Impulse Noise Removal from Color Images with Hopfield Neural Network and Improved Vector Median Filter

G. Phani Deepti, Maruti V Borker and Jayanthi Sivaswamy
Center for Visual Information Technology,
International Institute of Information Technology, Hyderabad -500032, India.
{phanideepti@students.,maruti@students.,jsivaswamy@}iiit.net

Abstract

In this paper, a novel and effective method for impulse noise removal in corrupted color images is discussed. The new method consists of two phases. The first phase is a noise detection phase where a modified Hopfield neural network is used to detect impulse noise pixels. The second is a noise filtering phase where the disadvantage of taking Vector Median in a single color space is addressed and a new algorithm based on performing Vector Median first in RGB space and then in HSI space is presented. The results of simulations performed on a set of standard test images on a wide range of noise corruption show that the proposed method is capable of detecting all the impulse noise pixels with almost zero false positive rates and removes noise while retaining finer image details. It outperforms the standard procedures and is yet simple and suitable for real time applications.

1. Introduction

Many types of distortions limit the quality of digital images during image acquisition, formation, storage and transmission. Often, images are corrupted by impulse noise. The intensity of impulse noise has the tendency of being either relatively high or low thereby causing loss of image details. It is important to eliminate noise in images before using them for other image processing techniques like edge detection, segmentation, registration etc. Several filtering methods have been proposed in the past to address impulse noise removal. One of the more famous filters for gray scale images is the standard median filter which rank orders the pixel intensities within a filtering window and replaces the center pixel with the median value. Extending the idea of a scalar median filter to color images is not straightforward due to the lack of a natural concept of ranking among the vectors. Color distortion may occur when the scalar median filter is applied separately to every single component of the color vectors. A method called Vector Median Filter (VMF) which considers all the three color components and rank orders the vectors has been proposed in [2]. Various modifications of the standard VMF have been introduced like Directional Median Filter [28] and Central Weighted Vector Median Filter [30]. The idea of applying a filter in a color space different from RGB was introduced in [29] and [3] which uses HSI and L*a*b* spaces respectively.

The biggest drawback of the conventional vector median approaches is that they apply median operation to each pixel, irrespective of it being corrupted or not. An intuitive solution to overcome this disadvantage is to first detect the corrupt pixels and then to apply filtering on those pixels alone. Several algorithms have been proposed which follow this two-step method. Of them, a large class [14][25][31][33] is based on rank ordering the color vectors and another exhaustive class [20][9][13] is based on fuzzy techniques.

One of the main problems with impulse noise detection is that it is difficult to differentiate between an edge and an impulse noise. In the intensity space, both these stand as peaks in their neighborhood. In [14], the difference between the center pixel with the minimum and maximum gray value in the filtering window is taken and if greater than a certain threshold, the center pixel is considered as noise. The disadvantage of this method is that the false positive rate is very high and most of the edges also get detected as noise. The same disadvantage applies to detection methods followed in [25], [31] and [33]. Moreover, [25] works only on random valued impulse noise and [33] is computationally expensive. In [34], a neural network was trained for impulse detection. But it works only on fixed impulse noise and the learning method of the network is supervised. In [20], which is a fuzzy technique, the membership function is very crucial for the performance of the noise detection step. This function depends on many parameters which need to be tuned.

Our goal is to develop a technique which is fast and ac-

978-0-7695-3476-3/08 $25.00 © 2008 IEEE
DOI 10.1109/ICVGIP.2008.75

curate and needs tuning of a minimal number of parameters. In this paper, we propose a novel two-step solution for impulse noise removal. The steps are:

1. Noise detection based on Hopfield Neural Network.

2. Noise Filtering based on VMF in two color spaces.

The noise detection is unsupervised since HNN is a self-learning network and it needs tuning of minimal parameters. Once tuned, the parameters work well on a varied set of images. The noise filtering method is developed by addressing the disadvantage(s) of applying VMF only in a single color-domain as was done till now. The proposed algorithm tested on different images across various levels of noise has shown superior performance both qualitatively and quantitatively in both PSNR and Mean Absolute Error (MAE) measurements.

Section 2 describes our algorithm for impulse noise detection. Section 3 describes the proposed noise filtering algorithm. Section 4 presents experiment details and results followed by discussion and concluding remarks in Sections 5 and 6.

2. Impulse Noise Detection

A Hopfield Neural Network (HNN) is a single-layered self feedback structure or a recurrent ANN where every unit is connected to every other. It takes binary values as inputs i.e., either -1 and +1 or 0 and 1 and gives binary values as outputs based on an activation function. Every connection in the network can be associated with a weight and is represented as a two-dimensional matrix called a weight matrix W. The Hopfield network is trained by supplying input feature vectors or pattern vectors corresponding to different classes. The network is then used to classify the test patterns into classes whose patterns it was trained with. Its architecture is shown in Figure 1.

The HNN was modified in [5] and was used for edge detection in gray scale images. We have adapted the modified HNN for impulse noise detection.

Let x be an input vector, W,the weight matrix, a,the activation function ($a = Wx_t$) and let $sgn[y]$ represent the signum function.

Then the transmission rule is :-

$$x_{t+1} = sgn[Wx_t], t = 1...T \quad (1)$$

$$sgn[a_j] = \begin{cases} a_j = 1, & If\ a_j > 0 \\ a_j = -1, & If\ a_j < 0 \\ Else & a_j = 0 \end{cases} \quad (2)$$

The learning rule is :

$$W_{k+1} = W_k + \eta[x_0 x_0^T - (Wx_t)(Wx_t)^T] \quad (3)$$

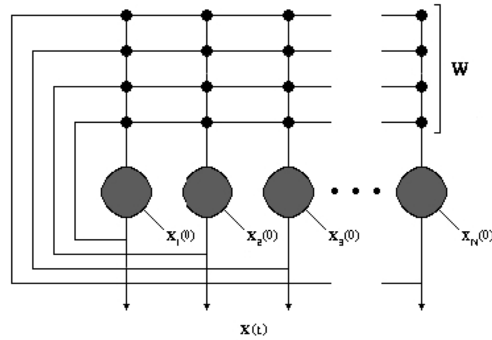

Figure 1. Architecture of N unit HNN (Image Courtesy of [5])

where η ($\eta > 0$) represents the learning coefficient and t ($t > 0$) represents the number of iterations the state vector has to perform before a weight matrix update.

Feature Vectors:

The feature vectors are derived from the image patches around randomly chosen pixels. For every chosen pixel, its 8-neighborhood (N8) is considered and the resulting 3×3 mask is row-ordered to obtain the 9-dimensional feature vector. Though the standard HNN takes only binary values, we modified it to take values which fall in the range of $[-1, 1]$ by rescaling the feature vector values. After one iteration, the signum function makes all the values binary. With these feature vectors, a trained network can detect those noise pixels which are in complete contrast with their N8 neighbors. In order to make our HNN distinguish even those noise pixels which vary to a lesser extent from their N8 neighbors, a further rescaling of the input vectors is done iteratively to focus the attention of the network on a small range of values at a time. The rescaling procedure followed in [5] is used.

Training the Network:

Since the feature vectors are small random patches of an image, a single image will serve the purpose of a training image. Since we want our HNN to detect impulse noise differentiating it from the edges and smooth regions, any image with good variety of edges and sufficient impulse noise is sufficient for training. The learning algorithm is the same as that used in [5]. Thus, the HNN is trained with the input vectors which correspond to three classes impulse noise, edges and smooth regions.

Identification of the noise pattern:

Before testing the network, it is important to understand the behavior of the network in the presence and absence of impulse noise. Since there are 9 nodes in our HNN, we get 9 outputs. The network will respond not only to the impulse noise in the image but also to the smooth regions and to the edges.

If the center pixel being considered is in a smooth neighborhood i.e., if all the pixels in a filtering window are not noisy, then all the 9 nodes behave in the same way and output the same value i.e.,either +1 or -1.

If the center pixel is corrupted, it's output behaves distinctly from that of it's 8-neighbors i.e., if all the 8 neighbors give +1 as an output, the output of the center pixel will be -1 and vice versa. The same has been illustrated below:

$+1$	$+1$	$+1$	$+1$	-1	$+1$	$+1$	$+1$	$+1$

(OR)

-1	-1	-1	-1	$+1$	-1	-1	-1	-1

It is desirable to obtain a single output, for example a +1 for a noise pixel and a 0 for a non-noise pixel. To achieve this, all the 9 nodes are connected to a single node called the output node. The function of this node is as follows:

$$output = \begin{cases} 1 & \text{if } \sum_{(i=1,i\neq 5)}^{9} x_i = 8 \;\; and \;\; x_5 = -1, \text{ or} \\ & \sum_{(i=1,i\neq 5)}^{9} x_i = -8 \;\; and \;\; x_5 = 1, \\ 0 & \text{otherwise.} \end{cases} \tag{4}$$

Thus, given a test image to a trained HNN, it performs a binary classification of every pixel into either of the two classes - noise or non-noise.

Extending the above method to color images:

The above discussed method works for gray scale images. In order to extend it to color images, we apply this method to each channel independently. If a pixel is classified as noisy in any of the three R, G or B channels, then it is classified as a corrupted pixel in the color image.

3. Noise Filtering

In general, Vector Median is calculated by taking the sum of the distances from every pixel to every other pixel in its neighborhood in the filtering window and the pixel which gives the minimum of these distances is taken to be the Vector Median of that filtering window.

The main disadvantage of Vector Median filtering is that, in practice, no distinct minimum exists. This gives rise to more than one pixel being fit for getting selected as the Vector Median. Let N be the total number of pixels and S represent the subset of pixels which are fit for being a Vector Median in the considered filtering window. If the region being considered is a smooth region, choosing any one of the pixels from S will not distort the visual perception of the image because all of them will have more or less the same value. But in the case of a non smooth region, selecting any pixel from S without further verification is likely to give unacceptable results because the pixels may not have similar values. Thus, there should be a criterion to solve this ambiguity. The standard VMF [2] does not impose any rule to pick the best fit pixel in this case.

Hence we propose an **Improved VMF** which is based on the fact that color distortions in the image have a direct influence on the color perception of the viewer. Accordingly, there should also be additional perception-referred criteria for the selection of the most suitable candidate in the filtering window.

Hue is considered as the single measurement of color experience and it represents the most sensitive direction in the 3-D space in color image analysis. Thus, in HSI space, the following rules have been used in [29] which perform VMF in HSI space only:

1. The change in the hue values should be as minimal as possible and

2. The change in the saturation should be as minimal as possible.

In our proposed algorithm, we resolve the ambiguity by first taking VMF in RGB space and if there are more than one suitable candidates, then move to HSI space. The details of the algorithm are presented below:

Algorithm for Improved VMF:

Step 1: Calculate Vector Median in RGB space in the filtering window.

Step 2: Collect all the pixels which are fit for being the median of the filtering window.

Step 3: If there is more than one pixel fit for being the median, then from these select that pixel which falls closest to the mean of the Hue in HSI space.

Step 4: If more than one pixel qualifies in Step 3, then select that pixel which falls closest to the mean of Saturation in HSI space.

This strategy was found to resolve the ambiguity at the end of Step 4.

4. Experiments and Results

The HNN was trained with a learning coefficient $\eta = 0.01$ and with 2000 randomly chosen feature vectors. In the training image, a random pixel is chosen and a 3×3 sub-window around it is used as the feature vector for training the network. Since the selection of the pixel is random, it is to be ensured that the network is trained with a sufficient number of noise pixels. Any gray scale image corrupted with greater than 5% of noise will ensure the same.

The image shown in Figure 2 has been used to train the HNN. It is a gray scale image and is corrupted with 10% of salt and pepper noise. This image has a good variety of edges as well as impulse noise so that the HNN learns the difference between both of them.

Figure 2. Grey scale image corrupted with salt and pepper noise which is used for training the HNN.

Noise Models:

We performed experiments by varying the amount of noise and also by introducing different types of noise. In general, a corrupted image can be modeled as:

$$X(i,j) = \begin{cases} O(i,j) & with\ a\ probability\ of\ 1-p \\ N(i,j) & with\ a\ probability\ of\ p \end{cases} \quad (5)$$

where (i,j) is the pixel location, p is the percentage of amount of noise and $N(i,j)$ is the value of the impulse noise and $O(i,j)$ is the original pixel value. There are mainly three types of noise Models used in this paper depending on the values which $N(i,j)$ can take. They are:

Fixed Impulse noise model: In this model, an L-bit image $N(i,j)$ can have only two values namely $\{0, 2^L - 1\}$. For an 8-bit image, the two values this noise model can take are 0 and 255

Random Impulse noise model: In this model, an L-bit image, $N(i,j)$ can have any value in the range $[0, 2^L - 1]$. For an 8-bit image, the value of $N(i,j)$ is chosen uniformly from the range $[0, 255]$.

Additive Impulse noise model: In this model, the value of $N(i,j)$ is as follows:

$$N(i,j) = O(i,j) + X \quad (6)$$

where X is a random value chosen uniformly from the range $[0, 255]$.

In all the above 3 models, a random pixel is selected and a random channel of that pixel is corrupted accordingly.

A sample result of different noise detection algorithms applied on a noisy baboon image with 15% fixed impulse noise is presented in Figure 3. The result shown in Figure 3(d) is generated by using a 7×7 mask whereas the rest use 3×3 mask for noise detection. In the shown results, the detected noise pixels are presented in red color.

The qualitative assessment of the recovered image is done by forming a difference (between the original and the recovered) image. For quantitative assessment of the restoration quality, the commonly used Peak Signal to Noise Ratio (PSNR) was used.

$$PSNR = 20 \log_{10} \left(\frac{255}{\sqrt{MSE}} \right) \quad (7)$$

$$MSE = \frac{\sum_{i=1}^{N} \sum_{k=1}^{m} (x_{ik} - o_{ik})^2}{Nm} \quad (8)$$

where m is the total number of color components, N the total number of image pixels and x_{ik} and o_{ik} the k^{th} component of the noisy image pixel channel and its original value at pixel position "i" respectively ([22])

For the evaluation of the detail-preservation capabilities of the proposed filter, the Mean Absolute Error has been used.

$$MAE = \frac{\sum_{i=1}^{N} \sum_{k=1}^{m} |x_{ik} - o_{ik}|}{Nm} \quad (9)$$

Superior performance of the technique is indicated by high PSNR values and low MAE values.

To justify the superior performance of the proposed noise filtering algorithm over the standard VMF, we apply both the methods on the corrupted baboon image with 12% random impulse noise without a prior noise detection. A sample result of this experiment is presented in Figure 4.

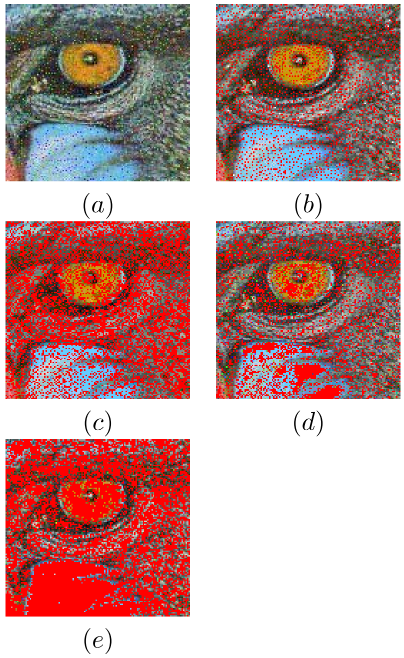

Figure 3. Results of different noise detection algorithms (a) Zoomed part of the original baboon image (b) Proposed detection algorithm (c) Adaptive Median filter [10] (d) ISAINF [21] (e) Method in [1]

Certain fuzzy techniques outperform even the best algorithms based on rank-ordering techniques. Thus, we compared our technique with some of the fuzzy techniques. The results are presented in Figure 6.

In order to compare the performance of our technique with different techniques, we performed experiments for different noise levels and noise models on different images. Figures 5 and 7 display the comparative results on the standard test images *Lena* and *Pepper* corrupted with 15% of additive and 5% of random impulse noise respectively.

For qualitative comparison, the difference between the original and the recovered images obtained from different algorithms has been depicted for Peppers. Each pixel value in the difference image is the vectorial difference between the original and the recovered image in RGB color space. Since most of the difference values lie in the range $[0 - 64]$, the pixel values in the difference images presented represent only those values which lie in the range $[0 - 64]$ but scaled by a factor of 4 to observe the difference more keenly.

5. Discussion

The results presented in Figure 3 indicate the efficiency of the proposed HNN based detection algorithm over widely used existing standard detection algorithms . The results indicate that the proposed detection algorithm has the least false positive rate even on a high detailed *baboon* image.

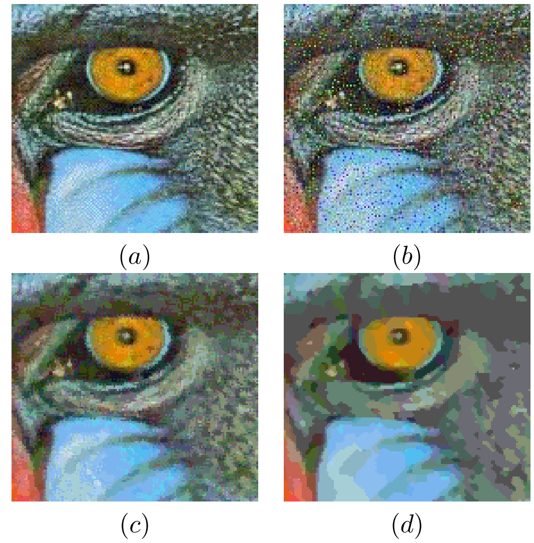

Figure 4. Comparision of noise filtering performance without noise detection on corrupted baboon image (a) Original Image (b)Corrupted Image (c) Proposed method (d) VMF in [2]

Figure 4 presents the results which clearly depict the efficiency of our noise filtering algorithm over the standard VMF which introduces a lot of smoothing.

Figure 5 compares different algorithms on the standard lena image for 15% additive impulse noise. It can be observed that VMF introduces a lot of smoothing whereas our technique works as good as the other techniques and preserves details and edges and does not introduce any artifacts.

We can notice from Figure 6 that the result of our technique is comparable to the best fuzzy technique FTSCF. Though FTSCF method can be considered as better than our technique,it has its own set of disadvantages. Firstly, the detection algorithm depends on a membership function which depends on fine tuning of a number of parameters. The ranges in which the values of these parameters can vary are wide. The noise removal method is not a single step but is iterative. In contrast, our detection algorithm depends on a single parameter which takes up a single value and gives good results for that value for a wide range of noise and for different images. Neither the detection nor the removal are iterative and thus are fast and can be used for real time applications.

The difference images presented in Figure 7 clearly show the efficiency of our algorithm over other algorithms.

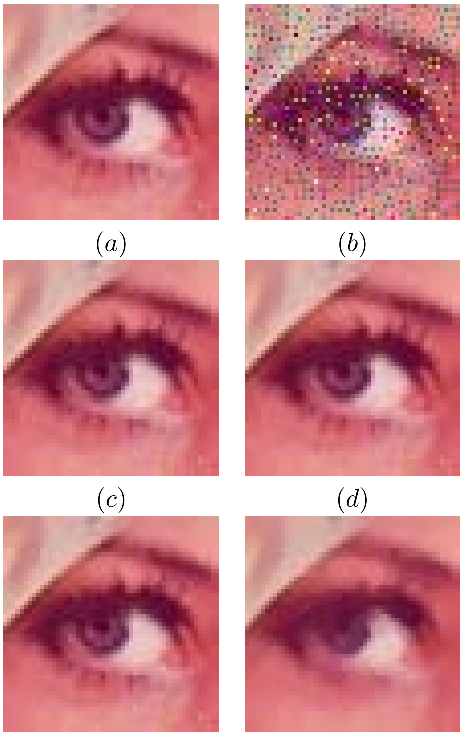

Figure 5. Comparing various algorithms on the standard Lena image (a) The Original zoomed part of the image (b) Corrupted Image (c) Proposed Algorithm (d) Method in [1] (e) AMF [10] (f) VMF [2]

Table 1. Comparing MAE and PSNR values for 10% fixed impulse noise on LENA (Courtesy of [22] and [20])

Method	MAE	PSNR
Noisy Image	7.70	18.65
FTSCF [20]	Not Available	51
Our Method	0.38	44.1
AVFF[11]	0.42	40.29
MICM[15]	0.45	38.72
SANRF[26]	0.53	38.63
FANRF[24]	0.55	38.49
AVLUMS[12]	0.90	37.85
FPGF[23]	0.58	37.5
AVMF[10]	0.61	37.45
VDOSF[16]	0.72	34.72
LIMF[4]	2.32	34.38
VMF[2]	3.41	32.85

Table 1 compares MAE and PSNR values with different algorithms for 10% fixed impulse noise in *lena*. The values depict the superior performance of FTSCF method. Nevertheless, the performance of our algorithm is also optimal because there is a decrease in MAE as well as an increase in PSNR. This behavior is not seen in other algorithms like AVLUMS which show a greater PSNR value than AVMF,FPGF, VDOSF etc but have a higher MAE value.

6. Conclusion

In this paper, we proposed a 2-step solution for impulse noise in color images - a noise detection method based on Hopfield Neural Network followed by a noise filtering method by exploiting the disadvantage of the standard Vector Median Filtering technique. The proposed solution is comparable with the best available filtering schemes and can be applied for the removal of impulse noise in natural images. It is not computationally intensive, so it can be used for real time applications. It was found that the proposed solution works for gray scale images also.

References

[1] N. Alajlana, M. Kamela, and E. Jernigan. Detail preserving impulsive noise removal. *Signal Processing: Image Communication*, 19(10):993–1003, November 2004.

[2] J. Astola, P. Haavisto, and Y. Neuvo. Vector median filters. *Proceedings of the IEEE*, 78(4):678–689, Apr 1990.

[3] M. Bartkowiak and M. Domanski. Vector median filters for processing of color images in various color spaces. *Fifth International Conference on Image Processing and its Applications*, pages 833–836, Jul 1995.

[4] A. Beghdadi and A. Khellaf. A noise-filtering method using a local information measure. *IEEE Transactions on Image Processing*, 6(6):879–882, Jun 1997.

[5] S. Chartier and R. Lepage. Learning and extracting edges from images by a modified hopfield neural network. *16th International Conference on Pattern Recognition*, 3:431–434 vol.3, 2002.

[6] F.Farbiz and M.B.Menhaj. A fuzzy logic control based approach for image filtering. In *Fuzzy Techniques in Image Processing*, volume 52, pages 194–221, Heidelberg, Germany, 2000. Physica Verlag.

[7] R. Hardie and C. Boncelet. Lum filters: A class of rank-order based filters for smoothing and sharpening. *IEEE Transactions on Image Processing*, 41(3):1061–1076, March 1993.

[8] H. K. Kwan and Y. Cai. Fuzzy filters for image filtering. *45th Midwest Symposium on Circuits and Systems*, 3(12):672–675, 2002.

[9] X. Liu, B. Li, Z. Su, X. Shi, and F. Liu. Impulsive noise removal by nonlocal fuzzy diffusion. *Fourth International*

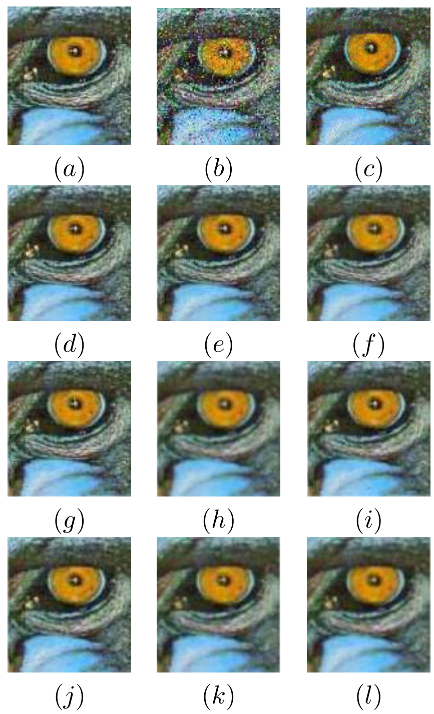

Figure 6. (a)Original Image (b)Corrupted Image (c) Our Method (d) FTSCF [20] (e) FIDRM [19] (f) DSFIRE [18] (g)PWLFIRE [17] (h) HAF [32] (i) LUM [7] (j)TSM [27] (k) MIFCF [6] (l) TMAV [8] (Images Courtesy of [20])

Conference on Image and Graphics, pages 155–159, Aug. 2007.

[10] R. Lukac. Adaptive vector median filtering. *Pattern Recognition Letters*, 24(12):1889–1899, 2003.

[11] R. Lukac, V. Fischer, G. Motyl, and M. Drutarovsky. Adaptive video filtering framework. *IJIST*, 14(6):223–237, 2004.

[12] R. Lukac and S. Marchevsky. Adaptive vector lum smoother. *International Conference on Image Processing*, 1:878–881 vol.1, 2001.

[13] W. Luo and D. Dang. An efficient method for the removal of impulse noise. *IEEE International Conference on Image Processing*, pages 2601–2604, Oct. 2006.

[14] K. Manglem Singh and P. Bora. Adaptive vector median filter for removal impulses from color images. *ISCAS '03. Proceedings of the International Symposium on Circuits and Systems*, 2:II–396–II–399 vol.2, May 2003.

[15] J. Park and L. Kurz. Image enhancement using the modified icm method. *IEEE Transactions on Image Processing*, 5(5):765–771, May 1996.

[16] R.Lukac. Color image filtering by vector directional order statistics. *Pattern Recognition and Image Analysis*, pages 279–285, 2002.

[17] F. Russo. Fire operators for image processing. *Fuzzy Sets and Systems*, 103(2):265–275, April 1999.

[18] F. Russo and G. Rumponi. Removal of impulse noise using a fire filter. *IEEE Transactions on Image Processing*, 1:975–978, 1996.

[19] S. Schulte, V. De Witte, M. Nachtegael, D. Van Der Weken, and E. Kerre. A fuzzy impulse noise detection and reduction method. *IEEE Transactions on Image Processing*, 15(5), 2006.

[20] S. Schulte, V. De Witte, M. Nachtegael, D. Van Der Weken, and E. Kerre. Fuzzy two-step filter for impulse noise reduction from color images. *IEEE Transactions on Image Processing*, 15(11):3567–3578, Nov. 2006.

[21] F.-l. C. Shitong Wang, Yueyang Li and M. Xu. An iterative self-adaptive algorithm to impulse noise filtering for color images. *International Journal of Information Technology*, 11(10), 2005.

[22] B. Smolka. On the adaptive impulsive noise attenuation in color images. *Image Analysis and Recognition*, 19(10):307–317, 2006.

[23] B. Smolka and A. Chydzinski. Fast detection and impulsive noise removal in color images. *Real-Time Imaging*, 11(5-6):389–402, 2005.

[24] B. Smolka, R. Lukac, A. Chydzinski, K. N. Plataniotis, and W. Wojciechowski. Fast adaptive similarity based impulsive noise reduction filter. *Real-Time Imaging*, 9(4):261–276, 2003.

[25] B. Smolka, R. Lukac, and K. Plataniotis. Fast noise reduction in cdna microarray images. *23rd Biennial Symposium on Communications*, pages 348–351, 29 - June 1, 2006.

[26] B. Smolka, K. Plataniotis, A. Chydzinski, M. Szczepanski, A. Venetsanopoulos, and K. Wojciechowski. Self-adaptive algorithm of impulsive noise reduction in color images. *Pattern Recognition*, 35(8):1771–1784, August 2002.

[27] K. M. T.Chen and L. Chen. Tri-state median filter for image denoising. *IEEE Transactions on Image Processing*, 8(12):1834–1838, December 1999.

[28] P. Trahanias, D. Karakos, and A. Venetsanopoulos. Directional processing of color images: theory and experimental results. *IEEE Transactions on Image Processing*, 5(6):868–880, Jun 1996.

[29] K. Valavanis, J. Zheng, and J. Gauch. On impulse noise removal in color images. *IEEE International Conference on Robotics and Automation*, pages 144–149 vol.1, Apr 1991.

[30] T. Viero, K. Oistamo, and Y. Neuvo. Three-dimensional median-related filters for color image sequence filtering. *IEEE Transactions on Circuits and Systems for Video Technology*, 4(2):129–142, 208–10, Apr 1994.

[31] Y. Wakabayashi and A. Taguchi. A new efficient approach for removal of impulse noise for color images. *APCCAS 2006. IEEE Asia Pacific Conference on Circuits and Systems*, pages 85–88, Dec. 2006.

[32] J.-H. Wang and H.-C. Chiu. Haf: An adaptive fuzzy filter for restoring highly corrupted images by histogram estimation. *Proc. Natl. Sci. Counc. ROC(A)*, 23(5):630–643, April 1999.

[33] M. Yu, G. Jiang, and B. Yu. Noise detection based impulse noise removal for color image. *The 2000 IEEE Asia-Pacific Conference on Circuits and Systems*, pages 453–456, 2000.

[34] P. Zvonarev, I. Apalkov, V. Khryashchev, and I. Reznikova. Neural network adaptive switching median filter for the restoration of impulse noise corrupted images. In *ICIAR05*, pages 223–230, 2005.

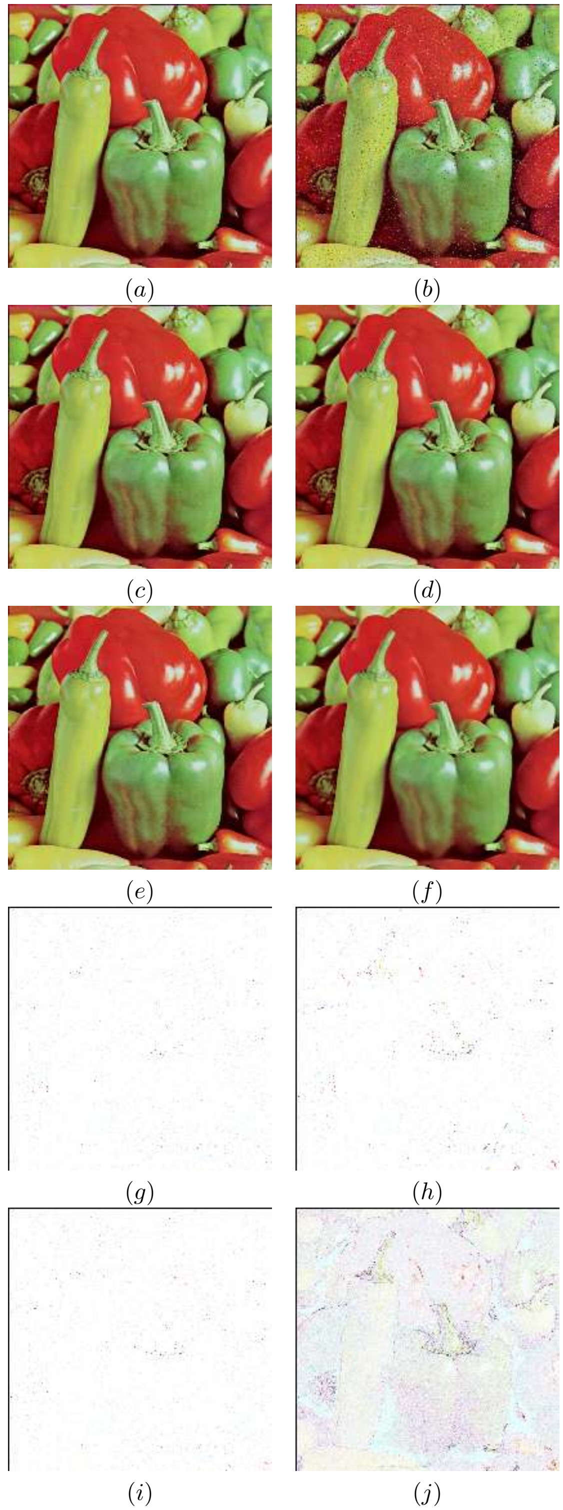

Figure 7. Results comparing different algorithms on Peppers image corrupted with 5% random impulse noise. (a) Original Image (b) Corrupted Image (c) Our Algorithm (d) Method in [1] (e) Adaptive Vector Median Filter [10] (f) VMF [2] (g) Difference Image obtained from our algorithm (h) Difference Image obtained from [1] (i) Difference Image obtained from [10] (j) Difference Image obtained from VMF [2]