Partition-Based Vector Filtering Technique for Suppression of Noise in Digital Color Images

Zhonghua Ma, *Member, IEEE*, Hong Ren Wu, and Dagan Feng, *Fellow, IEEE*

***Abstract*—A partition-based adaptive vector filter is proposed for the restoration of corrupted digital color images. The novelty of the filter lies in its unique three-stage adaptive estimation. The local image structure is first estimated by a series of center-weighted reference filters. Then the distances between the observed central pixel and estimated references are utilized to classify the local inputs into one of preset structure partition cells. Finally, a weighted filtering operation, indexed by the partition cell, is applied to the estimated references in order to restore the central pixel value. The weighted filtering operation is optimized off-line for each partition cell to achieve the best tradeoff between noise suppression and structure preservation. Recursive filtering operation and recursive weight training are also investigated to further boost the restoration performance. The proposed filter has demonstrated satisfactory results in suppressing many distinct types of noise in natural color images. Noticeable performance gains are demonstrated over other prior-art methods in terms of standard objective measurements, the visual image quality and the computational complexity.**

***Index Terms*—Center-weighted vector median (CWVM) filter, constrained least mean-square (LMS) algorithm, digital color image restoration, partition-based adaptive vector (PBTVM) filter.**

I. Introduction

NONLINEAR vector filtering techniques have generated much research interest in the last decade due to its importance in color image restoration. Numerous filtering techniques proposed to date are based on multivariate order statistics [1]–[4], which take the advantage of color inter-channel dependence and avoid unpleasant drawbacks of component-wise filtering techniques, i.e., pixel value rearranging and chromatic shifting [5], [6]. The most well-known vector filters include the vector median filters (VMF) [7], the vector directional filter (VDF) [8], and the directional-distance filter (DDF) [9]. They perform well in suppressing impulse noise and ouliers, but introduce blurring artifacts in edge and detail areas which feature high spacial frequency contents and variations. Weighted nonlinear vector filtering techniques have been proposed to achieve better performance in noise suppression and detail preservation, which include distance-based weighted vector filters [10]–[13], weighted VDFs [14], [15], and the generalized selection weighted vector filters [16]. Adaptive vector filters have been advanced for their effectiveness with respect to different types of noise distributions and image structures, which include filters utilizing local order statistics information [17]–[22], based on gradient information [23], with peer group classification [23], and using a digital path approach [25], to name a few. In the case where noise model cannot be known *a priori*, fuzzy vector filtering techniques have been developed and achieved robust performance [26]–[31]. Recently, a class of chromatic filters worked in the YCbCr color space is also proposed to achieve better chromatic smoothness [32]. Please refer to [33] for a comprehensive review on color image restoration techniques.

The fact that different types of noise contaminate color images in distinct ways poses a major challenge for the vector filtering adaption [6], [33], [34]. Impulse noise destroys only a small portion of an image and leaves other pixels noise free, while additive noise contaminates every image pixel with a certain type of statistical distribution. Detection-based vector filtering techniques, such as the adaptive vector median filter (AVMF) [12], the adaptive vector LUM smoother (AVLUM) [13], the modified weighted vector median (MWVM) filter [11], and the selection center-weighted vector directional filter (SCWVDF)/adaptive center-weighted vector directional filter (ACWVDF) [14], were specially designed to remove impulse noise. They utilize a series of weighted vector median filters to perform binary noise detection, and switch the outputs between an identity filter and a weighted vector median filter according to the detection results. Such a structure, while efficient for impulse noise removal, is inappropriate in dealing with additive noise or mixed noise contamination. In comparison, many adaptive vector filters, such as the adaptive nearest-neighbor multichannel filter (ANNF) [17], the adaptive vector directional filter (AVDF) [18], and the multichannel Π filter [23], yield output by a robust weighting estimation based on all image vectors in the filter window. They are designed to cope with additive and mixed noise corruption, but at the cost of possible image edges and details smearing. Some fuzzy vector filtering techniques [28], [31] have tried to combine different types of standard vector filters together by a large number of fuzzy rules. However, the systematic optimization of such an approach remains as an open problem. Others techniques

Manuscript received July 22, 2004; revised February 9, 2005; This work was supported by grants from the Australian Research Council. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Vicent Caselles.

Z. Ma was with the School of Computer Science and Software Engineering, Monash University, VIC 3800, Australia. He is now with the School of Information Technologies, University of Sydney, Sydney, NSW 2006, Australia (e-mail: frankma@it.usyd.edu.au).

H. R. Wu was with the School of Computer Science and Software Engineering, Monash University, Melbourne, VIC 3800, Australia. He is now with School of Electrical and Computer Engineering, Royal Melbourne Institute of Technology, Melbourne, VIC 3001, Australia.

D. Feng is with the School of Information Technologies, University of Sydney, Sydney, NSW 2006, Australia, and also with the Center for Multimedia Signal Processing, Department of Electronic and Information Engineering, Hong Kong Polytechnic University, Hong Kong.

Digital Object Identifier 10.1109/TIP.2006.877066

1057-7149/$20.00 © 2006 IEEE

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

[26], [27], [29] proposed a group of fuzzy similarity functions within the weighted filter structure to enhance the robustness of their performance. However, the selection of optimal similarity functions is left to the user to decide, more often than not, in an ad hoc manner. It is still a challenging task to design an efficient vector filtering framework to cope with all different types of noise in color images, i.e., impulse noise, additive noise, as well as mixed noise, with its performance comparable to that of the state-of-the-art filtering techniques designed for each of these distinct types of noise.

In this paper, a novel partition-based vector filtering framework is proposed for restoring color images contaminated by different types of noise. The novelty of the filter lies in its unique three-stage adaptive estimation, which consists of a novel *center-weighted trimmed vector median* (CWTVM) filter, a distance-based classification scheme, and a partition indexed weighted filtering operation. The CWTVM filter is used to provide robust reference estimates. The distance-based classification scheme is used to map the local inputs into one of preset structure cells. The weighted filtering operation is designed to provide the best reconstruction with regard to the specific structure partition classification. Off-line training is used to determine the filtering weights for each structure partition cell to achieve the best computation efficiency and performance, using a constrained least mean-square (LMS) algorithm well-defined in the vector space. The proposed vector filtering structure, named hereafter as the *partition-based trimmed vector median* (PBTVM) filter, has achieved superior performance to a number of well-known benchmarks in removing different types of impulse noise, additive noise and mixed noise, in terms of standard objective measurements and the subjective quality.

The paper is structured as follows. The central weighted reference filter is first formulated in Section II. Then in Section III, the framework of the PBTVM filter is described in detail. Section IV addresses the structure partitioning, the weight training, and the recursive implementation of the filtering/training. A number of experimental results are presented in Section V and a brief conclusion is drawn in Section VI.

II. Center-Weighted Reference Filter

A. Review of the CWVM Filter

A weighted vector median (WVM) filter [10], [35] has been widely used in color image restoration for its high flexibility in noise suppression and detail preservation, and a number of algorithms [16], [36], [37] have been developed to optimize its multiple filtering weights. Center-weighted vector median (CWVM) filter [11]–[15] is a special type of the WVM filter where only the weight of the CP is adjustable. It inherits the flexibility of the WVM filter but greatly reduces the complexity of implementation and optimization. In this section, the structure of the CWVM filter is reviewed along with associated terminologies that will be used in later sections.

Let $\mathbf{C} = \{\mathbf{c} = (c_1, c_2) \mid 1 \le c_1 \le H, 1 \le c_2 \le W\}$ be the pixel coordinates of an RGB color image, where H and W are the height and the width of the image, respectively. At each coordinate $\mathbf{c} \in \mathbf{C}$, a multivariate $\mathbf{x}(\mathbf{c}) = [x^R(\mathbf{c}), x^G(\mathbf{c}), x^B(\mathbf{c})]^T$ is used to represent its pixel value, and a square filter window,

$\mathbf{x}_1(\mathbf{c})$	$\mathbf{x}_2(\mathbf{c})$	$\mathbf{x}_3(\mathbf{c})$
$\mathbf{x}_4(\mathbf{c})$	$\mathbf{x}_5(\mathbf{c})$	$\mathbf{x}_6(\mathbf{c})$
$\mathbf{x}_7(\mathbf{c})$	$\mathbf{x}_8(\mathbf{c})$	$\mathbf{x}_9(\mathbf{c})$

Fig. 1. Indexing of the pixel samples in a 3×3 filter window.

$\mathbf{W}(\mathbf{c}) = \{\mathbf{x}_1(\mathbf{c}), \mathbf{x}_2(\mathbf{c}), \ldots, \mathbf{x}_N(\mathbf{c})\}$, is also defined, which centers at the coordinate $\mathbf{c}$ and contains N pixel samples (where N is an odd integer). The pixel samples of the window are indexed in a lexicographic scan order, so that the central pixel (CP) of the window, $\mathbf{x}(\mathbf{c})$, always equals to $\mathbf{x}_{\frac{N+1}{2}}(\mathbf{c})$. Fig. 1 illustrates the indexed window pixels of a 3×3 filter window, where $\mathbf{x}(\mathbf{c}) \triangleq \mathbf{x}_5(\mathbf{c})$ is the CP of the window.

Based on the terminologies above, the output of a CWVM filter with a positive integer central weight, k, is given by [10]–[13], [36]

$$\mathbf{y}_k(\mathbf{c}) = \mathbf{x}_m(\mathbf{c}) \in \mathbf{W}(\mathbf{c}) \mid m = \arg \min_{1 \le i \le N} \{R_i^k\} \quad (1)$$

where R_i^k is the center-weighted aggregated vector distance (CW-AVD) given by

$$R_i^k = (2k-1)\|\mathbf{x}_i(\mathbf{c}) - \mathbf{x}(\mathbf{c})\| + \sum_{j=1, j \ne \frac{N+1}{2}}^{N} \|\mathbf{x}_j(\mathbf{c}) - \mathbf{x}_i(\mathbf{c})\| \quad (2)$$

and $\|\cdot\|$ denotes the l_2 norm (Euclidian distance) in this paper, whereas other vector measures, such as the l_1 norm [7], the generated Minkowski metric [6], the vector directional distance [16], or other measures [33], [38], can be used here, as well.

The central weight of the CWVM filter can be adjusted efficiently to achieve different degrees of noise suppression and image preservation. When the central weight $k = 1$, the CWVM filter reduces to a standard VMF filter, which is powerful for noise suppression but smears small details in the filter window. As the value of k increases, the detail preservation of the filter is improved while the noise suppression performance is degraded. The CWVM filter becomes an identity filter when $k \ge (N+1)/2$. Proofs of such properties of the CWVM filter can be found in [10], [14]. An example of the outputs of the CWVM filter is provided in Appendix.

B. Formulation of the CWTVM Filter

Despite the advantages of the CWVM filter and its success in color image restoration, it has been found in our experiments that a CWVM filter, if working with a relatively high central weight value, often turns out to be oversensitive to impulse

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

noise. This renders the CWVM filter an inappropriate reference filter for the robust local structure classification in Section III. Therefore, a novel CWTVM filter, which with high central weights can achieve more robust performance than the CWVM filter, is proposed in this subsection.

The structure of the CWTVM filter is distinct from that of the CWVM filter in three aspects.

1) Before the CW-AVD calculation, all pixels of a filter window are ranked in an ascending order according to their Euclidian distances to the CP of the window.
2) Then a trimming technique is applied to the pixels of the window according to the central weight value. Pixels which have been ranked beyond a certain CP-local distance order are discarded.
3) Finally, a modified CW-AVD function is applied to the remaining pixels of the window to find the filtering output, using a central weighting scheme different from that of the CWVM filter.

Denoting $\mathbf{W}(\mathbf{c})$ as a filter window centered at coordinate $\mathbf{c}$, the Euclidian distances between the CP and all N pixels of the window are given by

$$d_j = \|\mathbf{x}_j(\mathbf{c}) - \mathbf{x}(\mathbf{c})\|, \quad 1 \le j \le N \tag{3}$$

where $\mathbf{x}(\mathbf{c})$ represents the CP of the window, $\mathbf{x}_j(\mathbf{c}) \in \mathbf{W}(\mathbf{c})$ denotes a pixel of the window, and $\mathbf{x}(\mathbf{c}) \triangleq \mathbf{x}_{\frac{N+1}{2}}(\mathbf{c})$ as in the previous subsection. The d_js are ranked in an ascending order such that $d_{(1)} \le d_{(2)} \le \cdots \le d_{(i)} \le \cdots \le d_{(N)}$ (Note that the coordinate variable $\mathbf{c}$ is omitted for d_js and $d_{(i)}$s for simplicity). Accordingly, the pixels of the filter window are re-expressed as

$$\mathbf{W}'(\mathbf{c}) = \{\mathbf{x}_{\langle i\rangle}(\mathbf{c}) \mid i = 1, 2, \ldots N\} \tag{4}$$

where $\mathbf{x}_{\langle i\rangle}(\mathbf{c})$ denotes a pixel of the window, which is indexed by its CP-local distance ranking order, that is,

$$\mathbf{x}_j(\mathbf{c}) \to \mathbf{x}_{\langle i\rangle}(\mathbf{c}) \Longleftrightarrow d_j \to d_{(i)}. \tag{5}$$

where the symbol "$\to$" represents a mapping from an original value to its ranked version. Note that $\mathbf{W}'(\mathbf{c})$ and $\mathbf{W}(\mathbf{c})$ consist of the same set of pixels, but in different orders.

Based on the ranked pixel set $\mathbf{W}'(\mathbf{c})$, the output of the CWTVM filter with an integer central weight, k, is defined as

$$\hat{\mathbf{y}}_k(\mathbf{c}) = \mathbf{x}_{\langle m\rangle}(\mathbf{c}) \in \mathbf{W}'(\mathbf{c}) \mid m = \arg \min_{1 \le i \le N} \left\{ \hat{R}_i^k \right\} \tag{6}$$

where $\hat{R}_i^k$ is the modified CW-AVD function (or MCW-AVD) given by

$$\hat{R}_i^k = (k-1)\|\mathbf{x}_{\langle i\rangle}(\mathbf{c}) - \mathbf{x}(\mathbf{c})\| + \sum_{j=1}^{N-(k-1)} \|\mathbf{x}_{\langle j\rangle}(\mathbf{c}) - \mathbf{x}_{\langle i\rangle}(\mathbf{c})\|. \tag{7}$$

TABLE I
PERFORMANCE OF THE CWVM AND THE CWTVM FILTERS REGARDING DIFFERENT RATIOS OF RANDOM IMPULSE CORRUPTION, WHERE THE MSE IS MEASURED

Noise ratio	CWVM filter with $k =$				CWTVM filter with $k =$			
	1	2	3	4	1	2	3	4
2%	29.61	13.76	6.22	16.63	29.61	16.18	7.67	9.81
5%	31.28	15.23	10.92	62.69	31.28	18.33	11.81	38.77
10%	34.03	18.91	25.92	197.1	34.03	22.51	21.80	127.2
15%	36.95	22.74	49.81	358.5	36.95	25.14	35.36	232.4
20%	41.67	30.82	93.95	572.6	41.67	32.29	61.81	381.6
30%	53.05	55.58	225.6	1041.	53.05	50.70	135.7	717.9

The proposed CWTVM filter shares a number of properties with its counterpart, the CWVM filter. The CWTVM filter reduces to a standard VMF filter when $k = 1$, or an identity filter when $k \ge \frac{N+1}{2}$. With k increased in the range of $[1, (N+1)/2)$, the detail preservation of the CWTVM filter is improved whereas its noise suppression performance degraded. However, the CWTVM filter can achieve more robustness against impulse noise than what the CWVM filter does when the central weight increases, due to its novel CP-distance trimmed technique given in (7). Based on the random distribution of impulse noise, such a technique is able to trim away some noise even before the multivariate aggregation ordering is applied.

Evaluation of the CWTVM filter is conducted on a wide range of impulse corrupted natural color images. Table I lists the results of the CWTVM and CWVM filters on a 512×512-pixel 24-bit RGB image, *Lena*, corrupted by different ratios of random impulse noise, where a 3×3 filter window is used. Noticeable gains are observed when the CWTVM filter works with the central weight $k = 3$ or 4 and the noise ratio is relatively high. The mechanism leading to the performance improvement is analyzed in Example 2 of Appendix. Such an improvement will greatly benefit the local structure estimation and classification proposed in Section III. The performance of the CWTVM filter is slightly lower than that of the CWVM filter when the central weight $k = 1$ or 2. However, its influence on the proposed PBTVM filter is negligible as long as the CWTVM filter outputs reference estimates with reduced noise.

III. PARTITION-BASED VECTOR FILTERING STRUCTURE

A. Review of Switch-Based Vector Median Filters

A number of switch-based vector median filters [11], [13], [14] have been reported in recent years, which take the advantages of the CWVM filter to detect impulse corruptions and to preserve fine image structures. Given a filter window $\mathbf{W}(\mathbf{c})$ centered at the pixel $\mathbf{x}(\mathbf{c})$ containing N pixel samples, these filters first utilize a series of CWVM filters with their central weights $k = 1, 2, \ldots, (N-1)/2$, respectively, to obtain a set of reference estimates, $\{\mathbf{y}_k(\mathbf{c}) \mid k = 1, 2, \ldots, (N-1)/2\}$, from the pixels of the window, where $\mathbf{y}_k(\mathbf{c})$ denotes the output of a

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

CWVM filter with a central weight k. Then the distance between the CP and each of the reference estimates is measured as

$$e_k(\mathbf{c}) = \|\mathbf{y}_k(\mathbf{c}) - \mathbf{x}(\mathbf{c})\|, \quad k = 1, 2, \ldots, \frac{(N-1)}{2} \tag{8}$$

where $\|\cdot\|$ represents the Euclidean distance for the AVLUM filter [13] and the MWVM filter [11], or the directional distance for the SCWVDF/ACWVDF filter [14]. Finally, a group of preset thresholds, $\{\mathrm{Tol}_k \mid k = 1, 2, \ldots, (N-1)/2\}$, is used to evaluate the *irregularity* of the CP-reference distance $e_k(\mathbf{c})$s, and accordingly, a binary decision mechanism is activated to determine the filter output, $\hat{\mathbf{x}}(\mathbf{c})$, as follows [13], [14]

$$\hat{\mathbf{x}}(\mathbf{c}) = \begin{cases} \mathbf{x}(\mathbf{c}), & \text{if all } e_k(\mathbf{c}) \le \mathrm{Tol}_k, 1 \le k \le \frac{N-1}{2} \\ \mathbf{y}_m(\mathbf{c}), & m = \min\{k \mid e_k(\mathbf{c}) > \mathrm{Tol}_k\}. \end{cases} \tag{9}$$

Despite their state-of-the-art performance in impulse suppression and structure preservation, these switch-based vector median filters still have a few drawbacks.

1) The accuracy of their binary noise detection heavily relies on the threshold values (Tol_ks) and the condition (e.g., impulse noise model and image structure) of the threshold optimization. It is hard for these filters to cope with different impulse models or varying image characteristics.
2) Although their switch-based reconstruction mechanism works efficiently for the impulse noise removal, such a selective mechanism is inappropriate for suppressing other distinct types of noise, such as additive noise or mixed noise which contaminates original images.

B. Formulation of the PBTVM Filter

A new filtering framework is proposed in this section to eliminate the drawbacks associated with the existing switch-based vector median filters. The new filter, named hereafter as the partition-based trimmed vector median (PBTVM) filter, combines the robust CWTVM filter of Section II-B with the partition-based weighted filtering techniques developed in [40], [41], inheriting the merits from both techniques. A group of estimates are first produced by the CWTVM reference filter. Then, a structure classifier is formulated to map the inputs into one of the partition cells. Finally, a weighted filtering operation is activated to achieve the best noise suppression and image preservation for the classified structure. The new PBTVM filter involves neither binary noise detection nor selective filtering mechanism. The schematic diagram of the new filter is presented in Fig. 2. Details of the CP-reference based structure classification and the partition-based weighted filtering are described in the following sections.

1) CP-Reference-Based Structure Classification: Given a filter window, $\mathbf{W}(\mathbf{c})$, which centers at the CP, $\mathbf{x}(\mathbf{c})$, and contains N pixel samples, a series of CWTVM filters with their central weights $k = 1, 2, \ldots, K$, respectively, are used to produce a set of reference estimates, $\{\hat{\mathbf{y}}_k(\mathbf{c}) \mid k = 1, 2, \ldots, K\}$, where $\hat{\mathbf{y}}_k(\mathbf{c})$ is the CWTVM filtering output defined by (6), and $K = (N-1)/2$. After the reference filtering, the distance

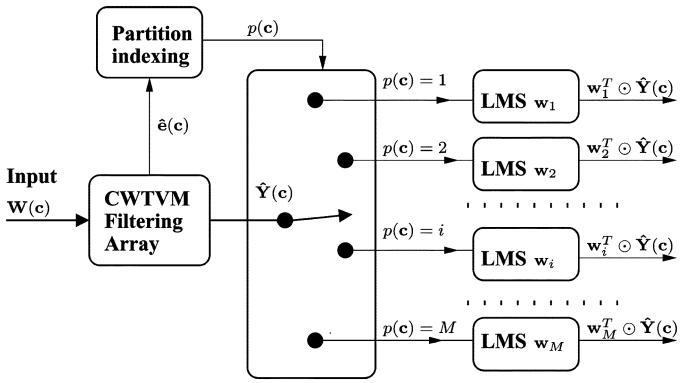

Fig. 2. Schematic diagram of the proposed PBTVM filter, where the difference vector. $\hat{\mathbf{e}}(\mathbf{c})$ is mapped into one of M mutually exclusive partitions to trigger an optimal weighted filtering, and the operator $\odot$ is defined in (15).

between each of the estimates and the CP of the window is measured as

$$\hat{e}_k(\mathbf{c}) = \|\hat{\mathbf{y}}_k(\mathbf{c}) - \mathbf{x}(\mathbf{c})\|, \quad k = 1, 2, \ldots, K \tag{10}$$

and a distance vector, $\hat{\mathbf{e}}(\mathbf{c})$, is formulated as follows:

$$\hat{\mathbf{e}}(c) = [\hat{e}_1(\mathbf{c}), \hat{e}_2(\mathbf{c}), \ldots, \hat{e}_K(\mathbf{c})]. \tag{11}$$

The distance vector $\hat{\mathbf{e}}(\mathbf{c})$ as a whole contains more local structure information than any of its component, $\hat{e}_k(\mathbf{c})$. In order to reveal most of the structured information in the vector $\hat{\mathbf{e}}(\mathbf{c})$, a structure classifier, $\mathcal{C}(\cdot)$, is formulated as a function of the vector $\hat{\mathbf{e}}(\mathbf{c})$. The classifier maps the inputs from an $\Re^K$ dimensional vector space to one of M mutually exclusive partition cells as follows:

$$\Psi_i = \{\hat{\mathbf{e}}(\mathbf{c}) \in \Re^K : \mathcal{C}[\hat{\mathbf{e}}(\mathbf{c})] = i\} \tag{12}$$

where the partition cells, $\{\Psi_i \mid i = 1, 2, \ldots, M\}$, satisfy $\bigcup_i \Psi_i = \Re^K$ and $\Psi_i \bigcap \Psi_j = 0$ for $i \neq j$.

Various partition methods, such as the scalar quantization (SQ) [42], the fuzzy quantization (FQ) [43], or the vector quantization (VQ) [44] can be used to construct the classifier. In this paper, the scalar quantization is selected due to its computational efficiency and high robustness. Given the distance vector $\hat{\mathbf{e}}(\mathbf{c})$, a group of decision levels (thresholds), $\{T_{k,l} \mid l = 0, 1, \ldots, L\}$, are defined for each of its components, $\hat{e}_k(\mathbf{c}) | 1 \le k \le K$. The levels are formulated in a monotonically ascending order so that $T_{k,l} \le T_{k,l+1}$ for any $0 \le l \le L-1$, where L is a preset value that controls the number of decision levels for each $\hat{e}_k(\mathbf{c})$, and accordingly, the total number of partition cells, i.e., $M = L^K$.

Following these definitions, the classifier $\mathcal{C}(\cdot)$ can be expressed by a partition index, $p(\mathbf{c})$, which is given by

$$p(\mathbf{c}) = 1 + \sum_{k=1}^{K} L^{(K-k)} \psi_k \tag{13}$$

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

where for $k = 1, 2, \ldots, K$

$$\psi_k = q, \text{ if } \hat{e}_k(\mathbf{c}) \in [T_{k,q}, T_{k,q+1}), \quad 0 \le q \le L-1. \quad (14)$$

An example is provided in Appendix to demonstrate the calculation of the partition index $p(\mathbf{c})$ for $K = 3$ and $L = 2$.

The proposed classification method is distinct from the noise detection methods used in the existing switch-based vector median filters [11], [13], [14]. In those filters, reference estimates are produced by the CWVM filters, and the CP-reference distances are compared with a set of thresholds individually to contribute a binary noise decision. However, in the proposed method, the new robust CWTVM filter is used for the reference estimate generation, and the classifier is formulated on a multivariate distance space which maps the local inputs to one of the partition cells. Moreover, each of these partition cells represents a specific local structure, which can be restored by an optimal weighted filtering operation, regardless of whether it is corrupted and how the corruption occurs.

2) Partition-Based Weighted Filtering: For the given filter window, $\mathbf{W}(\mathbf{c})$, all the reference estimates from the CWTVM filter, $\{\hat{\mathbf{y}}_k(\mathbf{c}) \mid k = 1, 2, \ldots, K\}$, and the input CP, $\mathbf{x}(\mathbf{c})$, are used to reconstruct the original CP value, $\hat{\mathbf{x}}(\mathbf{c})$, via a weighted filtering operation given by

$$\begin{aligned}\hat{\mathbf{x}}(\mathbf{c}) &= w_{i,0}\mathbf{x}(\mathbf{c}) + \sum_{k=1}^{K} w_{i,k}\hat{\mathbf{y}}_k(\mathbf{c}) \\ &\equiv \mathbf{w}_i^T \odot \hat{\mathbf{Y}}(\mathbf{c}) \end{aligned} \quad (15)$$

where "$\odot$" denotes an unconventional "dot" product between a row vector and a column vector. As shown in (15), each element of the row vector is a scaler while each element of the column vector is a 3×1 vector, and the number of elements (scalers) of the row vector must be the same as that of elements (3×1-column-vectors) of the column vector for "$\odot$" to apply. Between the elements of the two vectors, the product is a product between a scaler and a vector (or matrix) which has been defined in the conventional matrix algebra. The row vector

$$\mathbf{w}_i^T = [w_{i,0}, w_{i,1}, w_{i,2} \ldots, w_{i,K}] \quad (16)$$

is the weight vector for the filtering operation, where the value of $i = p(\mathbf{c})$ indicates the index to the classified partition cell, and the column vector

$$\hat{\mathbf{Y}}(\mathbf{c}) = \left[\mathbf{x}^T(\mathbf{c}), \hat{\mathbf{y}}_1^T(\mathbf{c}), \hat{\mathbf{y}}_2^T(\mathbf{c}), \ldots, \hat{\mathbf{y}}_K^T(\mathbf{c})\right]^T \quad (17)$$

consists of all inputs of the weighted filtering operation. To facilitate further discussion, the weight vector is rewritten as $\mathbf{w}_i = \left[w_{i,0}, \tilde{\mathbf{w}}_i{}^T\right]^T$, where $\tilde{\mathbf{w}}_i = [w_{i,1}, w_{i,2} \ldots, w_{i,K}]^T$ is the reduced weight vector. Accordingly, the inputs of the weighted filtering operation is rewritten as $\hat{\mathbf{Y}}(\mathbf{c}) = [\mathbf{x}^T(\mathbf{c}), \tilde{\mathbf{Y}}^T(\mathbf{c})]^T$, where $\tilde{\mathbf{Y}}(\mathbf{c}) = \left[\hat{\mathbf{y}}_1^T(\mathbf{c}), \hat{\mathbf{y}}_2^T(\mathbf{c}), \ldots, \hat{\mathbf{y}}_K^T(\mathbf{c})\right]^T$ is the reduced filtering input vector.

Local invariance constraint [39] is applied to the weight vector to guarantee an unbiased reconstruction, which leads to

$$w_{i,0} = 1 - \tilde{\mathbf{w}}_i^T \mathbf{1} \quad (18)$$

where $\mathbf{1} = [1, 1, \ldots, 1]^T$ is a $K \times 1$ unitary vector. As such, the output of the PBTVM filter defined by (15) is rewritten as

$$\hat{\mathbf{x}}(\mathbf{c}) = w_{i,0}\mathbf{x}(\mathbf{c}) + \tilde{\mathbf{w}}_i^T \odot \tilde{\mathbf{Y}}(\mathbf{c}). \quad (19)$$

Based on its flexible weight configurations, the proposed filtering structure can be of any between a highly nonlinear median filter and a linear weighted averaging filter, which provides an efficient solution to suppress distinct types of noise contaminations. The structure includes the switch-based vector median filters [11], [13] as a special case, provided that the filtering weights, w_ks, are switched between one and zero according to a binary noise detection. However, there are two aspects that render the new structure distinct from the existing adaptive vector filters [17], [26]. First, the inputs of the weighted filtering are the reference estimates from a series of robust CWTVM filters, while the inputs of the existing adaptive vector filters are pixel samples produced by a local multivariate ordering. Second, the proposed filtering weight vector is trained off-line and selected on-line according to the structure classification, while the weights of those adaptive filters are generated on-line by a robust statistics estimator or a fuzzy membership function preset for the entire image.

IV. Partition Threshold Formulation and Filtering Weight Training

A. Partition Threshold Formulation

From a theoretic point of view, the partitioning thresholds used in (14) and filtering weights related to each partition cell should be jointly optimized to achieve the best restoration performance. However, such a nonlinear coupled optimization process often incurs a high computational cost, and easily leads to local minima [44], [45]. In this paper, a decoupled optimization technique is developed to avoid aforesaid issues, which is depicted as follows.

1) For initialization, the total number of partition cells and the starting threshold values are specified using a heuristic approach and prior knowledge.
2) Based on the initialized partition thresholds, the filtering weights are updated using the constrained LMS algorithm described in Section IV-B and a training image contaminated by a specified noise source.
3) Given the updated filtering weights, the Simplex minimization method [46] is used to optimize the threshold values associated with each $\hat{e}_k(\mathbf{c})$ independently. The process terminates when all the threshold updating sizes are smaller than 0.5; else
4) Repeat from Step 2) using the updated thresholds.

It is clear that the threshold set obtained by the proposed optimization process theoretically may lead to a suboptimal solution. However, by tuning threshold values on a range of test im-

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

TABLE II
PROPOSED THRESHOLD, $T_{k,l}$, FOR THE PARTITIONING OF THE DISTANCE VECTOR SPACE OF A 3×3 FILTER WINDOW, WHERE $1 \leq k \leq 4$ AND $0 \leq l \leq 8$

k	Threshold index l								
	0	1	2	3	4	5	6	7	8
1	0	28.1	46.3	59.1	81.6	98.2	106.8	127.8	∞
2	0	9.3	24.8	40.8	56.3	81.1	90.0	116.2	∞
3	0	5.4	19.1	29.9	35.1	44.9	51.1	92.9	∞
4	0	2.6	12.9	15.4	21.1	31.3	42.5	54.7	∞

ages and noise corruption scenarios, it is found, as demonstrated in Section V, that such a decoupled optimization process can achieve a superior performance while maintaining a competitive computation efficiency.

Table II lists the threshold set obtained by the proposed optimization process using a 3×3 filter window ($N = 9$). The parameter $K \triangleq \frac{N-1}{2} = 4$, which is the maximum valid central weight of the CWTVM filter for the given filter window. The parameter L is preset to $(N-1) = 8$, so that the classifier has $(N-1)$ decision levels to match each reference estimate to the signal characteristics of all pixels in the window. Such a configuration leads to a total number of partition cells $M = L^K = 4096$. This partition threshold set is used in all our experiments without any further optimization. It is noted that other threshold values $(T_{k,l})s$ or a different partition cell number (M) may be available for the proposed PBTVM filter with a 3×3 filter window, and the partition threshold set for the PBTVM filter with a 5×5 filter window size can be obtained using the proposed formulation technique as well. Nevertheless, all the experiments in Section V suggest that the PBTVM filter with the partition threshold set of Table II and a 3×3 filter window, if implemented recursively, is able to achieve satisfactory performance under most of noise corruption conditions.

B. Constrained LMS Weight Training

For a given filter window $\mathbf{W}(\mathbf{c})$, the filtering weight set, $\mathbf{w}_i$, which relates to the classified structure partition cell Ψ_i, is trained to minimize the *mean-square error* (MSE) given by

$$J = E[||\mathbf{s}(\mathbf{c}) - \hat{\mathbf{x}}(\mathbf{c})||^2] \tag{20}$$

where $E[\cdot]$ is the expectation operator, $\mathbf{s}(\mathbf{c})$ and $\hat{\mathbf{x}}(\mathbf{c})$ denote, respectively, the original and the reconstructed pixel value by the PBTVM filter at coordinate $\mathbf{c}$. Since all the partition cells, $\{\Psi_i \mid i = 1, 2, \ldots, M\}$, are mutually exclusive to each other, the minimization of the overall MSE can be equivalently achieved by the independent MSE minimization with respect to each partition cell Ψ_i over the entire test image, that is

$$J_i = E[||\mathbf{s}(\mathbf{c}) - \hat{\mathbf{x}}(\mathbf{c})||^2 \mid p(\mathbf{c}) = i] \tag{21}$$

where $p(\mathbf{c})$ is the partition index defined by (13). Following the steps similar to [39] and [40], the updating procedure for the weight coefficients, $\mathbf{w}_i = [w_{i,0}, \tilde{\mathbf{w}}_i{}^T]^T$, at the nth iteration is given by

$$\tilde{\mathbf{w}}_i^{n+1} = \tilde{\mathbf{w}}_i^n + 2\lambda_i(\mathbf{I} \otimes [\mathbf{s}(\mathbf{c}) - \hat{\mathbf{x}}(\mathbf{c})]^T)\mathbf{D}(\mathbf{c}) \tag{22}$$

$$w_{0,i}^{n+1} = 1 - \mathbf{1}^T \tilde{\mathbf{w}}_i^{n+1} \tag{23}$$

where $i = p(\mathbf{c})$ is the index to the classified partition cell, $\mathbf{I}$ is a $K \times K$ identity matrix, $\otimes$ denotes the Kronecker (or Tensor) product [49], $\mathbf{1} = [1, 1, \ldots, 1]^T$ is a $K \times 1$ unitary vector, and

$$\mathbf{D}(\mathbf{c}) = \begin{bmatrix} \mathbf{d}_1(\mathbf{c}) \\ \mathbf{d}_2(\mathbf{c}) \\ \vdots \\ \mathbf{d}_K(\mathbf{c}) \end{bmatrix} = \tilde{\mathbf{Y}}(c) - (\mathbf{1} \otimes \mathbf{x}(\mathbf{c})) \tag{24}$$

is a column vector consisting of all the difference vectors between the CP and the reference estimates from the $\tilde{\mathbf{Y}}(c)$ of (19), so that each element of $\mathbf{D}(\mathbf{c})$ can be expressed as $\mathbf{d}_k(\mathbf{c}) = \hat{\mathbf{y}}_k(\mathbf{c}) - \mathbf{x}(\mathbf{c})$ for $1 \leq k \leq K$.

In order to obtain a steady convergence in a nonstationary environment such as image processing applications, the step-size parameter in (22) has to be adaptive to and, therefore, normalized by the input energy [39]. As a result, the parameter λ_i at the nth iteration is given by

$$\lambda_i = \frac{\lambda_0}{||\mathbf{D}(\mathbf{c})||^2} = \frac{\lambda_0}{\sum_{k=1}^{K} ||\mathbf{d}_k(\mathbf{c})||^2} \tag{25}$$

where $0 \leq \lambda_0 \leq (2/3)$ is a constant to guarantee the sufficient condition of a strict convergence.

Extensive tests have confirmed that the proposed weight training algorithm can steadily converge to a desired stopping criterion in a moderate number of iterations. Fig. 3 demonstrates the results with three 256×256-pixel 24-bit RGB images, *Lena*, *Peppers*, and *Parrots*, corrupted by 10% random impulse noise. Given a stopping criterion, $\triangle\text{MSE} \leq 0.01$, the training algorithm approachs a steady solution within, at most, 60 iterations. The tests with other corruption scenarios, although not presented in this paper, have demonstrated a very similar convergence trend.

The dependence between the error criterion (MSE) and the parameter λ_0 is also evaluated through the tests. Fig. 4 shows the results with *Lena, Peppers*, and *Parrots* corrupted by 5% and 10% impulse noise, respectively, where the value of λ_0 covers the entire range of $(0, 2/3]$ with an increment of 0.001. The optimal value of λ_0 always lies in the range of $[0.002, 0.05]$. Thus, a suboptimal value, $\lambda_0 = 0.02$, is used throughout our training and testing process.

C. Recursive Filtering and Weight Training

The recursive implementation of weighted nonlinear filters has been examined in earlier published work [40], [41]. As these filtering structures are able to adapt the weighting operation to the input values, better noise suppression is generally achieved

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

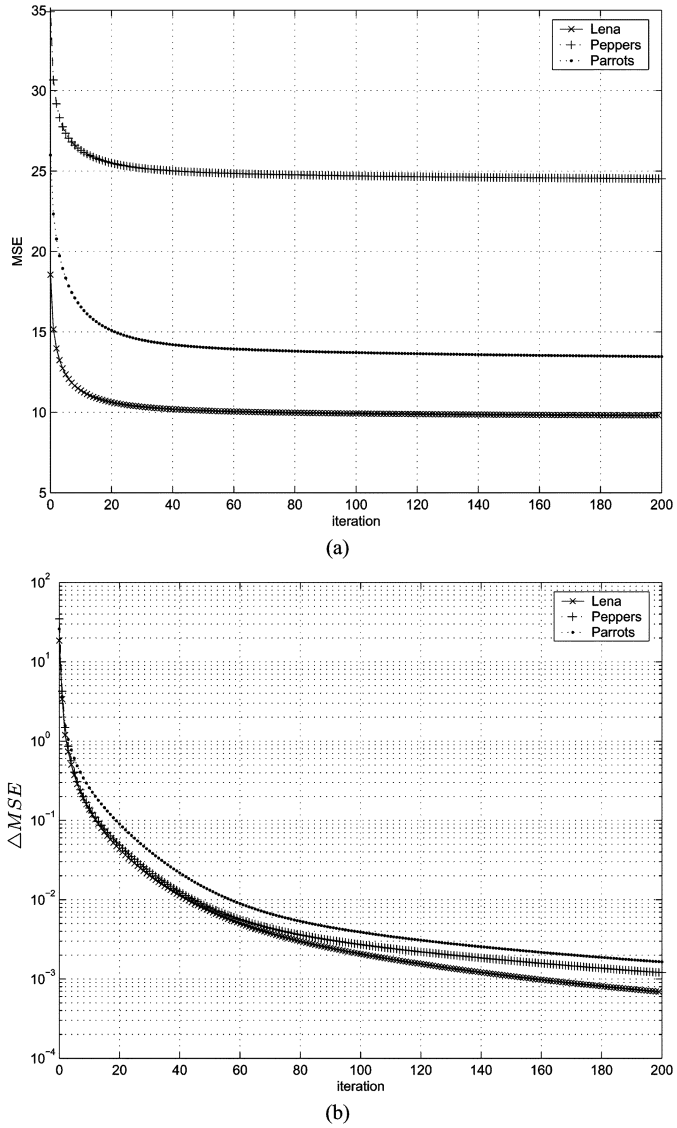

Fig. 3. Convergence evaluation of the proposed training algorithm using test images *Lena, Peppers, Parrots* corrupted by 10% random impulse noise. (a) MSE versus iteration. (b) Updating error versus iteration.

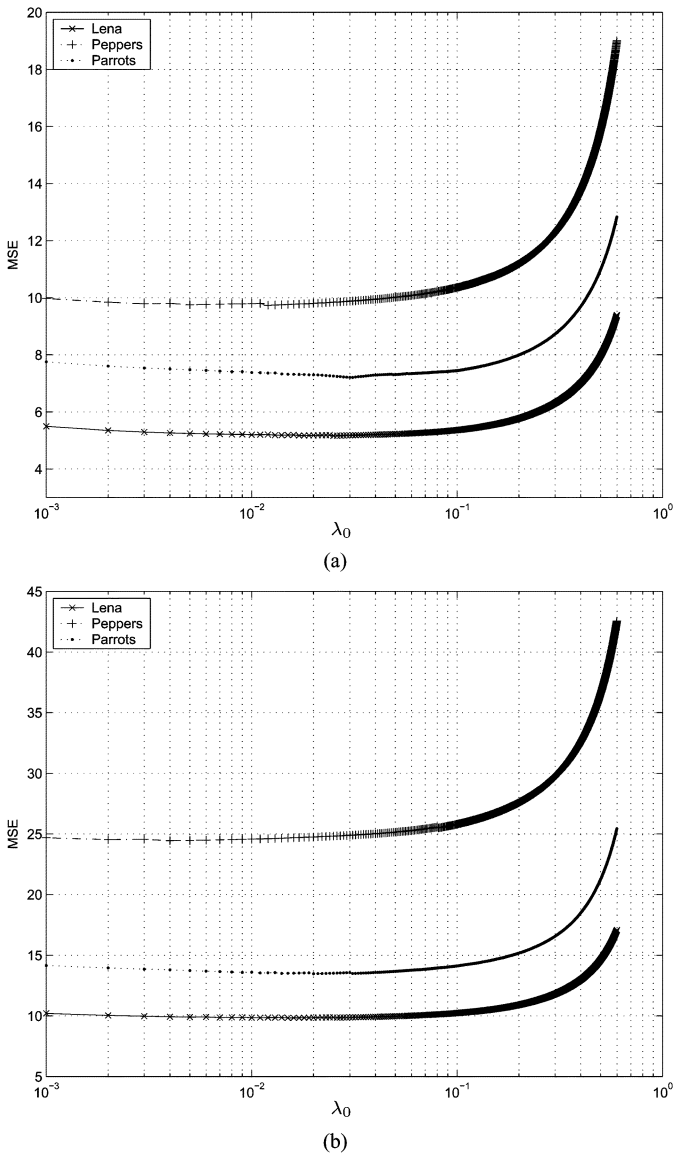

Fig. 4. Dependence of the error measurement (MSE) on the training step parameter λ_0 using test images *Lena, Peppers*, and *Parrots* corrupted by 5% and 10% impulse noise, respectively. (a) 5% random impulse noise. (b) 10% random impulse noise.

without introducing excessive blurring artifacts. In this subsection, a recursive implementation of the proposed PBTVM filter is formulated to achieve better restoration performance. Given a filter window, $\mathbf{W}(\mathbf{c})$, containing N pixel samples and denoting $K = (N-1)/2$, the input set of the recursive PBTVM filter is given in a lexicographic scan order as follows:

$$\hat{\mathbf{W}}(c) = \{\hat{\mathbf{x}}_1, \hat{\mathbf{x}}_2, \ldots, \hat{\mathbf{x}}_{K-1}, \mathbf{x}_K, \mathbf{x}_{K+1}, \ldots, \mathbf{x}_N\} \qquad (26)$$

where $\{\hat{\mathbf{x}}_k \mid k = 1, 2, \ldots, K-1\}$ are the outputs of the recursive PBTVM filter at previous pixel positions, and $\{\mathbf{x}_k \mid k = K, K+1, \ldots, N\}$ are the inputs directly from the current filter window $\mathbf{W}(\mathbf{c})$. Given such an input set $\hat{\mathbf{W}}(\mathbf{c})$, the implementation of the recursive PBTVM filter is very similar to its nonrecursive counterpart. In other words, the $\hat{\mathbf{W}}(\mathbf{c})$ is input into the CWTVM filter to produce the reference estimates through (3)–(7), then the reference estimates are fed into the structure classification via (10)–(14), and the filtering output is finally produced by the partition-based weighted operation given by (15)–(18).

Although the recursive PBTVM filter achieves a better overall performance, it precludes a closed-form analytical solution to its weight training process. This is because its input set, $\hat{\mathbf{W}}(\mathbf{c})$, substantially depends on the previous outputs of the filter. A heuristic weight training strategy is developed here as an alternative, which is able to satisfy the weight training of the recursive PBTVM filter, even though many assumptions of a steady convergence do not necessarily hold. Given the weight set at the tth iteration as $\mathcal{W}^t = \{\mathbf{w}_1^t, \mathbf{w}_2^t, \ldots, \mathbf{w}_M^t\}$, where $\mathbf{w}_i^t$ is the weight vector of the partition cell Ψ_i, and $i = 1, 2, \ldots, M$, the proposed weight training strategy is given as follows.

1) At the first iteration, the weight set is updated by the training algorithm defined in (22)–(25) and using the

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

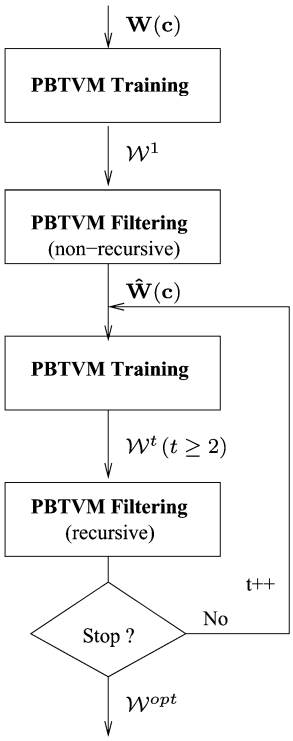

Fig. 5. Schematic diagram of the proposed recursive weight training strategy for the PBTVM filter.

nonrecursive input set $\mathbf{W}(\mathbf{c})$. The resultant weight set, denoted as $\mathcal{W}^1$, is then used by the PBTVM filter to produce the outputs, via (3)–(18).

2) At the second iteration, the outputs of step (1) is used to construct the input set $\hat{\mathbf{W}}(\mathbf{c})$ of (26). This recursive input set is then used to update the weight set, via the training algorithm given by (22)–(25).
3) For any iteration $t > 2$, the outputs of the PBTVM filter at iteration $(t-1)$ and the weight set $\mathcal{W}^{t-1}$ are used to construct the current recursive input set $\hat{\mathbf{W}}(\mathbf{c})$. This input set is then used for further weight set updating via the training algorithm given by (22)–(25).

Fig. 5 depicts the schematic diagram of the recursive weight training strategy. Despite its intractability in mathematic analysis, the proposed recursive training strategy always converges to solutions with a performance better than that of its nonrecursive counterpart in extensive tests conducted on real image data. Fig. 6 demonstrates the convergence of the recursive weight training strategy on several test images corrupted by different degrees of impulse noise, where the MSE values given at the first iteration correspond to those by the nonrecursive PBTVM implementation. The performance improvements of such a recursive implementation are significant when images are highly corrupted, while they are diminishing, in comparison with its nonrecursive counterpart, when the images become less corrupted by noise. Moreover, satisfactory results are always achieved at iteration $t = 2$. Therefore, the PBTVM filter is implemented recursively in all the following experiments.

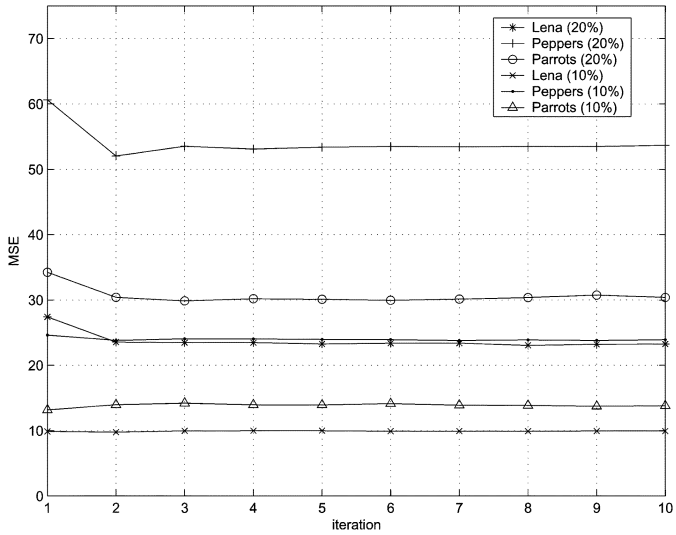

Fig. 6. Convergence plots of the recursive training strategy using test images *Lena, Peppers*, and *Parrots* corrupted by different degrees of random impulse noise.

V. Simulation Results

The proposed PBTVM filter has been evaluated by a extensive range of tests, and its performance is compared with a number of prior-art filtering techniques in the area of color image restoration. The tests and analysis are summarized as follows:

1) a performance comparison of the PBTVM filter with other prior-art impulse filtering techniques with respect to different levels of impulse corruptions;
2) a robustness assessment of the the PBTVM filter with respect to the image or noise characteristics variation in its weight training and filtering operation;
3) a performance comparison of the PBTVM filter with other state-of-the-art adaptive filters in suppressing additive noise or mixed noise corruption; and
4) a computational complexity analysis of the PBTVM filter framework.

Several objective criteria are used in the tests to measure the distortion in filtering outputs, which include the MSE, the mean absolute error (MAE), and the normalized color difference (NCD). The MSE and MAE are defined in the RGB color space and given as follows [12], [14]:

$$\mathrm{MSE} = \frac{1}{\mathrm{mHW}} \sum_{\mathbf{c} \in \mathbf{C}} \|\mathbf{s}(\mathbf{c}) - \hat{\mathbf{x}}(\mathbf{c})\|_2^2 \tag{27}$$

$$\mathrm{MAE} = \frac{1}{\mathrm{mHW}} \sum_{\mathbf{c} \in \mathbf{C}} \|\mathbf{s}(\mathbf{c}) - \hat{\mathbf{x}}(\mathbf{c})\|_1 \tag{28}$$

where $\mathbf{s}(\mathbf{c})$ and $\hat{\mathbf{x}}(\mathbf{c})$ are, respectively, the original and the restored pixel at pixel coordinate $\mathbf{c} \in \mathbf{C}$; W and H denote the width and the height of the test image, respectively; and m represents the number of color channels. Please note that $\|\cdot\|_2$ denotes the l_2 norm (Euclidean distance) and $\|\cdot\|_1$ denotes the

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

l_1 norm (city-block distance). The NCD measures the color distortion in the perceptual uniform CIELUV color space and is defined by [6], [14]

$$\mathrm{NCD} = \frac{\sum_{\mathbf{c}\in\mathbf{C}} \|\triangle E_{\mathrm{LUV}}(\mathbf{c})\|_2}{\sum_{\mathbf{c}\in\mathbf{C}} \|E_{\mathrm{LUV}}(\mathbf{c})\|_2} \tag{29}$$

where for a given pixel coordinate $\mathbf{c} \in \mathbf{C}$ in the CIELUV color space, $E_{\mathrm{LUV}}(\mathbf{c})$ and $\triangle E_{\mathrm{LUV}}(\mathbf{c})$ represent, respectively, the magnitude of the original pixel, and the difference between the original pixel and its reconstruction.

A. Impulse Noise Suppression

In terms of the impulse noise suppression, the impulse noise model proposed by [12], [14], [16] is used in the tests to simulate the impulse corruption present in natural color images, which is given by

$$\mathbf{x}(\mathbf{c}) = \begin{cases} \mathbf{v}(\mathbf{c}), & \text{with probability } p_v \\ \mathbf{s}(\mathbf{c}), & \text{with probability } 1 - p_v \end{cases} \tag{30}$$

where $\mathbf{s}(\mathbf{c})$ and $\mathbf{x}(\mathbf{c})$ denote, respectively, the original pixel and the sample pixel at a pixel coordinate $\mathbf{c}$, p_v denotes the noise ratio (the percentage of the corrupted pixels in a given image), and $\mathbf{v}(\mathbf{c})$ is a noisy version of $\mathbf{s}(\mathbf{c})$, who suffers from impulse corruption to, at least, one of its RGB components. For 24-bit RGB test images, when the impulse amplitudes take on the value of 0 or 255 with an equal probability, the impulse noise is known as the *pepper-and-salt* (PS) impulse. However, if the impulse amplitudes are distributed randomly in the range of $[0, 255]$, a more generalized type of impulse noise is generated and named as the *random* impulse noise. Such a random impulse noise is used in this section for all the tests.

All the impulse corruptions are generated according to the model in (30), using the random impulse noise and a noise ratio p_v varied from 0% to 30%. A two-step process [8], [26], [47] is applied to simulate the channel correlation of the impulse corruption. Given the preset noise ratio p_v, each RGB component of the test image is corrupted independently by the random impulse noise. Then a factor $\rho = 0.5$ is used to simulate the channel correlation for each corrupted pixel, that is, if a pixel is corrupted by random impulses on at least one of its components, then its noise-free components have a 50% probability to be corrupted, as well.

The evaluations of impulse suppression are conducted on three 256×256-pixel 24-bit RGB images, *Lena, Peppers*, and *Parrots*, which have been widely used by prior-art impulse filtering techniques [12]–[16] for their representive color characteristics and image structures. Two classes of vector filters are involved in the impulse suppression tests. The first includes classic vector filters such as the VMF [7], the generalized vector directional filter (GVDF) [8], and the DDF [9]. The second is the state-of-the-art techniques recently developed for the impulse suppression, which includes the self-adaptive algorithm (SAA) [19], the AVMF [12], the AVLUM [13], the histogram-based trimmed median (HBTM) filter [22], the SCWVDF/ACWVDF [14], and the selection weighted vector directional filter (SWVDF) [15]. The tested filters are implemented with a 3×3-pixel window in all the tests, which slide from pixel to pixel in a raster scanning fashion.

Experiment 1—Performance Within the Training Set: This experiment is designed to assess the performance of the vector filters aforementioned in restoring different levels of impulse corruptions. Parameter settings as proposed in the original works are used by the AVLUM, the HBTM, the AVMF, the ACWVDF/SWVDF during the tests. The noise ratio estimator is bypassed in the SAA filter and the actual noise ratio is used instead. The proposed PBTVM filter is applied within the training set, that is, the same image and noise ratio to be filtered are used in the off-line weight training of the filter.

Figs. 7–9 present the comparative results of restoring the test images *Lena, Peppers*, and *Parrots* corrupted by random impulse varied from 0% to 20%. The selected noise range enables a critical examination of the detail-preserving performance on images corrupted by low noise intensities, while still allowing a sufficient noise level for the performance consistency evaluation. The prior-art ACWVDF/SWVDF performs well at low noise ratios. However, its performance is degraded dramatically when the corruption level is increased, and is even inferior to classic vector filters such as the VMF or the DDF for serious noise corruptions with $p_v \geq 15\%$. In contrast, the PBTVM filter has demonstrated a quite consistent improvement over other filters in a wide range of noise ratios and image structures. The improvement is most significant in Fig. 7 in terms of the MSE criterion and with the image *Parrots* which contains high details and specific spectral characteristics. In Figs. 7–9, where the MAE and NCD criteria are used, the PBTVM filter shows a very noticeable improvement over other filters in most cases. Please note that the results with the VMF and the DDF are not shown in Figs. 7–9 because their MSE/NCD ranges are well beyond the display scales of the figures. The restoration results for noise corruptions with $p_v > 20\%$, although not shown in the current figures, have shown a very similar trend in the experiment.

Experiment 2—Performance Outside the Training Set: The robustness of a given filter in processing corrupted images outside of its training set is very important for any real restoration application. In order to assess the robustness of the PBTVM filter, three specific types of *outside training* modes are formulated as follows.

1) *Type-a*: The same test image but different noise corruptions are used in the weight training and the filtering operation.
2) *Type-b*: Different test images but the same noise corruption are used in the weight training and the filtering operation.
3) *Type-c*: Different test images and different noise corruptions are used in the weight training and the filtering operation.

Table III presents the experiment results on two test images, *Peppers* and *Parrots*, where the performance of the PBTVM filter working in all three outside training modes are evaluated. For *Type-a* PBTVM filter (PBTVM-a), the filtering weights are trained for a 15% random impulse corruption on the image to be restored in the filtering operation. For *Type-b* PBTVM filter (PBTVM-b), the filtering weights are trained using a different test image, *Lena*, corrupted by the same noise as that used in the filtering operation. The filtering weights in *Type-c* PBTVM filter

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

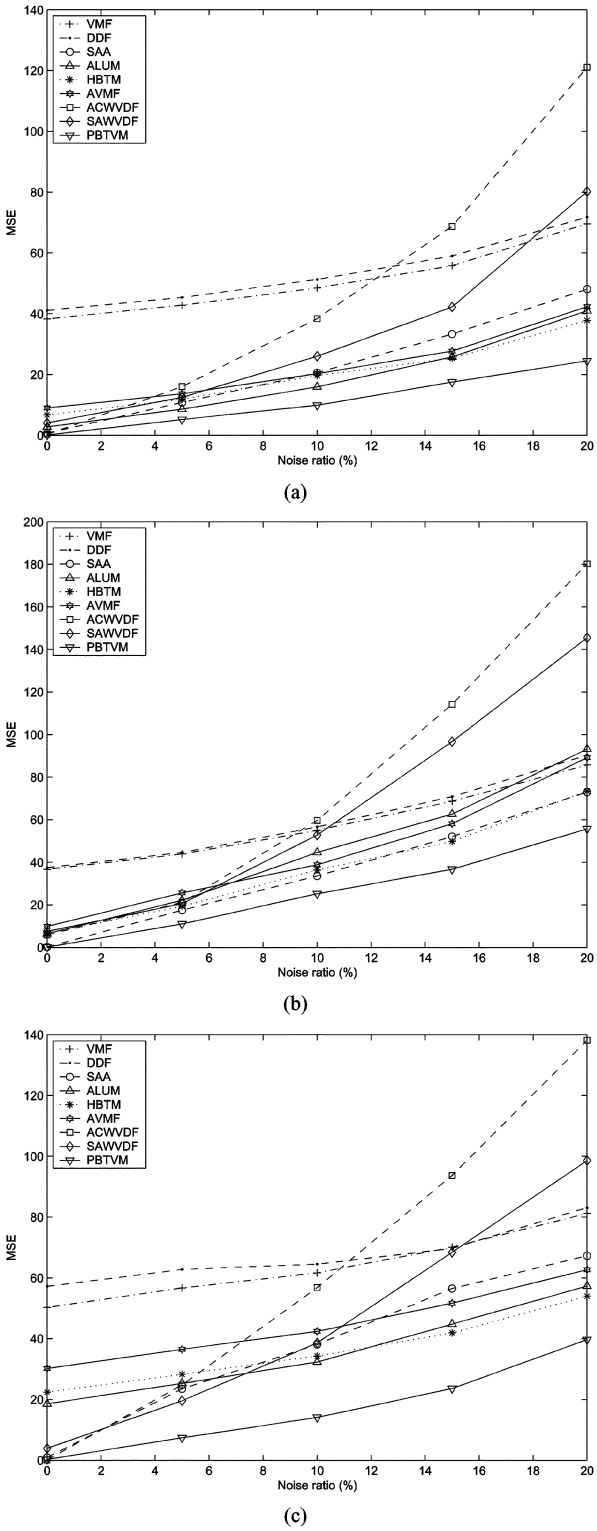

Fig. 7. Performance of the filters in terms of the MSE, using different test images and impulse noise ratios. (a) *Lena*. (b) *Peppers*. (c) *Parrots*.

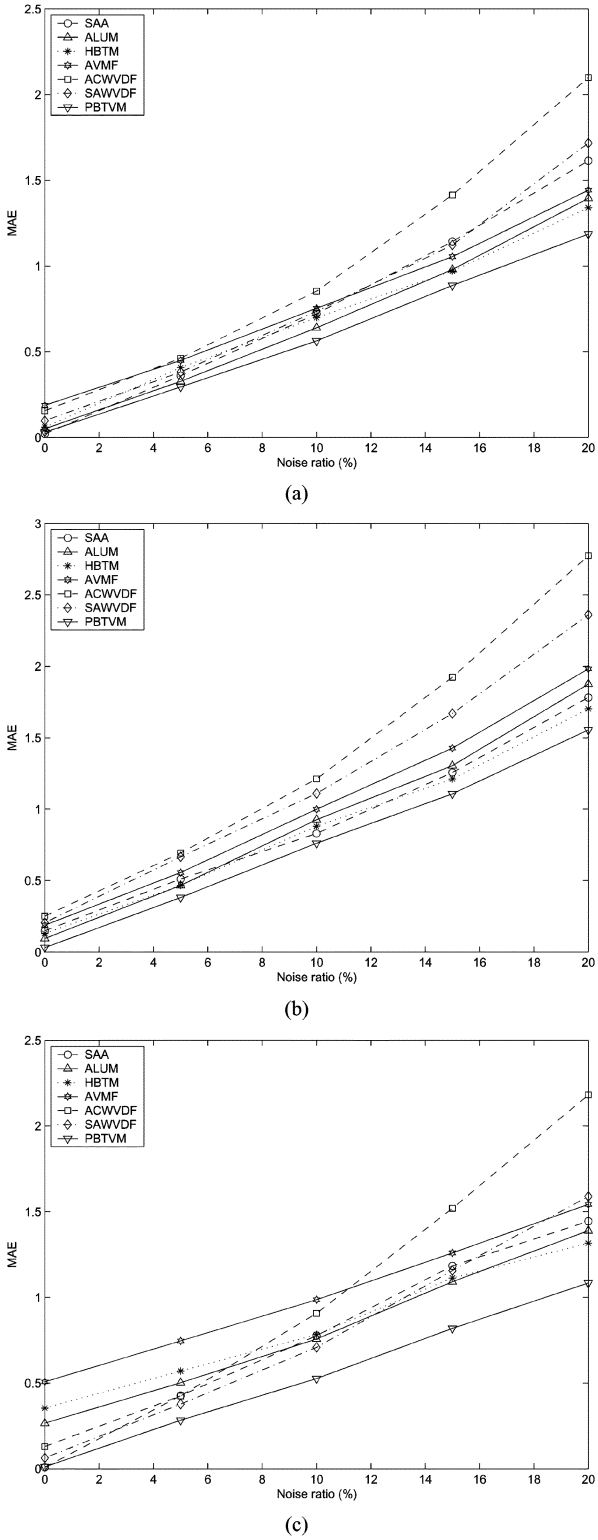

Fig. 8. Performance of the filters in terms of the MAE, using different test images and impulse noise ratios. (a) *Lena*. (b) *Peppers*. (c) *Parrots*.

(PBTVM-c) are trained using the test image *Lena* with a 15% random impulse corruption. Performance of the most relevant state-of-the-art techniques, which include the SAA filter (with the actual noise ratio), the SWVDF (with parameter $p = 0.5$, trained using the test image *Lena* and a 10% random impulse corruption), and other filters with their recommended parameter settings, are also listed in the table for a comparison. Moreover, the results of the PBTVM filter with ideal training (or *within the training set* as mentioned previously) are also attached in the last row of the table for readers' reference.

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

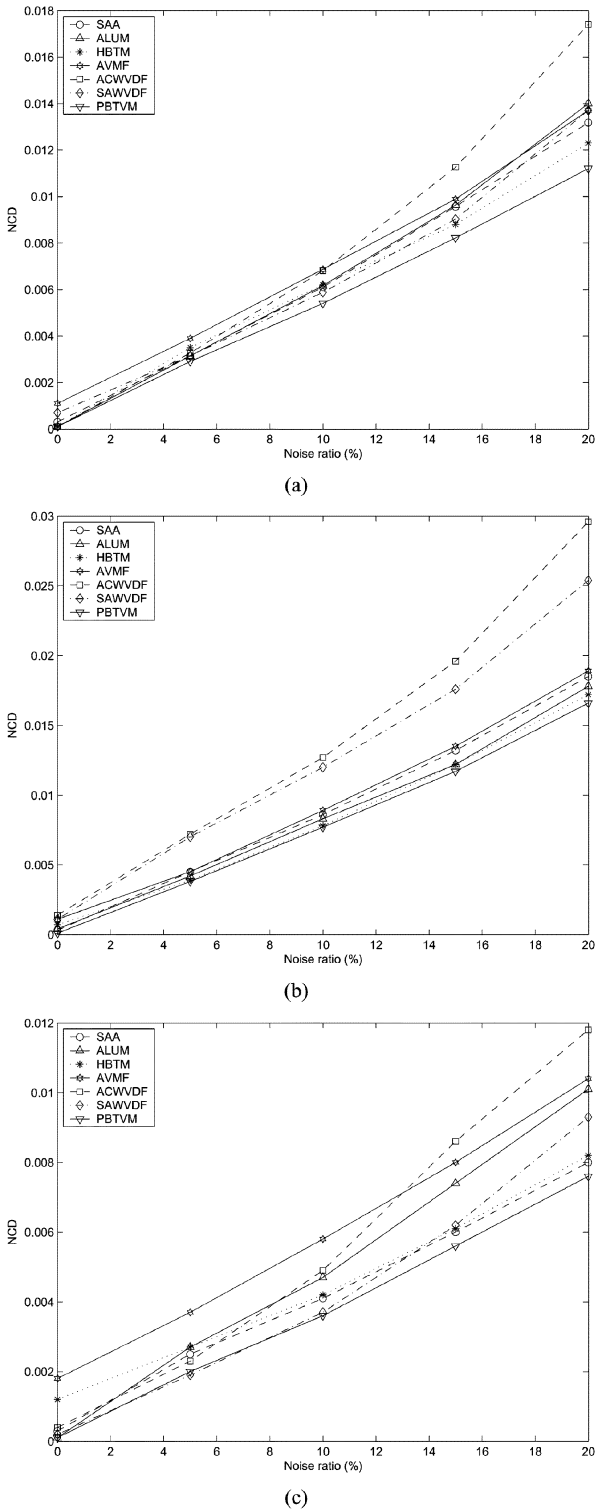

Fig. 9. Performance of the filters in terms of the NCD, using different test images and impulse noise ratios. (a) *Lena*. (b) *Peppers*. (c) *Parrots*.

An important observation from Table III is that the PBTVM filter works quite robustly throughout all the corruption scenarios, despite the fact that the filtering weights used to restore the corrupted images are actually outside the training set. In most of the cases, the PBTVM filter in all three modes has produced noticeable improvements over other filtering techniques in terms of most, if not all, of performance criteria and measures. Negligible performance degradations are observed between the PBTVM filter in three test modes and the filter by the ideal training. The PBTVM filter in *Type-a* mode has, in general, achieved the best performance over other two modes. Even the PBTVM filter in *Type-c* mode, where both the training image and the training corruption are different from the corrupted images to be restored, still achieves a better performance in most scenarios in comparison with other vector filters. Extensive performance evaluations and results with other test images and noise contamination scenarios, although not shown in this paper, have reached a very similar conclusion.

Besides the objective performance improvements, the proposed PBTVM filter also achieves a consistently better perceptual quality than other impulse filtering techniques. Fig. 10 demonstrates the reconstructions of several state-of-the-art filtering techniques for the image *Parrots*, where a 5% random impulse corruption is selected to reveal the detail preservation capability of the filters with respect to low noise intensity corruptions, and the filtering weights of the PBTVM filter are trained off-line on image *Lena* corrupted by a 15% random impulse. It is observed that the PBTVM filter provides an almost noise free reconstruction, whilst the SCWVDF and the SAA filter still have noticeable impulses left in the background area between two birds. Moreover, the reconstruction of the PBTVM filter has well-preserved stripe structures around the eye area, which is almost as the same as, or even better than those by the SCWVDF and the SAA techniques. The subjective quality of the new filter can be further improved if a more similar image or noise corruption model is used in the weight training process.

B. Restoration of Other Noise Contamination

In addition to its superior performance in impulse noise rejection, the new filter also works satisfactorily in suppressing the Gaussian noise as well as the mixed Gaussian and impulse noise. The noise models defined in [6], [30], [47] are used in the experiments to simulate corruptions. The Gaussian noise contamination is simulated by adding in each component of image pixels an random noise which follows a zero mean Gaussian distribution and a preset standard deviation, σ. The mixed noise contamination is simulated by a Gaussian noise contamination followed by a random impulse corruption of (30). A group of prior-art techniques for the Gaussian and mixed noise restorations have been involved in the performance comparison, which include the adaptive hybrid multivariate (AHM) filter [4], the ANNF [17], the AVDF [18], and the fuzzy vector median filter (FVMF) [26]. Classic vector filters, such as the VMF [7] and the GVDF [8], are also included in the test for references and benchmarks. A 3×3 filter window size has been used in all the cases.

Table IV summarizes the results on two 256×256-pixel RGB test images, *Peppers* and *Parrots*. Four corruption scenarios are listed in the table, where G10 and G20 denote two types of the Gaussian noise contaminations, respectively, with the standard deviation $\sigma = 10$ and $\sigma = 20$, whilst G10I2 and G20I5 represent two types of the mixed noise contaminations, the former being mixed by the Gaussian noise ($\sigma = 10$) and the random impulse noise ($p_v = 2\%$), the latter by the Gaussian noise

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

TABLE III
IMPULSE SUPPRESSION PERFORMANCE OF THE NEW FILTER COMPARED WITH OTHER TECHNIQUES, WHERE THE PBTVM FILTER, WORKING OUTSIDE THE TRAINING SET AND WITH A 3×3 WINDOW SIZE, IS USED. (a) COLOR IMAGE *PEPPERS* CORRUPTED BY DIFFERENT LEVELS OF RANDOM IMPULSE. (b) COLOR IMAGE *PARROTS* CORRUPTED BY DIFFERENT LEVELS OF RANDOM IMPULSE

(a)

Filters	5%			10%			20%		
	MSE	MAE	NCD	MSE	MAE	NCD	MSE	MAE	NCD
SAA [19]	17.5	0.511	0.0045	33.5	0.829	0.0086	73.0	1.782	0.0185
AVLUM [13]	22.1	0.465	0.0042	44.6	0.926	0.0083	93.2	1.875	0.0178
HBTM [22]	19.3	0.466	0.0039	36.4	0.880	0.0078	73.4	1.702	0.0172
AVMF [12]	25.6	0.556	0.0045	38.8	0.999	0.0089	89.2	1.982	0.0189
ACWVDF [14]	21.1	0.693	0.0072	59.7	1.212	0.0127	180.2	2.774	0.0296
SCWVDF [14]	20.5	0.666	0.0070	52.8	1.110	0.0120	145.5	2.361	0.0254
SWVDF [15]	18.1	0.962	0.0142	43.1	1.381	0.0196	132.5	2.615	0.0314
PBTVM-a	14.8	0.472	0.0045	29.4	0.869	0.0089	64.8	1.748	0.0186
PBTVM-b	14.1	0.432	0.0039	29.2	0.822	0.0078	72.1	1.747	0.0179
PBTVM-c	16.8	0.476	0.0044	32.5	0.869	0.0082	65.4	1.652	0.0167
PBTVM	11.0	0.381	0.0038	25.2	0.761	0.0077	55.7	1.555	0.0166

(b)

Filters	5%			10%			20%		
	MSE	MAE	NCD	MSE	MAE	NCD	MSE	MAE	NCD
SAA [19]	23.5	0.426	0.0025	38.1	0.776	0.0041	67.3	1.444	0.0076
AVLUM [13]	25.3	0.502	0.0027	32.3	0.757	0.0047	57.3	1.389	0.0101
HBTM [22]	28.3	0.570	0.0027	34.3	0.783	0.0042	54.0	1.316	0.0082
AVMF [12]	36.5	0.746	0.0037	42.4	0.986	0.0058	62.7	1.543	0.0104
ACWVDF [14]	24.8	0.425	0.0023	56.7	0.907	0.0049	138.2	2.181	0.0118
SCWVDF [14]	19.6	0.377	0.0019	38.6	0.708	0.0037	98.6	1.589	0.0093
SWVDF [15]	18.5	0.745	0.0033	29.8	1.021	0.0049	104.8	1.857	0.0113
PBTVM-a	14.4	0.444	0.0025	21.8	0.674	0.0040	39.7	1.201	0.0078
PBTVM-b	17.2	0.467	0.0025	24.8	0.711	0.0041	41.7	1.249	0.0079
PBTVM-c	20.5	0.514	0.0027	26.1	0.745	0.0042	42.2	1.253	0.0080
PBTVM	7.4	0.283	0.0020	14.1	0.526	0.0036	39.7	1.200	0.0079

($\sigma = 20$) and the random impulse ($p_v = 5\%$). The *outside training* mode is used by the PBTVM filter to assess its performance robustness regarding different images and noise distributions. Its filtering weights for the Gaussian noise suppression are trained off-line on another 256×256-pixel RGB test image, *Lena*, corrupted by a Gaussian noise with $\sigma = 20$, and the filtering weights for the mixed noise suppression are trained off-line using *Lena* image corrupted by the mixed noise G20I5.

Table IV presents the results of the test filters in restoring color images contaminated by distinct types of noise. For the

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

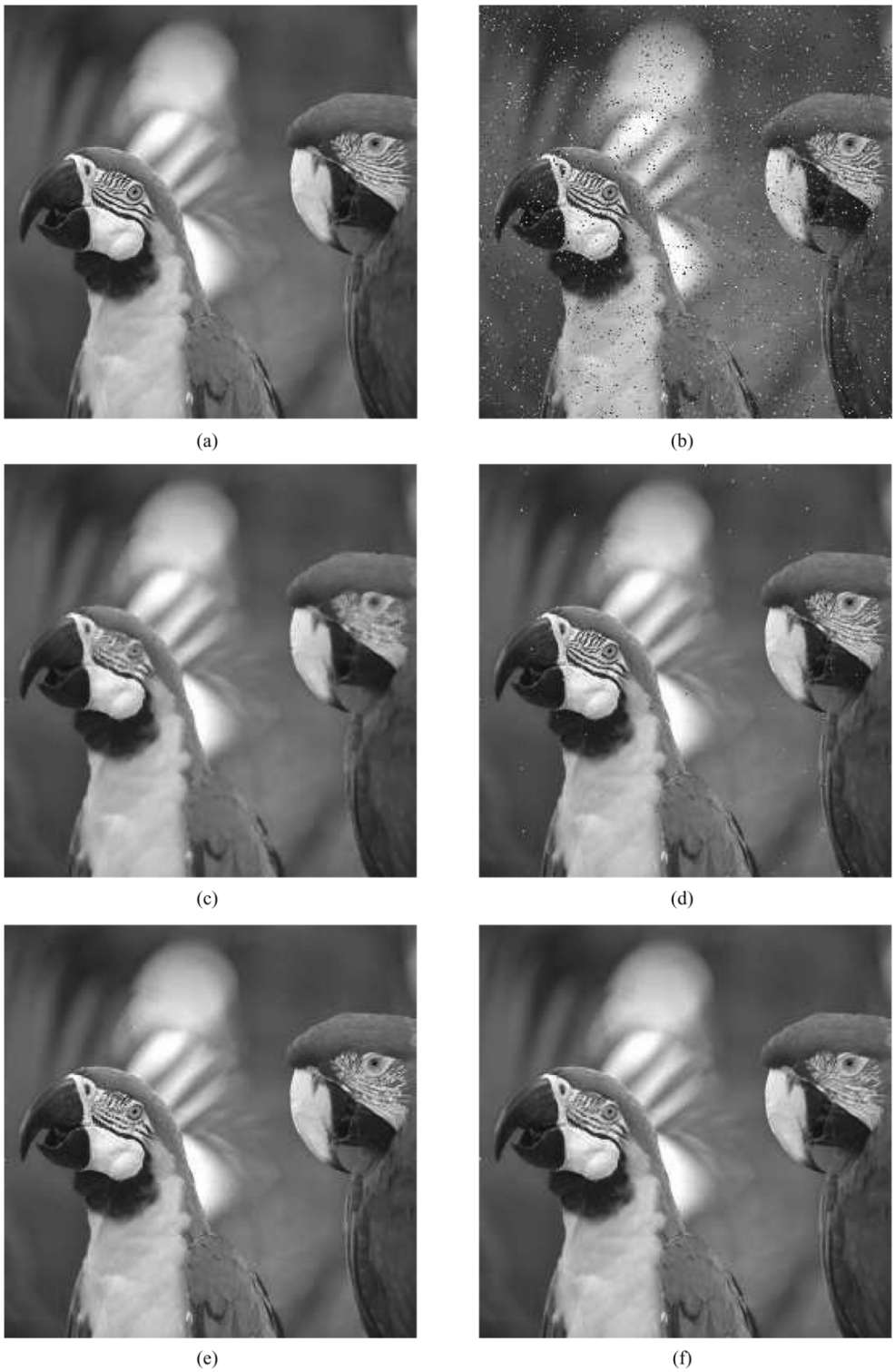

Fig. 10. Reconstruction of the proposed PBTVM filter compared with those by other techniques, where the test image *Parrots* is corrupted by the random impulse with $p_v = 5\%$. (a) Original image *Parrots*, (b) 5% random impulse corruption, (c) VMF output, (d) SCWVDF output, (e) SAA output, and (f) PBTVM output.

scenarios where the noise contaminations are the same as those used in weight training, the new filter achieve a considerable improvement over other techniques in terms of all the objective criteria. For the scenarios where the training condition and filtering condition are totally different, the new filter still delivers a robust performance, with its results slightly worse than the top performers (e.g., the ANNF) but is still quite comparable to those of the rest techniques. There are a few cases where the NCD measures of the PBTVM filter are slightly worse than those of prior-art techniques, but the new filter always achieves a better subject quality than its counterparts.

Fig. 11 highlights the image *Parrots* restored by the PBTVM filter and other prior-art techniques from a mixed noise contamination, G20I5. Blocks of color distortion remains in the output

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

TABLE IV
RESULTS OF RESTORING COLOR IMAGES CORRUPTED BY VARIOUS TYPES OF GAUSSIAN AND MIXED NOISE, WHERE A 3×3 WINDOW SIZE IS USED. (a) *PEPPER* CORRUPTED BY VARIOUS ADDITIVE AND MIXED NOISE. (b) *PARROTS* CORRUPTED BY VARIOUS ADDITIVE AND MIXED NOISE

(a)

Filters	G10			G20			G10I2			G20I5		
	MSE	MAE	NCD	MSE	MAE	NCD	MSE	MAE	NCD	MSE	MAE	NCD
VMF [7]	72.1	5.911	0.0771	164.4	9.639	0.1334	76.6	6.030	0.0785	185.5	10.066	0.1381
GVDF [8]	144.1	6.588	0.0692	250.8	10.257	0.1146	145.7	6.578	0.0695	266.1	10.224	0.1151
AHM [4]	49.7	4.914	0.0637	105.7	7.682	0.1095	60.8	5.164	0.0677	147.8	8.506	0.1208
ANNF [17]	53.2	4.953	0.0606	101.9	7.567	0.1013	57.1	5.071	0.0618	114.0	7.820	0.1038
AVDF [18]	108.3	5.669	0.0623	160.8	8.025	0.0971	111.6	5.806	0.0637	185.2	8.597	0.1027
FVMF [26]	85.3	5.438	0.0629	139.4	8.196	0.1028	89.4	5.524	0.0638	157.3	8.425	0.1051
PBTVM	57.9	5.474	0.0693	93.8	7.203	0.0971	66.1	5.739	0.0731	109.6	7.568	0.1008

(b)

Filters	G10			G20			G10I2			G20I5		
	MSE	MAE	NCD	MSE	MAE	NCD	MSE	MAE	NCD	MSE	MAE	NCD
VMF [7]	82.4	5.592	0.0630	169.7	9.448	0.1194	85.0	5.684	0.0639	184.0	9.811	0.1230
GVDF [8]	110.4	5.983	0.0539	205.8	9.826	0.1011	108.1	5.911	0.0533	222.0	10.224	0.1151
AHM [4]	53.3	4.444	0.0491	102.5	7.281	0.0906	63.4	4.669	0.0520	148.2	8.143	0.1012
ANNF [17]	64.6	4.594	0.0461	110.0	7.354	0.0866	65.7	4.631	0.0462	114.7	7.459	0.0867
AVDF [18]	75.9	4.788	0.0437	120.6	7.346	0.0814	79.6	4.976	0.0451	141.6	8.039	0.0865
FVMF [26]	72.6	4.754	0.0469	127.5	7.726	0.0879	73.5	4.771	0.0469	157.3	8.425	0.1051
PBTVM	59.1	5.021	0.0524	80.9	6.500	0.0748	71.5	5.845	0.0574	99.6	6.876	0.0776

of the recursive mean filter. The AHM filter works quite well in preserving the image details (i.e., the stripes around birds' eyes). However, some pepper-like impulses are left in the reconstruction as a result of inaccurate/inadequate local statistics estimation. The ANNF and the AVDF preserve the edge integrity, but fine structures are blurred and the background is highly distorted by chromatic blotches. A better result is achieved by the PBTVM filter. It provides a smoother chromatic background, while the fine stripes and bird eyes are clearly preserved.

C. Computational Complexity Analysis

A general framework has been used to analyze the computational requirements of image filters [15], [19], [30], [48]. The framework evaluates the computational complexity of a specific filter by counting the total elementary operations required to processing an $n \times n$ window which contains n^2 vector samples. Several elementary operations are frequently taken into account in the analysis, which include additions (*ADDs*), multiplications (*MULTs*), divisions(*DIVs*), square roots (*SQRTs*), comparisons (*COMPs*), exponents (*EXPs*), and arc cosines (*ARCCOSs*).

Table V presents the elementary operations required for a number of vector filters, where the VMF, the SAA and the AVLUM use the Euclidian distance, the GVDF and the SCWVDF employ the vector directional distance, and the SWVDF utilizes the distance-directional distance. The weights of the SWVDF and the PBTVM filter are trained off-line, and the noise/parameter estimations of the SAA filter are bypassed for a maximum computation efficiency. The HBTM and the SAA filters occur a quite low computational cost which is comparable to the VMF. The computational complexity of the PBTVM filter is fairly reasonable despite its three-stage adaptation structure. Its operations associated with the reference estimation, structure partition, and weighted filtering are of $O(n^6)$ on the pixel basis, which is quite common to any selective vector filter, such as the AVLUM or the SWVDF. However, since the arc cosine operation usually cost more computation than other operations, the computational complexity of the PBTVM filter is usually much lower than that of the SCWVDF, and slightly higher than that of the AVLUM filter.

Table VI lists the execution times of a number of filters in processing the 256×256 RGB image *Lena* corrupted by 10%

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

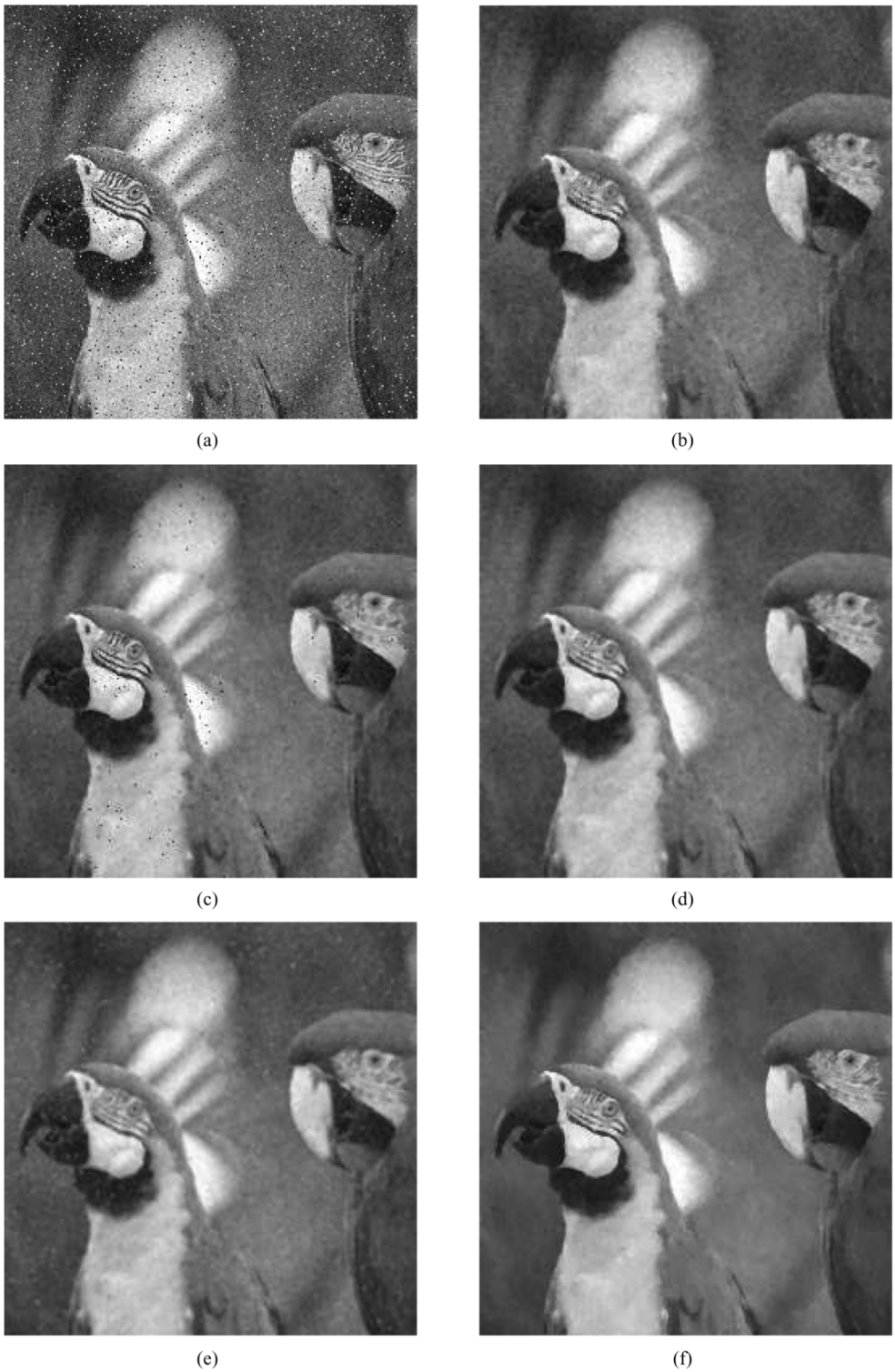

Fig. 11. Reconstruction of the proposed PBTVM filter compared with those by other techniques, where the test image *Parrots* is corrupted by the mixed Gaussian noise ($\sigma = 20$) and random impulse ($p_v = 5\%$). (a) Mixed noise corruption, (b) VMF output, (c) AHM output, (d) ANNF output, (e) AVDF output, and (f) PBTVM output.

random impulse noise using a 3×3-pixel window. Please note that the program was written in C++ and run on the same simulation platform (P4 3.0 GHz/2096 MB DDR/Window XP Pro) without any code optimization. Two work modes of the SAA filter are tested separately, i.e., the maximum computation efficiency model (SAA-a) where both the noise estimator and the parameter estimation were bypassed, and the normal work mode (SAA-b) where both the noise estimator and the parameter estimation are activated. It is interesting to note that execution time of the PBTVM filter is just slightly longer than that of the AVLUM, while comparable to that of the SAA-b filter where the on-line parameter estimation is activated. In contrast,

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

TABLE V
COMPUTATIONAL COMPLEXITY ANALYSIS OF THE PROPOSED PBTVM FILTER COMPARED WITH OTHER TECHNIQUES

Operations	*ADDs*	*MULTs/DIVs*	*SQRTs*	*COMPs*	*EXPs*	*ARCCOSs*
VMF [7]	$O(n^4)$	–	$O(n^4)$	$O(n^2)$	–	–
GVDF [8]	$O(n^4)$	$O(n^4)$	$O(n^4)$	$O(n^2)$	–	$O(n^4)$
SAA [19]	$O(n^4)$	$O(n^4)$	$O(n^4)$	$O(n^2)$	$O(n^4)$	–
HBTM [22]	$O(n^4)$	$O(n^2)$	$O(n^4)$	$O(n^2)$	–	–
SWVDF [15]	$O(n^4)$	$O(n^4)$	$O(n^4)$	$O(n^2)$	$O(n^2)$	$O(n^4)$
AVLUM [13]	$O(n^6)$	–	$O(n^4)$	$O(n^4)$	–	–
SCWVDF [14]	$O(n^6)$	$O(n^4)$	$O(n^4)$	$O(n^4)$	–	$O(n^4)$
PBTVM	$O(n^6)$	$O(n^4)$	$O(n^4)$	$O(n^4)$	–	–

TABLE VI
EXECUTION TIME OF THE PROPOSED PBTVM FILTER COMPARED WITH OTHER TECHNIQUES, WHERE THE 256×256 RGB IMAGE LENA CORRUPTED BY 10% RANDOM IMPULSE NOISE IS USED, AND ALL FILTERS RAN WITH A 3×3 FILTER WINDOW

Filter	VMF	GVDF	AVMF	SAA-a	SAA-b	HBTM	SWVDF	AVLUM	SCWVDF	PBTVM
Time (s)	2.2	3.4	2.5	2.5	11.2	2.7	4.4	8.4	16.4	10.1

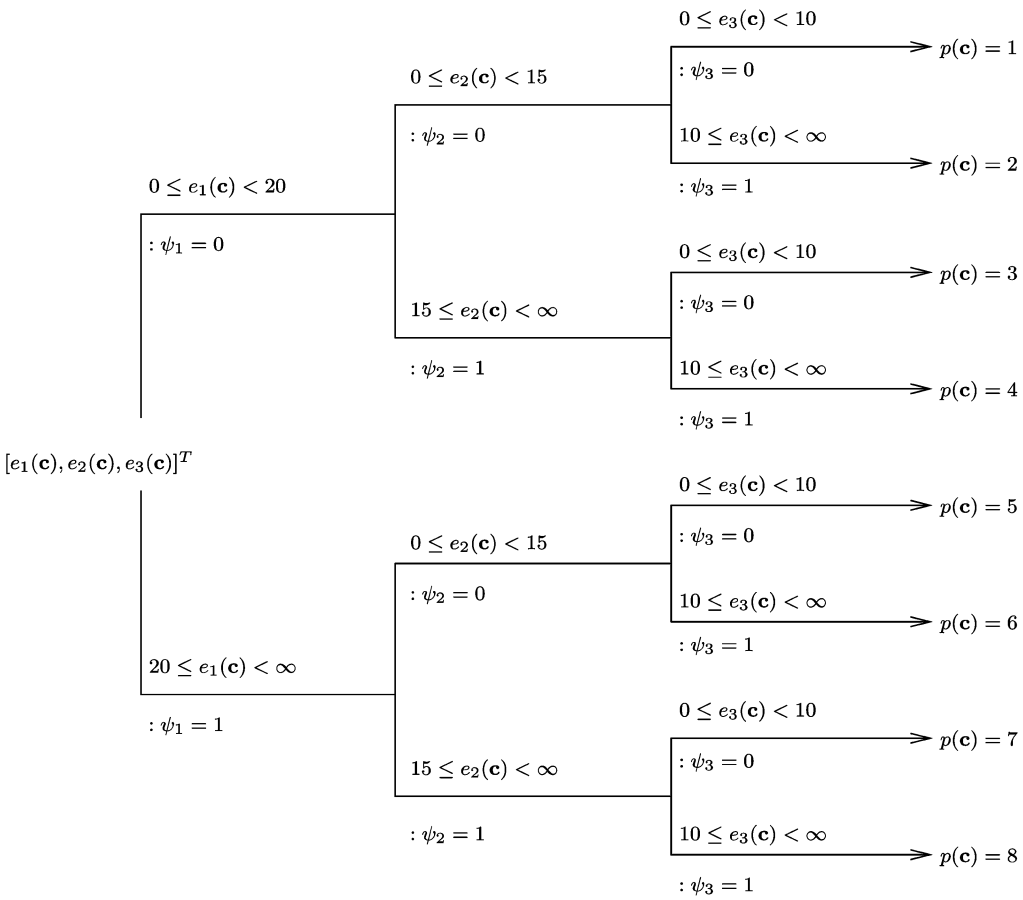

Fig. 12. Calculation of the partition index $p(c)$ for the input distance vector, $\hat{\mathbf{e}}(\mathbf{c}) = [e1(\mathbf{c}), e2(\mathbf{c}), e3(\mathbf{c})]^T$, where the partition threshold set is given by Table VII with $K = 3$ and $L = 2$.

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

TABLE VII
EXEMPLARY PARTITION THRESHOLD SET FOR $K = 3$ AND $L = 2$

$T_{k,l}$	$l = 0$	$l = 1$	$l = 2$
$k = 1$	0	20	∞
$k = 2$	0	15	∞
$k = 3$	0	10	∞

the SCWVDF, which follows a similar structure to that of the AVLUM but using the vector directional distance, requires almost 1.6 times the computational cost of the PBTVM filter to process the same corrupted image.

VI. CONCLUSION

A partition-based adaptive vector filter framework is proposed for suppressing distinct types of noise in digital color images. The filter uses a series of robust reference filters to estimate the local structure, and utilizes a structure classification to map the inputs into mutually exclusive partition cells. Adaptive weighting filtering is formulated to achieve the best restoration performance for the given local structure, where the filtering weights are optimized off-line for each partition cell through a constrained LMS algorithm. The proposed filtering technique demonstrates a superior performance over other state-of-the-art vector filters in suppressing the random impulse noise, the Gaussian noise, as well as the mixed Gaussian and impulse noise. Significant improvements have been observed in terms of standard objective measurements and perceived image quality.

APPENDIX

Example 1: Given the pixel samples of a 3×3 filter window, $\mathbf{W}(\mathbf{c})$, in a lexicographic scan order as follows:

$$\begin{Bmatrix} \mathbf{x}_1^T(\mathbf{c}) \\ \mathbf{x}_2^T(\mathbf{c}) \\ \mathbf{x}_3^T(\mathbf{c}) \\ \mathbf{x}_4^T(\mathbf{c}) \\ \mathbf{x}_5^T(\mathbf{c}) \\ \mathbf{x}_6^T(\mathbf{c}) \\ \mathbf{x}_7^T(\mathbf{c}) \\ \mathbf{x}_8^T(\mathbf{c}) \\ \mathbf{x}_9^T(\mathbf{c}) \end{Bmatrix} = \begin{Bmatrix} 76 & 76 & 51 \\ 72 & 71 & 49 \\ 71 & 255 & 255 \\ 75 & 255 & 0 \\ 74 & 255 & 52 \\ 70 & 68 & 51 \\ 73 & 75 & 52 \\ 71 & 69 & 50 \\ 70 & 65 & 51 \end{Bmatrix}$$

where three local pixels, $\mathbf{x}_3(\mathbf{c})$, $\mathbf{x}_4(\mathbf{c})$, and $\mathbf{x}_5(\mathbf{c})$, have been corrupted by the *Salt-and-pepper* impulse noise, and the central pixel (CP), $\mathbf{x}(\mathbf{c}) \triangleq \mathbf{x}_5(\mathbf{c}) = [74, 255, 52]^T$. The outputs of the CWVM filter with central weight $k = 1, 2, 3, 4$, respectively, are produced according to (1)–(2) and listed as follows:

$$\begin{Bmatrix} \mathbf{y}_1^T(\mathbf{c}) \\ \mathbf{y}_2^T(\mathbf{c}) \\ \mathbf{y}_3^T(\mathbf{c}) \\ \mathbf{y}_4^T(\mathbf{c}) \end{Bmatrix} = \begin{Bmatrix} 73 & 75 & 52 \\ 73 & 75 & 52 \\ 74 & 255 & 52 \\ 74 & 255 & 52 \end{Bmatrix}.$$

When a low center weight value is applied, the outputs of the CWVM filter, e.g., $\mathbf{y}_1(\mathbf{c})$ and $\mathbf{y}_2(\mathbf{c})$, are able to assemble the noise-free CP value. However, when the center weight becomes higher, the outputs of the CWVM filter, e.g., $\mathbf{y}_3(\mathbf{c})$ and $\mathbf{y}_4(\mathbf{c})$, suffer from serious impulse corruption.

Example 2: Consider again the filter window $\mathbf{W}(\mathbf{c})$ given in Example 1, where $N = 9$ and $\mathbf{x}(\mathbf{c}) = \mathbf{x}_5(\mathbf{c}) = [74, 255, 52]^T$. Applying the CP-distance ranking method of (3)–(5) to the observed pixels will result in a re-ordered pixel sequence $\hat{\mathbf{W}}(\mathbf{c})$ as follows:

$$\begin{Bmatrix} \mathbf{x}_{(1)}^T(\mathbf{c}) \\ \mathbf{x}_{(2)}^T(\mathbf{c}) \\ \mathbf{x}_{(3)}^T(\mathbf{c}) \\ \mathbf{x}_{(4)}^T(\mathbf{c}) \\ \mathbf{x}_{(5)}^T(\mathbf{c}) \\ \mathbf{x}_{(6)}^T(\mathbf{c}) \\ \mathbf{x}_{(7)}^T(\mathbf{c}) \\ \mathbf{x}_{(8)}^T(\mathbf{c}) \\ \mathbf{x}_{(9)}^T(\mathbf{c}) \end{Bmatrix} = \begin{Bmatrix} 74 & 255 & 52 \\ 75 & 255 & 0 \\ 76 & 76 & 51 \\ 73 & 75 & 52 \\ 72 & 71 & 49 \\ 71 & 69 & 50 \\ 70 & 68 & 51 \\ 70 & 65 & 51 \\ 71 & 255 & 255 \end{Bmatrix}.$$

Note that three corrupted pixels, $\mathbf{x}_3(\mathbf{c})$, $\mathbf{x}_4(\mathbf{c})$, and $\mathbf{x}_5(\mathbf{c})$ of the $\mathbf{W}(\mathbf{c})$, are now ranked in the last $(\mathbf{x}_{(9)}(\mathbf{c}))$, the second $(\mathbf{x}_{(2)}(\mathbf{c}))$, and the first $(\mathbf{x}_{(1)}(\mathbf{c}))$ positions in the $\hat{\mathbf{W}}(\mathbf{c})$, respectively. So, when the central weight $k > 1$, one of these corrupted pixels, $\mathbf{x}_3(\mathbf{c})$ (i.e., $\mathbf{x}_{(9)}(\mathbf{c})$), will be trimmed out of the modified CW-AVD calculation (7). As a result, the outputs of the CWTVM filter with central weights, $k = 1, 2, 3, 4$, will be

$$\begin{Bmatrix} \hat{\mathbf{y}}_1^T(\mathbf{c}) \\ \hat{\mathbf{y}}_2^T(\mathbf{c}) \\ \hat{\mathbf{y}}_3^T(\mathbf{c}) \\ \hat{\mathbf{y}}_4^T(\mathbf{c}) \end{Bmatrix} = \begin{Bmatrix} 73 & 75 & 52 \\ 73 & 75 & 52 \\ 73 & 75 & 52 \\ 74 & 255 & 52 \end{Bmatrix}.$$

With the contribution of the CP-distance trimming technique, the CWTVM filter with $k = 3$ is able to produce a noise free output, whilst the output of the CWVM filter with the same center weight still suffers from the impulse corruption.

Example 3: Given $K = 3$ and $L = 2$, the total number of partition cells $M = L^K = 8$. Fig. 12 demonstrates the calculation of the partition index $p(\mathbf{c})$ according to (13) and (14), where the input distance vector is $\hat{\mathbf{e}}(\mathbf{c}) = [e_1(\mathbf{c}), e_2(\mathbf{c}), e_3(\mathbf{c})]^T$, and the threshold set $\{T_{k,l} \mid 1 \le k \le K, 0 \le l \le L\}$ is given by Table VII.

ACKNOWLEDGMENT

The authors would like to thank Dr. R. Lukac of the Bell Canada Multimedia Laboratory, University of Toronto, for kindly providing all the corrupted test images, and the anonymous reviewers for their valuable comments which helped to improve the quality of the paper.

REFERENCES

[1] V. Barnett, "The ordering of multivariate data," *J. Statist. Soc. Amer. A.*, vol. 139, no. 3, pp. 318–354, 1976.
[2] R. C. Hardie and G. R. Arce, "Ranking in R^p and its use in multivariate image estimation," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 1, no. 2, pp. 197–209, Jun. 1991.

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

[3] I. Pitas and P. Tsakalides, "Multivariate ordering in color image filtering," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 1, no. 3, pp. 247–259, Sep. 1991.
[4] K. Tang, J. Astola, and Y. Neuvo, "Nonlinear multivariate image filtering techniques," *IEEE Trans. Image Process.*, vol. 4, no. 6, pp. 788–798, Jun. 1995.
[5] K. N. Plataniotis and A. N. Venetsanopoulos, *Colour Image Processing Handbook*. Cambridge, U.K.: Chapman & Hall, 1997, ch. Vector processing, pp. 188–209.
[6] K. N. Plataniotis and A. N. Venetsanopoulos, *Color Image Processing and Applications*. Berlin, Germany: Springer, 2000.
[7] J. Astola, P. Haavisto, and Y. Neuov, "Vector median filter," *Proc. IEEE*, vol. 78, no. 4, pp. 678–689, Apr. 1990.
[8] P. E. Trahanias and A. N. Venetsanopoulos, "Vector directional filters: A new class of multichannel image processing filter," *IEEE Trans. Image Process.*, vol. 2, no. 4, pp. 528–534, Oct. 1993.
[9] D. G. Karakos and P. E. Trahania, "Combining vector median and vector directional filters: The directional-distance filter," in *Proc. IEEE Int. Conf. Image Processing*, Washington, DC, Oct. 1995, vol. 1, pp. 171–174.
[10] T. Viero, K. Oistamo, and Y. Neuvo, "Three-dimensional median-related filters for color image sequence filtering," *IEEE Trans. Circuits Syst.*, vol. 4, no. 2, pp. 129–142, Apr. 1994.
[11] B. Smolka, M. Szczepanski, K. N. Plataniotis, and A. N. Venetsanopoulos, "On the modified weighted vector median filter," in *Proc. IEEE Int. Conf. Digital Signal Process*, Santorini, Greece, Jul. 2002, vol. 1, pp. 939–942.
[12] R. Lukac, "Adaptive vector median filtering," *Pattern Recognit. Lett.*, vol. 24, no. 12, pp. 1889–1899, Aug. 2003.
[13] R. Lukac and S. Marchevsky, "Adaptive vector LUM smoother," in *Proc. IEEE Int. Conf. Image Processing*, Oct. 2001, vol. 1, pp. 878–881.
[14] R. Lukac, "Adaptive color image filtering based on center-weighted vector directional filters," *Multidimensional Syst. Signal Process.*, vol. 15, no. 2, pp. 169–196, Apr. 2004.
[15] R. Lukac, B. Smolka, K. N. Plataniotis, and A. N. Venetsanopoulos, "Selection weighted vector directional filter," *Comput. Vis. Image Understand.*, vol. 94, no. 1–3, pp. 140–167, Apr. 2004.
[16] R. Lukac, K. N. Plataniotis, B. Smolka, and A. N. Venetsanopoulos, "Generalized selection weighted vector filter," *EURASIP J. Appl. Signal Process.*, vol. 2004, no. 12, pp. 1870–1885, Oct. 2004.
[17] K. N. Plataniotis, S. Vinayagamoorthy, D. Androutsos, and A. N. Venetsanopoulos, "An adaptive nearest neighbor multichannel filter," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 6, pp. 699–703, Dec. 1996.
[18] K. N. Plataniotis, D. Androutsos, and A. N. Venetsanopoulos, "Color image processing using adaptive vector directional filters," *IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process.*, vol. 45, no. 10, pp. 1414–1419, Oct. 1998.
[19] B. Smolka, A. Chydzinski, K. Wojciechowski, K. N. Plataniotis, and A. N. Venetsanopoulos, "Self-adaptive algorithm for impulsive noise reduction in color images," *Pattern Recognit.*, vol. 35, no. 8, pp. 1771–1784, Aug. 2002.
[20] B. Smolka, R. Lukac, A. Chydzinski, K. N. Plataniotis, and K. Wojciechowski, "Fast adaptive similarity based impulse noise reduction filter," *Real-Time Imag.*, vol. 9, no. 4, pp. 261–276, Aug. 2003.
[21] E. S. Hore, B. Qiu, and H. R. Wu, "Prediction based image restoration using a multiple window configuration," *Opt. Eng.*, vol. 41, no. 8, pp. 1–11, Aug. 2002.
[22] Z. Ma and H. R. Wu, "A histogram based adaptive vector filter for color image restoration," in *Proc. IEEE 4th Int. Pacific-Rim Conf. Inf., Commun., Signal Process. Multimedia*, Singapore, Dec. 15–18, 2003, pp. 1A3.4.1–5.
[23] R. S. Lin and Y. C. Hsueh, "Multichannel filtering by gradient information," *Signal Process.*, vol. 80, no. 2, pp. 279–293, Feb. 2000.
[24] C. Kenney, Y. Deng, B. S. Manjunath, and G. Hewer, "Peer group image enhancement," *IEEE Trans. Image Process.*, vol. 6, no. 2, pp. 326–334, Feb. 2001.
[25] M. Szczepanski, B. Smolka, K. N. Plataniotis, and A. N. Venetsanopoulos, "On the geodesic paths approach to color image filter," *Signal Process.*, vol. 83, no. 6, pp. 1309–1342, Jun. 2003.
[26] K. N. Plataniotis, D. Androutsos, and A. N. Venetsanopoulos, "Adaptive fuzzy systems for multichannel signal processing," *Proc. IEEE*, vol. 87, no. 9, pp. 1601–1622, Sep. 1999.
[27] V. Chatzis and I. Pitas, "Fuzzy scalar and vector median filters based on fuzzy distances," *IEEE Trans. Image Process.*, vol. 8, no. 5, pp. 731–734, May 1999.
[28] H. H. Tsai and P. T. Yu, "Genetic-based fuzzy hybrid multichannel filters for color image restoration," *Fuzzy Sets Syst.*, vol. 114, no. 9, pp. 203–224, Sep. 2000.
[29] L. Khriji and M. Gabbouj, "Adaptive fuzzy order statistics-rational hybrid filters for color image processing," *Fuzzy Sets Syst.*, vol. 128, no. 1, pp. 35–46, Mar. 2002.
[30] B. Smolka, K. N. Plataniotis, and A. N. Venetsanopoulos, *Nonlinear Signal and Image Provessing: Theory, Methods, and Applications*. Boca Raton, FL: CRC, 2003, ch. Nonlinear techniques for color image processing, pp. 445–505.
[31] E. S. Hore, B. Qiu, and H. R. Wu, "Improved color image vector filtering using fuzzy noise detection," *Opt. Eng.*, vol. 42, no. 6, pp. 1656–1664, Jun. 2003.
[32] L. Lucchese and S. K. Mitra, "A new class of chromatic filters for color image processing: Theory and applications," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 534–548, Apr. 2004.
[33] R. Lukac, B. Smolka, K. Martin, K. N. Plataniotis, and A. N. Venetsanopoulos, "Vector filtering for color imaging," *IEEE Signal Process. Mag.*, vol. 22, no. 1, pp. 74–86, Jan. 2005.
[34] J. Astola and P. Kuosmanen, *Fundamentals of Nolinear Digital Filtering*. Boca Raton, FL: CRC, 1997.
[35] L. Alparone, M. Barni, F. Bartolini, and R. Caldelli, "Regularization of optic flow estimates by means of weighted vector median filtering," *IEEE Trans. Image Process.*, vol. 8, no. 10, pp. 1462–1467, Oct. 1999.
[36] L. Lucat, P. Siohan, and D. Barba, "Adaptive and global optimization methods for weighted vector median filters," *Signal Process.: Image Commun.*, vol. 17, no. 7, pp. 509–524, Aug. 2002.
[37] R. Lukac, K. N. Plataniotis, B. Smolka, and A. N. Venetsanopoulos, "Weighted vector median optimization," in *Proc. 4th EURASIP Conf. Video/Image Processing and Multimedia Communications*, Jul. 2003, vol. 1, pp. 227–232.
[38] R. M. Nosovsky, "Choice, similarity and the context theory of classification," *J. Exp. Psych. Learn., Memory Cogn.*, vol. 10, pp. 104–114, Jan. 1984.
[39] C. Kotropoulos and I. Pitas, "Adaptive LMS L-filter for noise suppression in image," *IEEE Trans. Image Process.*, vol. 5, no. 12, pp. 1596–1609, Dec. 1996.
[40] E. Abreu, M. Lightstone, S. K. Mitra, and K. Arakawa, "A new efficient approach for the removal of impulsive noise from highly corrupted images," *IEEE Trans. Image Process.*, vol. 5, no. 6, pp. 1012–1025, Jun. 1996.
[41] T. Chen and H. R. Wu, "Application of partition-based median type filters for suppressing noise in images," *IEEE Trans. Image Process.*, vol. 10, no. 6, pp. 829–836, Jun. 2001.
[42] A. K. Jain, *Fundamentals of Digital Image Processing*. Englewood Cliffs, N.J: Prentice-Hall, 1989.
[43] J. C. Bezdek, *Pattern Recoginition With Fuzzy Objective Function Algorithm*. New York: Plenum, 1981.
[44] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*. Norwell, MA: Kluwer, 1992.
[45] K. E. Barner, A. M. Sarhan, and R. C. Hardie, "Partition-based weighted sum filters for image restoration," *IEEE Trans. Image Process.*, vol. 8, no. 5, pp. 740–745, May 1999.
[46] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery, *Numberical recipes in C*. Cambridge, U.K.: Cambridge Univ. Press, 1999.
[47] Z. Ma, H. R. Wu, and B. Qiu, "An structure adaptive hybrid vector filter for the restoration of digital color images," *IEEE Trans. Image Process.*, vol. 14, no. 12, pp. 1900–2001, Dec. 2005.
[48] M. Barni and V. Cappellini, "On the computational complexity of multivariate median filters," *Signal Process.*, vol. 71, no. 1, pp. 45–54, Nov. 1998.
[49] R. E. Blahut, *Fast Algorithms for Digital Signal Processing*. Reading, MA: Addison-Wesley, 1985.

Zhonghua Ma (M'02) was born in Zhejiang, China, in 1973. He received the B.Eng. degree from Shanghai Jiaotong University, China, the M.Eng. degree from the Dalian University of Technology, China, in 1995 and 1999, respectively, and the Ph.D. degree in communications and information systems from Shanghai Jiaotong University, China, in 2002.

From 2002 to 2004, he was a Research Fellow conducting an Australia Research Council (ARC) funded project with the School of Computer Science and Software Engineering, Monash University, Melbourne, Australia. Since 2004, he has been a Research Fellow with the

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.

Biomedical and Multimedia Information (BMIT) Group in School of Information Technologies, University of Sydney, Sydney, Australia. His research interests include multivariance signal processing, color image restoration, video compression, and robust streaming.

Hong Ren Wu was born in Beijing, China, in 1956. He received the B.Eng. and M.Eng. degrees from the University of Science and Technology (formerly Beijing University of Iron and Steel Technology), Beijing, in 1982 and 1985, respectively, and the Ph.D. degree in electrical and computer engineering from the University of Wollongong, Wollongong, Australia, in 1990.

From 1982 to 1985, he was an Assistant Lecturer with the Department of Industrial Automation, University of Science and Technology. He joined the Department of Robotics and Digital Technology, Chisholm Institute of Technology, and then the Faculty of Information Technology, Monash University, Melbourne, Australia, in 1990, where he served as a Lecturer (1990 to 1992), Senior Lecturer (1992 to 1996), and Associate Professor of digital systems (1997 to 2005). Since 2005, he has been with the School of Electrical and Computer Engineering, Royal Melbourne Institute of Technology, Melbourne, Australia, where he is on the academic staff as Professor of visual communications engineering and the Director of Software and Network Engineering. Since 1987, he has participated in and headed numerous research projects and industrial contracts in the fields of signal processing, video and image processing and enhancement, perceptual coding of natural and medical images, as well as digital video coding, compression, and transmission. He is the Co-Editor of the book *Digital Video Image Quality and Perceptual Coding* (CRC, 2005).

Dagan Feng (S'88–M'88–SM'94–F'03) was born in Nanking, China. He received the M.E. degree in electrical engineering and computer science from Shanghai Jiaotong University, Shanghai, China, in 1982, and the M.Sc. degree in biocybernetics and the Ph.D. degree in computer science from the University of California, Los Angeles, in 1985 and 1988, respectively.

After being an Assistant Professor in the U.S., he joined the University of Sydney, Sydney, Australia, as Lecturer, Senior Lecturer, Reader, and then Professor. He is the former Head of Department of Computer Science/School of Information Technologies. He is currently Associate Dean of the Faculty of Science and Director of the Biomedical and Multmedia Information (BMIT) Research Group, University of Sydney, and Chair Professor and Deputy Director, Center for Multimedia Signal Processing, Department of Electronic and Information Engineering, Hong Kong Polytechnic University. He has published over 300 scholarly research papers, pioneered several new research directions, and made a number of landmark contributions in his field. More important, however, is that many of his research results have been translated into solutions to real-life problems worldwide and have made tremendous improvements to the quality of life.

Dr. Feng is a Fellow of ACS, ATSE, HKIE, and IEE. He received the Crump Prize for Excellence in Medical Engineering. He is currently the Special Area Editor of the IEEE Transactions on Information Technology in Biomedicine and Chairman of IFAC-TC-BIOMED.

Authorized licensed use limited to: RMIT University. Downloaded on November 23, 2008 at 21:32 from IEEE Xplore. Restrictions apply.