error (MSE) of the filter output.

3.4. **Multi-class Dempster classifier.** The multi-class Dempster classifier shown in Figure 2 partitions the final belief value of noise-corrupted pixels to obtain the weight function $\alpha_i(k)$. The feature vector $V\{k\}$ serves as the input data set to the multi-class Dempster classifier. The signals are assigned to one of two classes: S (noisy) and R (noise-free). The hypotheses to be considered in the D-S formulation are: ϕ, singleton hypotheses S and R, and compound hypothesis SR (for convenience, $S \cup R$ is denoted as SR). The variable $c(k)$ can be represented as the mass function $m_{c(k)}(S)$. A large $m_{c(k)}(S)$ value indicates that the input $x(k)$ is dissimilar to the median value of the filter window $w\{k\}$; that is, the evidence suggests that the central pixel $x(k)$ is corrupted by impulse noise.

From the D-S evidence theory, mass functions have the following constraints:

$$\begin{aligned} &m_{c(k)}(\phi) = 0, \\ &m_{c(k)}(S) + m_{c(k)}(R) + m_{c(k)}(SR) = 1. \end{aligned} \tag{14}$$

A non-null mass can be applied to their union SR when the two classes S and R are not distinguishable by the evidence. From Equation (14), $m_{c(k)}(R) + m_{c(k)}(SR)$ is equal to $1 - m_{c(k)}(S)$. Thus, the strategy for selecting mass functions R and SR is:

$$m_{c(k)}(R) = \beta(1 - m_{c(k)}(S)), \tag{15}$$

$$m_{c(k)}(SR) = \beta/2(1 - m_{c(k)}(S)), \tag{16}$$

where β is predefined as $2/3$ empirically [5]. Just as with evidence $c(k)$, the strategy can be used to determine the mass function values $m_{b(k)}(S)$, $m_{h(k)}(S)$, $m_{g(k)}(S)$ according to evidence $b(k)$, $h(k)$, $g(k)$, respectively. $m_{b(k)}(R)$, $m_{h(k)}(R)$, $m_{g(k)}(R)$ and $m_{b(k)}(SR)$, $m_{h(k)}(SR)$, $m_{g(k)}(SR)$ can also be derived like what happens in Equations (15) and (16). Finally, the belief function can be computed from Equation (4) based on the final mass function m.

The belief space B is divided (partitioned) into a set of M mutually exclusive blocks, defined as $\{\Theta_1, \Theta_2, \ldots, \Theta_M\}$, given by:

$$\Theta_i = \{Bel(S) \in B : d(Bel(S)) = i\}, \quad i = 1, 2, \ldots, M, \tag{17}$$

where d is the multi-class Dempster classifier shown in Figure 2. The block corresponding to each input $Bel(S)$ is given such that $d(Bel(S)) = i$ for $Bel(S) \in \Theta_i$. That is, d is a function of the belief value $Bel(S)$. As such, the M blocks Θ_i, $i = 1, 2, \ldots, M$, satisfy:

$$B = \bigcup_{i=1}^{M} \Theta_i \text{ and } \Theta_i \cap \Theta_j = \phi, \text{ for } i \neq j. \tag{18}$$

Notably, each input data $x(k)$ corresponding to its $Bel(S)$ is only included in one of the M blocks [5].

In this work, a step-like function comprising a finite number of piecewise regions is used as the multi-class Dempster classifier because of its efficiency and simple computation.

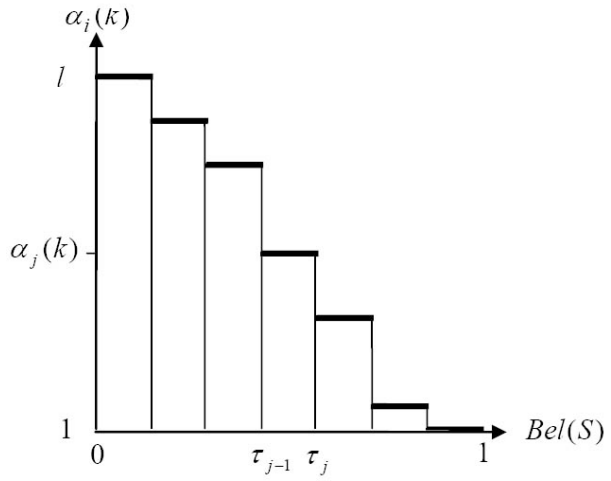

FIGURE 4. Approximation of weight function $\alpha_i(k)$ to step-like function

Figure 4 shows an example of how the value of $Bel(S)$ can be approximated by a step-like function [5]. Each input data $x(k)$ corresponding to its $Bel(S)$ is only included in one of those M blocks according to the function. According to the partitioning of the belief space, the weight function for the pixel $x(k)$ can be expressed as $\alpha_i(k)$, $i = 1, 2, \ldots, M$.

The performance of the LUM filtering result is settled by the weight-setting function $\alpha_i(k)$. Each independent weight $\alpha_i(k)$ can be obtained using the LMS learning algorithm subject to minimizing the cost function $\hat{\varepsilon}_i(k)$ with respect to block Θ_i [38]:

$$\hat{\varepsilon}_i(k) = \frac{1}{2} \sum_k (o(k) - y(k))^2, \tag{19}$$

where $o(k)$ denotes the desired output and $y(k)$ denotes the physical output of the MCL filter. Since the M blocks are mutually exclusive, the total minimum MSE can be expressed as:

$$\varepsilon = \sum_{i=1}^{M} \hat{\varepsilon}_i(k). \tag{20}$$

By differentiating Equation (19), the updated rule associated with the training algorithm can be formulated as:

$$\alpha_i^{(t+1)}(k) = \alpha_i^{(t)}(k) - \eta_i f(k) \left(x^{(t)}(k) - o^{(t)}(k) \right). \tag{21}$$

Here, η_i denotes the learning rate (which is a convergence factor), $f(k)$ is the difference between the desired output $o(k)$ and the physical output $y(k)$, and t is the time point when pixel $x(k)$ is processed; the time point is $t + 1$ when the next pixel $x(k)$ is also included in the ith block [5]. The value $\alpha_i(k)$ becomes optimal, giving the minimum MSE when $\alpha_i(k)$ sufficiently converges. That is, the weight function $\alpha_i(k)$ has the knowledge of where noises and image signals.

3.5. **ROM noise detector.** The SVM noise detector may make two types of mistake. Type I errors (also known as misses [41]) happen when undetected noisy pixels remain in the restored image (i.e., the SVM noise detector does not detect them to be noisy pixels). Type II errors occur when mis-detected pixels appear in the restored image. The LUM filter modifies these pixels even though they are good pixels. Hence, a simple decision filter is required to improve the performance of the MCL filter. The second-pass noise filtering of the MCL filter is based on an ROM detector and a median filter, as shown

in Figure 2. The observation samples $w'\{k\}$ are defined as an eight-element observation vector of a 3×3 filter window centered around $x(k)$ (excluding $x(k)$ itself) such that:

$$w'\{k\} = \{x_1(k), \ldots, x_n(k), x_{n+2}(k), \ldots, x_9(k)\}. \tag{22}$$

The observation samples $w'(k)$ can be sorted in ascending order, yielding:

$$w'\{k\} = \{a_1(k), a_2(k), \ldots, a_8(k)\}, \tag{23}$$

where $a_1(k) \leq a_2(k) \leq \ldots \leq a_8(k)$ are elements of $w'\{k\}$. Thus, ROM(k) is defined as:

$$\mathrm{ROM}(k) = \frac{a_4(k) + a_5(k)}{2}. \tag{24}$$

The rank-ordered differences are defined as [20]:

$$\mathrm{d}(k) = [d_1(k), d_2(k), d_3(k), d_4(k)],$$

where

$$\mathrm{d}_i(k) = \begin{cases} a_i(k) - x(k), & x(k) \leq \mathrm{ROM}(k) \\ x(k) - a_{9-i}(k), & x(k) > \mathrm{ROM}(k), \end{cases} \quad \text{for } i = 1, \ldots, 4. \tag{25}$$

The rank-ordered differences provide evidence about the likelihood of corruption of input pixel $x(k)$ [20]. The second-pass filter for detecting corrupted pixels is based on the ROM noise detector, which operates as follows:

$$\mathrm{d}_i(k) > T_i, \quad i = 1, \ldots, 4, \tag{26}$$

where T_1, T_2, T_3, T_4 are threshold values, with $T_1 < T_2 < T_3 < T_4$. The ROM noise detector detects $x(k)$ as a noisy pixel if Equation (26) is true for any i. Then, the second-pass filter employs the threshold values to determine the final output $y(k)$:

$$y(k) = \begin{cases} MED(w'\{k\}), & \text{if } \mathrm{d}_i(k) > T_i \\ y'(k), & \text{else.} \end{cases} \tag{27}$$

If the input pixel is classified as noisy by the ROM noise detector, the pixel is replaced with the median value. Otherwise, no filtering is activated.

4. **Experimental Results.** A source image corrupted by impulse noise of density (or noise ratio) p can be described as follows:

$$x(k) = \begin{cases} a(k), & \text{with probability } p, \\ z(k), & \text{with probability } 1 - p, \end{cases} \tag{28}$$

where $a(k)$ and $z(k)$ represent the noise substitutions for the original pixel value and original noise-free pixel value, respectively. There are two types of impulse noise: fixed-valued and random-valued. In an 8-bit gray-scale image, fixed-valued impulse noise, which is also known as salt and pepper noise, has equal probability of noise intensity at 0 and 255. Random-valued impulse noise is uniformly distributed over the range of $[0, 255]$ with noise ratio p.

Extensive experiments were conducted on a variety of 512×512 test images to evaluate the performance of the proposed MCL filter. 3×3 filter windows were used in all the experiments. The peak signal-to-noise ratio (PSNR) and mean absolute error (MAE) were used to quantitatively evaluate the restoration performance and the detail preservation capability, respectively. PSNR is defined as:

$$PSNR = 10 \log_{10} \left(\frac{\sum_k 255^2}{\sum_k (o(k) - y(k))^2} \right) dB \tag{29}$$

where 255 is the peak gray-level of the image, $o(k)$ represents the value of the desired output, and $y(k)$ represents the value of the physical output. MAE is defined as:

$$MAE = \frac{\sum_k |o(k) - y(k)|}{H \times W} \tag{30}$$

where H and W are the image height and width, respectively.

The optimal separating hyperplane was obtained using training image 'Couple' corrupted by 20% impulse noise in the training process. The tested images were outside the training set to test the generalization capability. Table 1 shows the accuracy of the SVM noise detector for some images corrupted by 20% impulse noise.

TABLE 1. Classification accuracy of SVM classifier

Image	Lena	Lake	Boat	F16	Cameraman	Goldhill
Fixed-valued impulse	99.61%	98.70%	98.94%	99.32%	98.34%	99.30%
Random-valued impulse	98.58%	97.54%	98.28%	98.41%	97.43%	99.12%

The reference image 'Couple' corrupted by 20% impulse noise was taken as the training image for the LUM weight controller. To approximate the weight function to a step-like function, the parameters concerning the width of each step such as τ_{j-1} and τ_j in Figure 4 are required. The axis was equally divided into piecewise regions with the number of blocks. That is, the optimal number of blocks in partitioning needs to be obtained for the weight function $\alpha_i(k)$ in the MCL filter. Therefore, all the weights $\alpha_i(k)$, $i = 1, 2, \ldots, M$ in the belief space can be trained independently using Equation (21). Here, M denotes the number of blocks. The optimal weights remained constant during the filtering stage throughout the experiments. Figure 5 shows the PSNR comparison results of an image corrupted by 20% fixed-valued impulse noise for various numbers of blocks used and Figure 6 shows the PSNR comparison results of an image corrupted by 20% random-valued impulse noise. The number of blocks was quite consistent for the fixed-valued and random-valued impulse noise, as shown in Figures 5 and 6. In this work, to achieve good robustness, the number of blocks M was preset to 25 (obtained from extensive experiments) for all the experiments.

The effectiveness of the proposed MCL filter was assessed by comparing its results with those obtained using the MED filter, SD-ROM filter [20], SWM-I filter [16], TSM filter

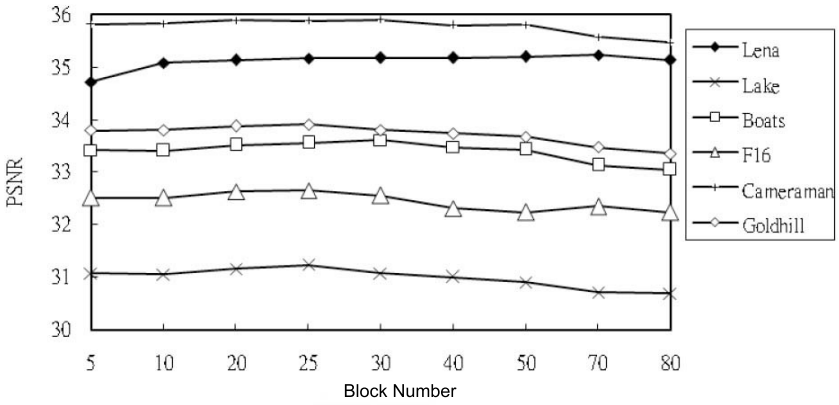

FIGURE 5. PSNR (dB) comparison results of image corrupted by 20% fixed-valued impulse noise for various numbers of blocks

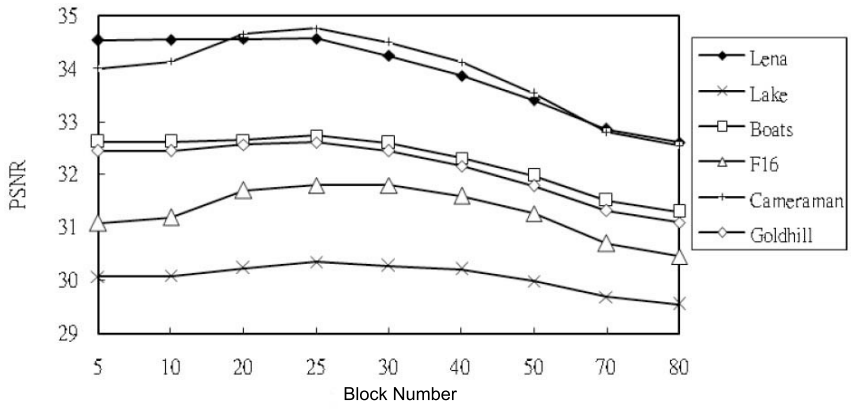

FIGURE 6. PSNR (dB) comparison results of image corrupted by 20% random-valued impulse noise for various numbers of blocks

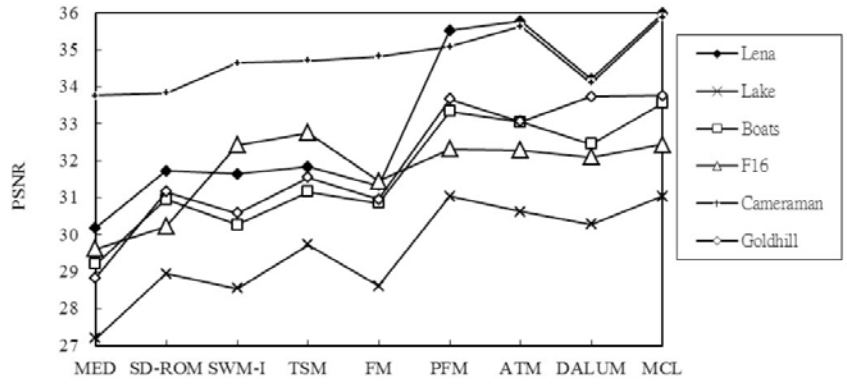

FIGURE 7. Comparison of PSNR (dB) results for 20% fixed-valued impulse noise

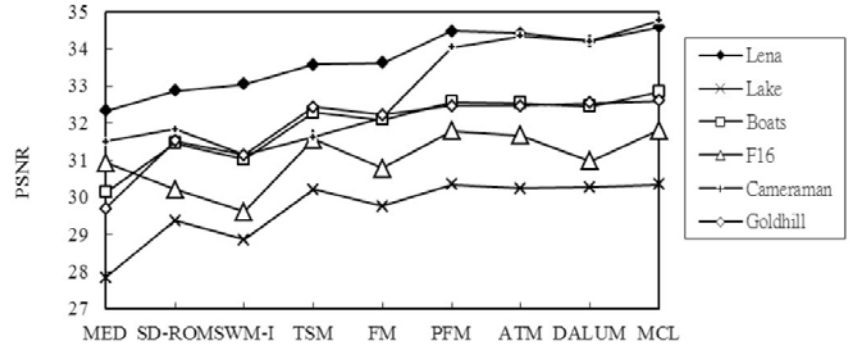

FIGURE 8. Comparison of PSNR (dB) results for 20% random-valued impulse noise

[21], fuzzy median (FM) filter [14], partition fuzzy median (PFM) filter [15], ATM filter [25], and decision-based adaptive low-upper-middle (DALUM) filter [43]. The optimal parameters were set for each method (as suggested by its author(s)). Figure 7 shows the PSNR comparison results of noise removal for images corrupted by 20% fixed-valued impulse noise. The proposed MCL filter has the highest PSNR values. Figure 8 shows the PSNR comparison results of noise removal for images corrupted by 20% random-valued impulse noise. The proposed MCL filter again has the highest PSNR values. Figure 9 shows the results of filtering the 'Goldhill' image corrupted by 20% random-valued impulse noise and the corresponding MAE values. The MCL filter gives the best visual quality

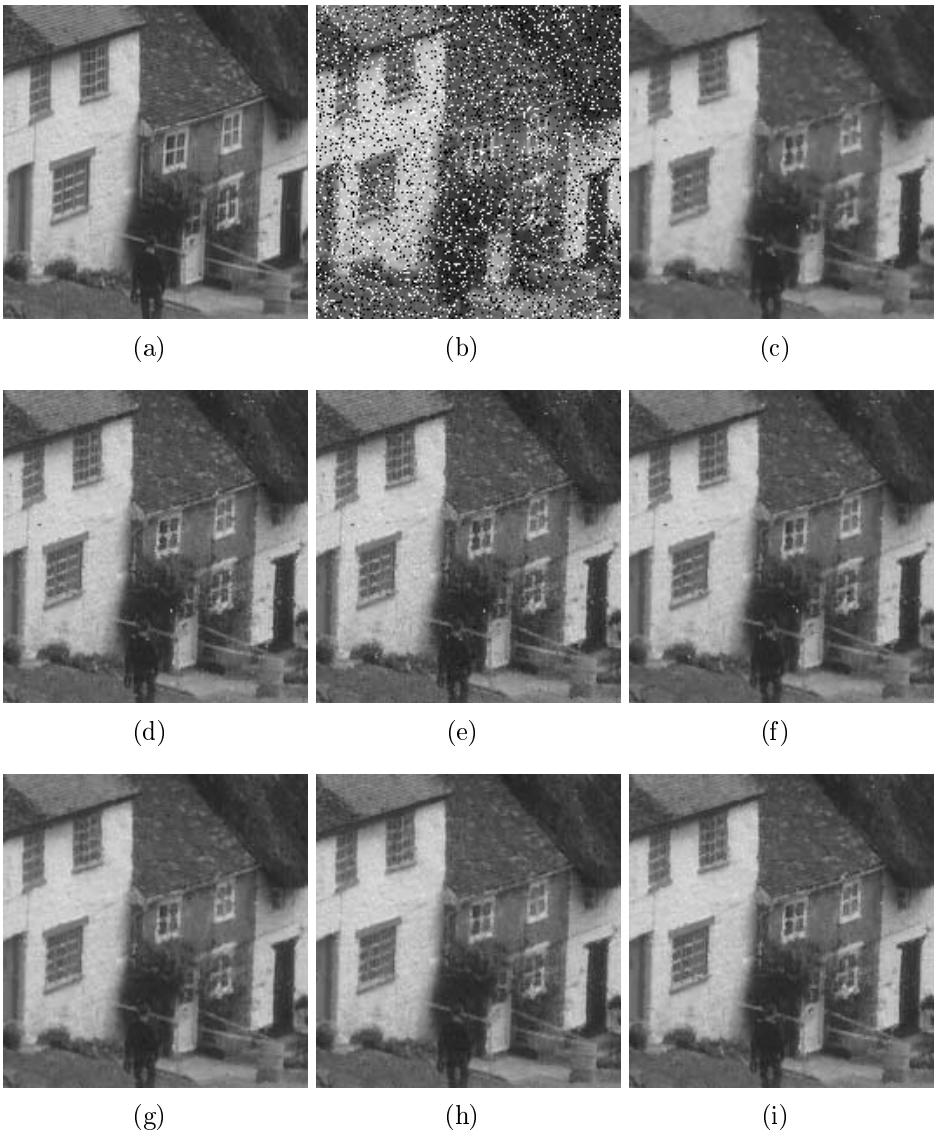

FIGURE 9. Restoration performance comparison for 'Goldhill' image degraded by 20% impulse noise: (a) original image, (b) noisy image, and images filtered by (c) MED filter, MAE = 4.84, (d) SD-ROM filter, MAE = 2.59, (e) SWM-I filter, MAE = 2.92, (f) FM filter, MAE = 2.59, (g) PFM filter, MAE = 2.68, (h) ATM filter, MAE = 2.29, and (i) MCL filter, MAE = 2.17.

for the restored image with more noise attenuation and detail preservation, giving the smallest MAE value. The robustness of the MCL filter was also tested. Figures 10 and 11 show the PSNR comparison with restored image 'Boats' corrupted by 5% to 40% impulse noise. As shown in the figure, the MCL filter outperforms the other techniques in terms of robustness, even though the 20% impulse noise training image 'Couple' is independent of the actual corruption percentage in filtering.

5. **Conclusions.** In this work, a switching-based filter was proposed for improving the performance of median-based filters, preserving image details while effectively suppressing noise. An SVM noise detector was proposed to judge whether an input pixel is noisy. Sources of evidence are extracted, and the mass function is defined by using the local

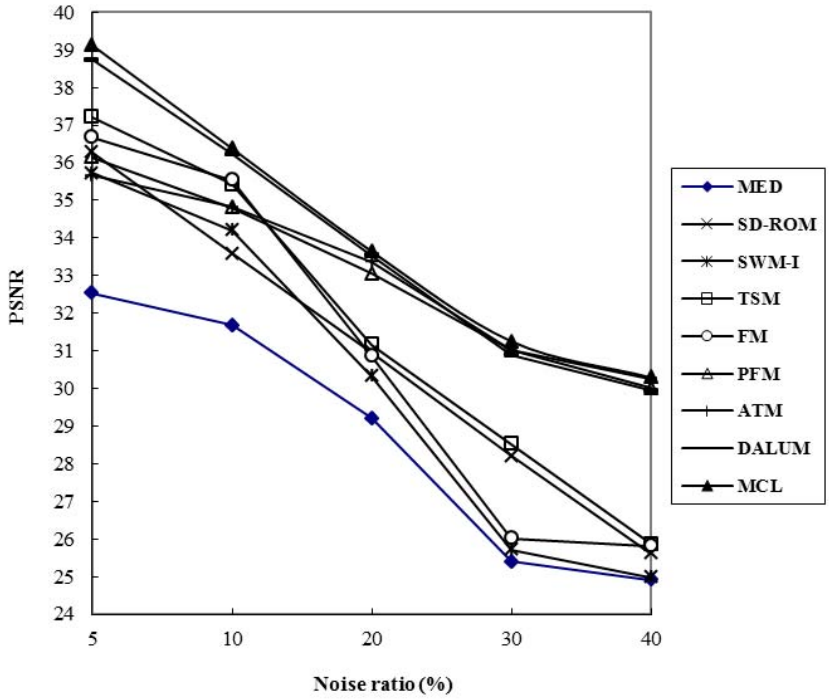

FIGURE 10. PSNR performance evaluation of filtering 'Boats' image corrupted by fixed-valued impulse noise at various noise ratios

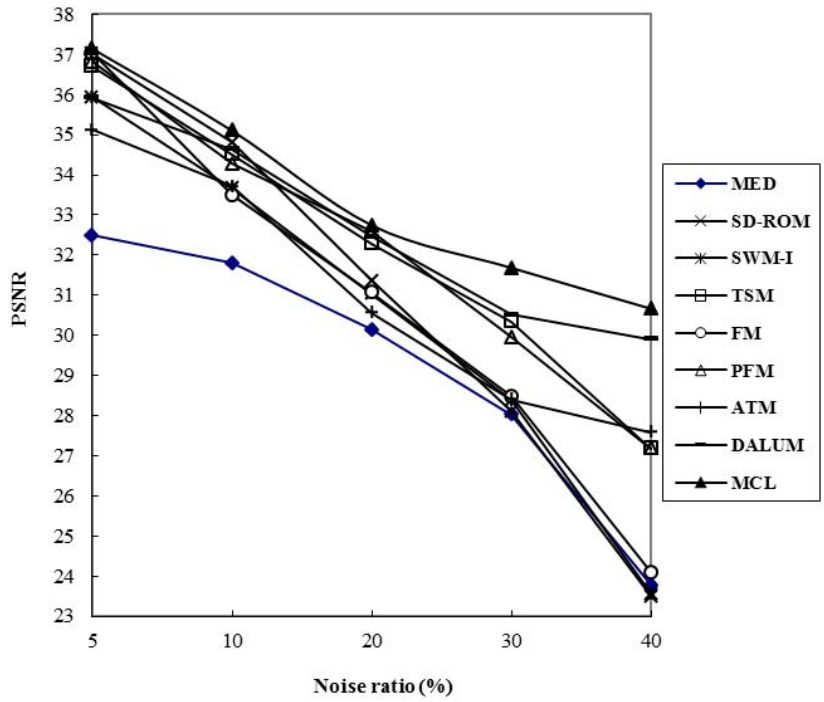

FIGURE 11. PSNR performance evaluation of filtering 'Boats' image corrupted by random-valued impulse noise at various noise ratios

information in the filter window to decide the belief value. An efficient step-like function is used to partition the belief space, and the LMS algorithm is applied to obtain the optimal weight for each block. Therefore, an adaptive LUM filtering operation can be activated. To improve the final filtering performance, an ROM filter is also adopted. Experimental results show that the proposed filter substantially outperforms many existing filters in terms of image restoration.

Acknowledgment. The author would like to thank Prof. Chih-Jen Lin for providing LIBSVM software, a library for SVMs (version 2.83), and the National Science Council of Taiwan for financially supporting this research under grant NSC 102-2221-E-274-006.

REFERENCES

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

[2] A. M. Mirza, A. Chaudhry and B. Munir, Spatially adaptive image restoration using fuzzy punctual kriging, *Journal of Computer Science and Technology*, vol.22, pp.580-589, 2007.

[3] X.-H. Han, Y.-W. Chen and J.-M. Lei, A spatio-chromatic ICA based noise reduction in color images, *International Journal of Innovative Computing, Information and Control*, vol.4, no.3, pp.661-669, 2008.

[4] X. Guo, H. Zhang and Z. Chang, Image thresholding algorithm based on image gradient and fuzzy set distance, *ICIC Express Letters*, vol.4, no.3(B), pp.1059-1063, 2010.

[5] T.-C. Lin, Partition belief median filter based on Dempster-Shafer theory in image processing, *Pattern Recognition*, vol.41, pp.139-151, 2008.

[6] M.-H. Tsai, Y.-B. Lin and C.-M, Wang, Image sharing with steganography and cheater identification, *International Journal of Innovative Computing, Information and Control*, vol.6, no.3(A), pp.1165-1178, 2010.

[7] P. E. Trahanias, D. Karakos and A. N. Venetsanopoulos, Directional processing of color images: Theory and experimental results, *IEEE Trans. Process.*, vol.5, pp.868-880, 1996.

[8] J. Astola, P. Haavisto and Y. Neuvo, Vector median filters, *Proc. of the IEEE*, vol.78, pp.678-689, 1990.

[9] L. Jin and D. Li, Improved directional-distance filter, *Opt. Precis. Eng.*, vol.15, pp.798-806, 2007.

[10] O. Yli-Harja, J. Astola and Y. Neuvo, Analysis of the propertities of median and weighted median filters using threshold logic and stack filter representation, *IEEE Trans. Signal Process.*, vol.39, pp.395-410, 1991.

[11] S. J. Ko and Y. H. Lee, Center weighted median filters and their applications to image enhancement, *IEEE Trans. Circuits and Systems*, vol.38, pp.984-993, 1991.

[12] L. Jin, C. Xiong and D. Li, Adaptive center-weighted median filter, *J. Huazhong Univ. Sci. Technol.*, vol.36, pp.9-12, 2008.

[13] T.-C. Lin, A new adaptive center weighted median filter for suppressing noise in images, *Information Sciences*, vol.177, pp.1073-1087, 2007.

[14] K. Arakawa, Median filters based on fuzzy rules and its application to image restoration, *Fuzzy Sets and Systems*, vol.77, pp.3-13, 1996.

[15] T.-C. Lin and P.-T. Yu, Partition fuzzy median filter based on fuzzy rules for image restoration, *Fuzzy Sets and Systems*, vol.147, pp.75-97, 2004.

[16] T. Sun and Y. Neuvo, Detail-preserving median based filters in image processing, *Pattern Recognition Letters*, vol.15, pp.341-347, 1994.

[17] R. Lukac, K. N. Plataniotis and A. N. Venetsanopoulos, A statistically switched adaptive vector median filter, *J. Intell. Robot. Syst.*, vol.38, pp.984-993, 1991.

[18] V. Gregorijon-Gerard Camarena, S. Morillas and A. Sapena, Fast detection and removal of impulse noise using peer groups and fuzzy metrics, *J. Visual Commun. Image Represent.*, vol.19, pp.20-29, 2008.

[19] B. Smolka, Peer group switching filter for impulse noise reduction in color images, *Pattern Recognition Lett.*, vol.12, pp.1016-1025, 2009.

[20] E. Abreu and S. K. Mitra, A signal-dependent rank ordered mean (SD-ROM) filter: A new approach for removal of impulses from highly corrupted images, *Proc. of the IEEE ICASSP-95*, Detroit, MI, pp.2371-2374, 1995.

[21] T. Chen, K. K. Ma and L. H. Chen, Tri-state median filter for image denoising, *IEEE Trans. Image Processing*, vol.8, pp.1834-1838, 1999.
[22] R. Garnett, T. Huegerich, C. Chui and W. He, A universal noise removal algorithm with an impulse detector, *IEEE Trans. Image Processing*, vol.11, pp.1747-1754, 2005.
[23] R. Lukac, Binary LUM smoothing, *IEEE Signal Processing Letters*, vol.19, pp.400-403, 2002.
[24] K. S. Pankaj and M. Banshidhar, An improved adaptive impulsive noise suppression scheme for digital images, *International Journal of Electronics and Communication*, vol.64, pp.322-328, 2010.
[25] T.-C. Lin and P.-T. Yu, Adaptive two-pass median filter based on support vector machine for image restoration, *Neural Computation*, vol.16, pp.333-354, 2004.
[26] H. Liu, F. Sun and Z. Sun, Image filtering using support vector machine, *Lecture Notes in Computer Science*, vol.3972, pp.533-538, 2006.
[27] V. Vapnik, *Statistical Learning Theory*, Wiley, New York, 1998.
[28] C.-C. Chang and C.-J. Lin, Training nu-support vector regression: Theory and algorithms, *Neural Computation*, vol.14, pp.1959-1977, 2002.
[29] C.-W. Hsu and C.-J. Lin, A simple decomposition method for support vector machines, *Machine Learning*, vol.46, pp.291-314, 2002.
[30] C.-C. Chang and C.-J. Lin, *LIBSVM: A Library for Support Vector Machines*, 2001.
[31] J. W. Guan and D. A. Bell, *Evidence Theory and Its Applications*, North-Holland, New York, 1991.
[32] G. Shafer, *A Mathematical Theory of Evidence*, Princeton University Press, Princeton, NJ, 1976.
[33] R. Lukac and S. Marchevsky, Boolean expression of LUM smoothers, *IEEE Signal Processing Letters*, vol.8, pp.292-294, 2001.
[34] R. Lukac, Performance boundaries of optimal weighted median filters, *International Journal of Image and Graphics*, vol.4, pp.157-182, 2004.
[35] T.-C. Lin and P.-T. Yu, Partition fuzzy median filter based on fuzzy rules for image restoration, *Fuzzy Sets and Systems*, vol.147, pp.75-97, 2004.
[36] P. Kumar Sa and B. Majhi, An improved adaptive impulsive noise suppression scheme for digital images, *International Journal of Electronics and Communications*, vol.64, pp.322-328, 2010.
[37] T.-C. Lin, Decision-based filter based on SVM and evidence theory for image noise removal, *Neural Computing and Applications*, vol.21, pp.695-703, 2012.
[38] C. Kotropoulos and I. Pitas, Constrained adaptive LMS L-filters, *Signal Processing*, vol.26, pp.335-358, 1992.
[39] B. Smolka and A. Chydzinski, Fast detection and impulsive noise removal in color images, *Rea-Time Imaging*, vol.11, pp.389-402, 2005.
[40] H. L. Huang and F. L. Chang, ESVM: Evolutionary support vector machine for automatic feature selection and classification of microarray data, *BioSystems*, vol.90, pp.516-528, 2007.
[41] X. Xu, E. L. Miller, D. Chen and M. Sarhadi, Adaptive two-pass rank order filter to remove impulse noise in highly corrupted images, *IEEE Trans. Image Processing*, vol.13, pp.238-247, 2004.
[42] X.-Y. Wang, H.-Y. Yang, Y. Zhang and Z.-K. Fu, Image denoising using SVM classification in nonsubsampled contourlet transform domain, *Information Sciences*, vol.246, pp.155-176, 2013.
[43] T.-C. Lin and C.-M. Lin, Decision-based adaptive low-upper-middle filter for image processing, *International Journal of Innovative Computing, Information and Control*, vol.7, no.10, pp.5977-5990, 2011.
[44] S. K. Kayhan, An effective 2-stage method for removing impulse noise in images, *Journal of Visual Communication and Image Representation*, vol.25, pp.478-486, 2014.
[45] Y. Li, J. Sun and H. Luo, A neuro-fuzzy network based impulse noise filtering for gray scale images, *Neurocomputing*, vol.127, pp.190-199, 2014.
[46] B. Zhou, S. Wang, Y. Ma, X. Mei, B. Li, H. Li and F. Fan, An infrared image impulse noise suppression algorithm based on fuzzy logic, *Infrared Physics and Technology*, vol.60, pp.346-358, 2013.
[47] X. Lan and Z. Zuo, Random-valued impulse noise removal by the adaptive switching median detectors and detail-preserving regularization, *International Journal for Light and Electron Optics*, vol.125, pp.1101-1105, 2014.
[48] U. Ghanekar and R. Pandey, An intensity independent fixed valued impulse noise detector for image restoration, *International Journal of Electronics and Communications*, vol.68, pp.210-215, 2014.