A Comparison of Median Filter Techniques For Noise Removal in Color Images

Andreas Koschan and Mongi Abidi

Imaging, Robotics, and Intelligent Systems Laboratory
Department of Electrical and Computer Engineering
The University of Tennessee, Knoxville
koschan@iristown.engr.utk.edu

Abstract. Noise removal or noise suppression is an important task in image processing. In general, the results of the noise removal have a strong influence on the quality of the following image processing techniques. Several techniques for noise removal are well established in gray value image processing. However, it is not easy to adapt these techniques to color image processing. The median filter is often applied to gray value images due to its property of edge preserving smoothing. Extending the median operator to color images is not a simple task since the color vectors have to be sorted according to an order. In this paper, we discuss seven different approaches to median filters for color images. Furthermore, we present results for three selected techniques and we compare the results for these techniques.

1 Introduction

Noise suppression or noise removal is an important task in image processing. The median filter is often applied to gray value images due to its property of edge preserving smoothing. The median filter is a nonlinear operator that arranges the pixels in a local window according to the size of their intensity values and replaces the value of the pixel in the result image by the middle value in this order. The extension of the concept of scalar median filtering to color image processing is not a simple procedure. One essential difficulty in defining the median in a set of vector values is the lack of a "natural" concept of rank regarding vectors. The problems occurring here are outlined based on the following example.

Consider the following example: three color pixels in the *RGB* color space are defined by the three color vectors $\mathbf{p}_1 = (10,40,50)^T$, $\mathbf{p}_2 = (80,50,10)^T$, and $\mathbf{p}_3 = (50,100,150)^T$. If the median filter is applied separately to each vector component, the resulting vector is $\mathbf{p}' = (50,50,50)^T$. This vector does not exist in the input data and it represents an achromatic gray value. Furthermore, the separate median filtering may cause different shifts of the local maxima in the particulars color components [12]. This problem is illustrated in the following on the basis of a one-dimensional linear vector signal with two entries (see Fig. 1).

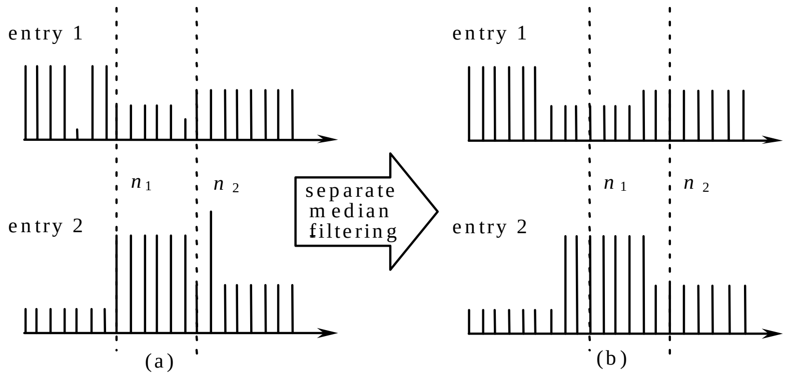

Figure 1: The edge jitter problem caused by separate median filtering (after [12]).

The original signal represents three segments: left band bounded-right at n_1, the middle band bounded-left by n_1 and bounded-right by n_2, and right band bounded-left by n_2. Both entries are contaminated by impulse noise: one at the position $n_1 - 3$ and the other at $n_2 + 1$ (see Fig. 1(a)). The result of separate filtering with a window size of five pixels is shown in Fig. 1(b). The impulse at $n_1 - 1$ in entry 1 is reduced and the distinction, which separates two segments in the original image, is shifted by one pixel position to the left. A shift also occurs at n_2 in entry 2. From this observation, one may conclude that individual processing does not remove the impulse noise. Instead, it moves the noise position to affect its neighbor values in the image [12].

2 Median Techniques for Color Images

The example specified above shows that both a color distortion and the loss of the characteristic of edge preservation may occur when the median filter is applied separately to each single component of the color vectors. If however all three color components are regarded at the same time, then an order must be defined for the sequence of color vectors that defines, for example, when a vector is larger or smaller than another vector. Several techniques have been proposed for median filtering in color images, e.g.:

a) an adaptive scalar median filter [9],
b) a vector median filter (with weighting [11] and without weighting [1]),
c) a reduced vector median filter [6],
d) a median filter applied to chromaticity in the *HSI* space [3],
e) a median filter based on conditional ordering in the *HSV* space [10], and
f) a vector directional filters [5, 7, 8].

Furthermore, vector median filters can be connected to morphological operations considering a lexicographic order. Detailed information about the mathematical theory on this connection may be found in [2]. In the following, we discuss the seven techniques mentioned above under a) to f).

2.1 Adaptive Scalar Median

Valavanis et al. [9] presented a color-oriented modification of the median filter using scalar medians. The algorithm is quite heuristic and requires a more exact explanation. For this some notations are specified. Consider a color image **C** in the *RGB* color space. It holds

$$\mathbf{C} = \begin{pmatrix} R \\ G \\ B \end{pmatrix} \quad \text{or rather} \quad C(x,y) = \begin{pmatrix} R(x,y) \\ G(x,y) \\ B(x,y) \end{pmatrix},$$

where $\mathbf{C}(x,y)$ is the pixel vector at location (*x*,*y*) in the color image **C**. Furthermore, denote

$$\mathbf{C}(x_R, y_R) = med_R\{\mathbf{C}(x,y) \mid x, y \in W\},$$
$$\mathbf{C}(x_G, y_G) = med_G\{\mathbf{C}(x,y) \mid x, y \in W\},$$
$$\mathbf{C}(x_B, y_B) = med_B\{\mathbf{C}(x,y) \mid x, y \in W\},$$

where *W* is a window of odd size. $\mathbf{C}(x_R, y_R)$, $\mathbf{C}(x_G, y_G)$, and $\mathbf{C}(x_B, y_B)$ are the coordinates of the median values for the respective color channel within the window *W* and $med_R\{\ \}$, $med_G\{\ \}$, and $med_B\{\ \}$ are the median operators for the computation of the resulting *RGB* values. Since only one component of the vector is regarded at one time, three vectors result. Each of those vectors contains a median element and two assigned elements. The three vectors can be combined to a "median matrix" **M** given by

$$\mathbf{M} = \left(\mathbf{C}(x_R, y_R), \mathbf{C}(x_G, y_G), \mathbf{C}(x_B, y_B)\right)$$
$$= \begin{pmatrix} R(x_R, y_R) & R(x_G, y_G) & R(x_B, y_B) \\ G(x_R, y_R) & G(x_G, y_G) & G(x_B, y_B) \\ B(x_R, y_R) & B(x_G, y_G) & B(x_B, y_B) \end{pmatrix}.$$

Every column in **M** represents a real pixel vector in the original image. A monochromatically oriented (separate) median filtering in each vector component yields a new “median“ as $\left(R(x_R, y_R), G(x_R, y_R), B(x_B, y_B)\right)^T$ consisting of the diagonal elements of **M**. This vector does not necessarily exist in the input image and its use may cause color distortions in the filtered image. Since color distortions in the image have a direct influence on the color perception of the viewer, perception-referred criteria for the selection of the "most suitable" candidate are consulted there. For this a representation in the intuitive perception adapted *HSI* color space is used. The abstract quantitative values of a single vector pixel are named *h*, *s*, and *i*.

$$h = H(R,G,B),\ s = S(R,G,B), \text{ and } i = I(R,G,B). \qquad (1)$$

The transformation of the *RGB* values into a *hsi*-representation constitutes an ill-posed problem. Non-removable singularities can occur. To decrease the probability for the occurrence of these disturbances, Zheng et al. [12] suggest the following determinations for *h*, *s* and *i*:

$$h = H(\bar{r}, \bar{g}, \bar{b}),\ s = S(\bar{r}, \bar{g}, \bar{b}), \text{ and } i = I(\bar{r}, \bar{g}, \bar{b}), \tag{2}$$

where $\bar{r}, \bar{g}$, and $\bar{b}$ denote the mean values for the red, green and blue channel within a window. In order to minimize the distortions in color appearance in a processed image, the following criteria are proposed in [12]:

(1) The hue changes should be minimized.
(2) The shift of saturation should be as small as possible and it is better to increase than to decrease the saturation.
(3) It is desirable to maximize the (relative) luminance contrast.

The three criteria are not examined at the same time, but successively, which is in some way similar to a conditional ordering. For this purpose, candidates are regarded according to the first criterion:

$$(r_l, g_m, b_n)^T = (r_i, g_j, b_k)^T \text{ if } \quad min\{H(r_i, g_j, b_k) - h\}, \tag{3}$$

where $l, m, n \in \{i, j, k \mid i, j, k = 1,2,3\}$. The notation $min\{\cdot\}$ denotes the minimum value of the difference measurement. If several possible vector medians possess the same hue and are similar to the overall average hue in the region, then the second criterion is applied:

$$(r_x, g_y, b_z)^T = (r_l, g_m, b_n)^T \text{ if } \quad min\{S(r_l, g_m, b_n) - s\}, \tag{4}$$

where $x, y, z \in \{l, m, n \mid l, m, n = 1,2,3\}$. Finally the vector median with the largest intensity value is selected.

The search, described above, for a "heuristically perception-optimal" result based on the individual values of the median filters preserves, as far as possible, the characteristics of the median filtering and causes no large color disturbances.

2.2 Vector Median Filter

One important characteristic of the median filter consists of the fact that it does not produce values for a pixel that do not exist in the original image. This characteristic is not always guaranteed during the adaptive scalar median filtering. One way to overcome this problem consists in the application of a vector median filter [4]. For a set of *N* vectors $x_1, \ldots, x_N$ within a right-angled window and any vector norm $\|\cdot\|_L$ the *vector median filter* VM is defined by the equation (see [1])

$$VM\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\} = \mathbf{x}_{VM},$$

where

$$\mathbf{x}_{VM} \in \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$$

and

$$\sum_{i=1}^{N} \left\| \mathbf{x}_{VM} - \mathbf{x}_i \right\|_L \leq \sum_{i=1}^{N} \left\| \mathbf{x}_j - \mathbf{x}_i \right\|_L , \qquad j = 1,2,\ldots,N. \tag{5}$$

The result of this filter operation selects that vector in the window, which minimizes the sum of the distances to the other N -1 vectors regarding the L-norm. Additionally, weightings can be specified for the vector median filter. Generally distance weights $w_i,\ i = 1,\ldots,N$ and component weights $v_i,\ i = 1,\ldots,N$ can be defined. The result of the weighted vector median filter is the vector $\mathbf{x}_{WVM}$ with

$$\mathbf{x}_{WVM} \in \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$$

and

$$\sum_{i=1}^{N} w_i \left\| v_i \otimes (\mathbf{x}_{WVM} - \mathbf{x}_i) \right\|_L \leq \sum_{i=1}^{N} w_i \left\| v_i \otimes (\mathbf{x}_j - \mathbf{x}_i) \right\|_L , \quad j = 1,2,\ldots,N. \tag{6}$$

The point-wise multiplication is denoted with $\otimes$, i.e. if $c = a \otimes b$ then $c_i = a_i \cdot b_i$ holds for all vector components. If several vectors fulfill equations (5) and/or (6), then that vector that is closest to the vector in the center of the window, is selected. The result of these filter operations is on one hand not unambiguous and beyond that also dependent on the selected L-norm. Additionally, the weightings have an influence on the result when the weighted filter is applied. Thus, weightings must be selected very carefully; otherwise the characteristic of edge preservation can be lost. In each constellation it is however guaranteed that the filtering produces no additional new color vector. An investigation on the influence of the weighting on the processing result is to be found in [11].

2.3 Reduced Vector Median Filter

The computation of the vector median for the entire color image is quite time-consuming. Regazzoni and Teschioni [6] suggest an approximation of the vector median filtering, which they call “reduced vector median filtering” (RVMF). They use for this so-called “space filling curves”, as are also partly used with scanners, in order to map the three-dimensional color vectors into a one-dimensional space. In this one-dimensional space the median is then determined in conventional way (as in gray value images). A detailed representation of the RVMF technique may be found in [6]. Due to [6] the signal to noise ratios related to the non-distorted original images and the filtered distorted images are similar to those of the original vector median filtering. The signal to noise ratios for the reduced vector median filter are however both for Gaussian noise and for impulse noise always worse than the values for the "original" vector median filter.

2.4 Median Filter Applied to the Chromaticity in the *HSI* Space

The difficulty of the definition of an order of rank between the color vectors arises also if a representation in the *HSI* color space is selected. Here the hue is indicated as angle and the ranking is likewise not easily be specified. Frey [3] suggested a procedure that works in the *HSI* model and already comes very close to a kind of median filter. He searches for the mean value in the chromaticity plane and guarantees thus that the value in the output image is identical with a value in the regarded window in the input image. This procedure is a variant of the vector median described above and exclusively works on the chromaticity.

Let the chromaticity be defined as a complex function with $j = \sqrt{-1}$, where the hue $t(x, y)$ is noted as phase and the saturation $s(x, y)$ is noted as absolute value (see [3]). The chromaticity $b(x, y)$ is given by

$$b(x, y) = s(x, y) \cdot e^{jt(x,y)} . \tag{7}$$

The real part ($\Re$) and the imaginary part ($\Im$) of $b(x, y)$ can be computed due to Euler's formula by

$$\Re\big(b(x, y)\big) = s(x, y) \cdot \cos(t(x, y)) , \tag{8}$$

$$\Im\big(b(x, y)\big) = s(x, y) \cdot \sin(t(x, y)) , \tag{9}$$

The chromaticity $b(x, y)$ is defined by

$$b(x, y) = \Re\big(b(x, y)\big) + j \cdot \Im\big(b(x, y)\big) \text{ or} \tag{10}$$

$$b(x, y) = s(x, y) \cdot \cos\big(t(x, y)\big) + j \cdot s(x, y) \cdot \sin\big(t(x, y)\big). \tag{11}$$

That particular pixel in the window is looked for, for which the sum of the squared distances to all other pixels of the window is minimal. In the output image this chromaticity is then registered for the pixel. For a window of size $m \times n$ with $k = m \cdot n$ pixels, the squared Euclidean distance d_{ij}^2 of the pixel *i* to the pixel *j* is given by

$$d_{ij}^2 = (\Re(b_i) - \Re(b_j))^2 + (\Im(b_i) - \Im(b_j))^2 \quad \text{with} \quad 1 \le i, j \le k. \tag{12}$$

The sum of the squared distances d_i^2 of pixel *i* to all others is denoted as

$$d_i^2 = \sum_{j=1}^{k} d_{ij}^2 \text{ for } i \ne j . \tag{13}$$

The chromaticity of that pixel, which fulfills $d^2 = \min\{d_i^2\}$ is selected for the result image. Just as with the vector median this minimum is not always unambiguous. If several pixels fulfill the above condition, then that value that is most similar to the original value, is selected for the chromaticity.

2.5 Median Filter Based on Conditional Ordering in the *HSV* Space

In *conditional ordering,* vectors are first ordered according to the ordered values of one of the components, e.g. of the first component. Afterwards, vectors having the same value for the first component are ordered according to the ordered values of another component, e.g. the second component, etc. A color **c** in the *HSV* space is denoted by **c**(*h*,*s*,*v*) with the *hue* value $h \in [0,360)$, the *saturation* value $s \in [0,1]$, and the *value* $v \in [0,1]$. Vardavoulia et al. [10] suggest the following ordering of *HSV* space vectors:

(1) First vectors are sorted from those with smallest *v* to those with greatest *v*.

(2) Vectors having the same value of *v* are then sorted from those with greatest *s* to those with smallest *s*.

(3) Finally, vectors having the same values of *s* and *v* are sorted from those with smallest *h* to those with greatest *h*.

Using mathematical notation for the three ordering criteria mentioned above, two operators $<_c$ and $=_c$ (see [10]) can be defined for two colors by

$$\begin{aligned}\mathbf{c}(h_i, s_i, v_i) <_c \mathbf{c}(h_k, s_k, vk) \Leftrightarrow \\ (v_i < v_k) \vee (v_i = v_k \wedge s_i > s_k) \vee (v_i = v_k \wedge s_i = s_k \wedge h_i < h_k),\end{aligned} \tag{14}$$

$$\mathbf{c}(h_i, s_i, v_i) =_c \mathbf{c}(h_k, s_k, vk) \Leftrightarrow (v_i = v_k) \wedge (s_i = s_k) \wedge (h_i = h_k)\,. \tag{15}$$

The *n* color vectors $\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_n$ representing the colors of the pixels inside a window are placed in ascending order, forming the set of ordered values $\{\mathbf{c}_1, \mathbf{c}_2, \ldots, \mathbf{c}_n\}$, in which $\mathbf{c}_1 <_c \mathbf{c}_2 <_c \ldots <_c \mathbf{c}_n$. The middle vector in this order is called the vector median and is denoted by $vmed_{HSV}$. In the case of a gray value image the median is identical to the conventional definition of a median.

2.6 Vector Directional Filters

Vector directional filters (VDF) are a class of multivariate filters that are based on polar coordinates and vector ordering principles considering the angle between the color image vectors as ordering criterion [5, 7, 8]. The class of VDFs operates on the direction of the image vectors with the objective of eliminating vectors with atypical directions (large chromaticity errors) in the vector space. A detailed investigation on the statistic characteristics of vector direction filters is to be found in [5, 7]. Similar to the median filter applied to the chromaticity (see 2.4) the VDFs operate on the chromaticity components of a color. In other words, they are designed to detect chromaticity errors, but not intensity outliers.

3 Experiments and Results

Since noise removal techniques are designed to enhance the image quality, we discuss their performance regarding three criteria. Criterion 1 considers the quality of the resulting color image based on its visual impression. Criterion 2 considers the quantity of the removed noise and criterion 3 considers the computational cost of the technique. The three techniques mentioned in 2.1, 2.2, and 2.4 have been applied to real color images with different noise distortions. In Fig. 3 the results of a filtering applying the vector median and the adaptive scalar median to a test image disturbed by impulse noise are visualized. The color impulse noise is controlled via a noise rate and a noise impulse height. The noise rate indicates how many pixels of a color component are altered. The noise impulse height indicates the absolute value, by which the pixel concerned is changed. In the example represented in Fig. 3 the noise rate is equal to 11 and the noise impulse height is equal to 66. Thus each eleventh pixel in a color component is altered by around $\pm$ 66 values. To enhance the visibility of the differences in the filtering results, Fig. 3 shows in addition the difference images (intensified by the factor three) between the original image and both filtered images. It is to be recognized that a stronger noise reduction occurs when applying the vector median to the image than when applying the adaptive scalar median.

This applies in general also to different noise rates. In Fig. 2 the interpolated standard deviations for the differences between original image and filtered image (by means of adaptive scalar median, vector median and chromaticity median) are indicated for different noise rates. The noise impulse height always amounts to 66 values. Fig. 2 shows the computed values for two test images in (a) and (b). With all three techniques the results improve with increasing noise rate. The worst results were obtained for both test images with the adaptive scalar median. These results are not representative however. Already with two images better results are obtained in the case of a small noise rate, sometimes with the chromaticity median and sometimes with the vector median, depending upon image content.

Regarding criterion 1 the vector median performed best in our experiments, while no significant differences between VM and $vmed_{HSI}$ could be recognized in [10]. Regarding criterion 2 the adaptive scalar median performed worst in our experiments, while the performance of the other two filters were dependent on the noise. Due to [10] $vmed_{HSI}$ outperforms VM and VDFs based on a noise model in the *HSI* space, while better signal-noise-ratios are reported for VDFs than for VM in [7, 8]. Regarding criterion 3 the vector median is the computationally most costly operator in our experiments. Its actual cost is dependent on the vector norm. Since most of the other operators are designed for a special color space, their costs also include the transformations between the selected color spaces.

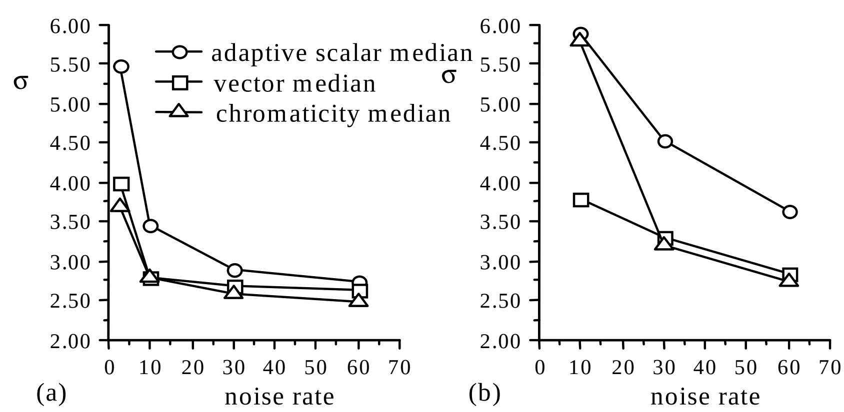

Figure 2: Interpolated standard deviations for the differences between original image and filtered image are indicated for different noise rates and two test images in (a) and (b) respectively. The noise impulse height always amounts to 66 values.

4 Conclusion

The extension of the concept of scalar median filtering to color image processing is not a simple procedure. In this paper, we discussed seven different approaches to median filters for color images. Furthermore, we presented results for three selected techniques and we compared the results for these techniques. So far there does not exist any best technique. The performance of the operators depends on the image content and the kind of noise with which they are degraded. Moreover, the selection of a color space may also influence the selection of a median operator, since most of the operators are designed for a special color space. Furthermore, dependent on the application some requirements may exist regarding the visual appearance of the results or regarding the computational efficiency. In our comparison the vector median filter performed best. This filter is however the most costly operator presented in this paper.

Acknowledgement

We thank Peter Hannemann from Technical University Berlin for implementing some of the algorithms. In addition, this work was supported by the University Research Program in Robotics under grant DOE-DE-FG02-86NE37968, by the DOD/TACOM/NAC/ARC Program, R01-1344-18, and by FAA/NSSA Program, R01-1344-48/49.

References

1. Argenti, F., Barni, M., Cappellini, V., Mecocci, A.: Vector Median Deblurring Filter For Color Image Restoration. Electronics Letters **27** (1991) 1899-1900.
2. Caselles, V. Sapiro, G., Chung, D.H.: Vector Median Filters, Inf-Sup Operations, and Coupled PDE's: Theoretical Connections. J. of Math. Imaging and Vision **12** (2000) 109-120.
3. Frey, H.: Digitale Bildverarbeitung in Farbräumen. Doktorarbeit (Ph.D. thesis), University Ulm, 1988.
4. Pitas, I., Tsalides, P.: Multivariate Ordering in Color Image Filtering. IEEE Trans. on Circuits and Systems for Video Technology **1** (1991) 247-259.
5. Plataniotis, K.N., Venetsanopoulos, A.N.: Color Image Processing and Applications. Springer-Verlag, Berlin Heidelberg New York, (2000).
6. Regazzoni, C.S., Teschioni, A.: A New Approach to Vector Median Filtering Based on Space Filling Curves. IEEE Trans. on Image Processing **6** (1997) 1025-1037.
7. Trahanias, P.E., Karakos, D., Venetsanopoulos, A.N.: Directional Processing of Color Images: Theory and Experimental Results. IEEE Trans. on Image Processing **5** (1996) 868-880.
8. Trahanias, P.E., Venetsanopoulos, A.N.: Vector Directional Filters - a New Class of Multichannel Image Processing Filters. IEEE Trans. on Image Processing **2** (1993) 528-534.
9. Valavanis, K.P., Zheng, J., Gauch, J.M.: On Impulse Noise Removal in Color Images. In: Proc. Int. Conf. on Robotics and Automation, Sacramento, Ca., (1991) 144-149.
10. Vardavoulia, M.I., Andreadis, I., Tsalides, Ph.: A New Vector Median Filter for Colour Image Processing. Pattern Recognition Letters 22 (2001) 675-689.
11. Wichman, R., Öistämö, K., Liu, Q., Grundström, M., Neuvo, Y.: Weighted Vector Median Operation for Filtering Multispectral Data. In: Proc. SPIE **1818**, Visual Communications and Image Processing (1992) 376-383.
12. Zheng, J., Valavanis, K.P., Gauch, J.M.: Noise Removal from Color Images. J. Intelligent and Robotic Systems **7** (1993), 257-285.

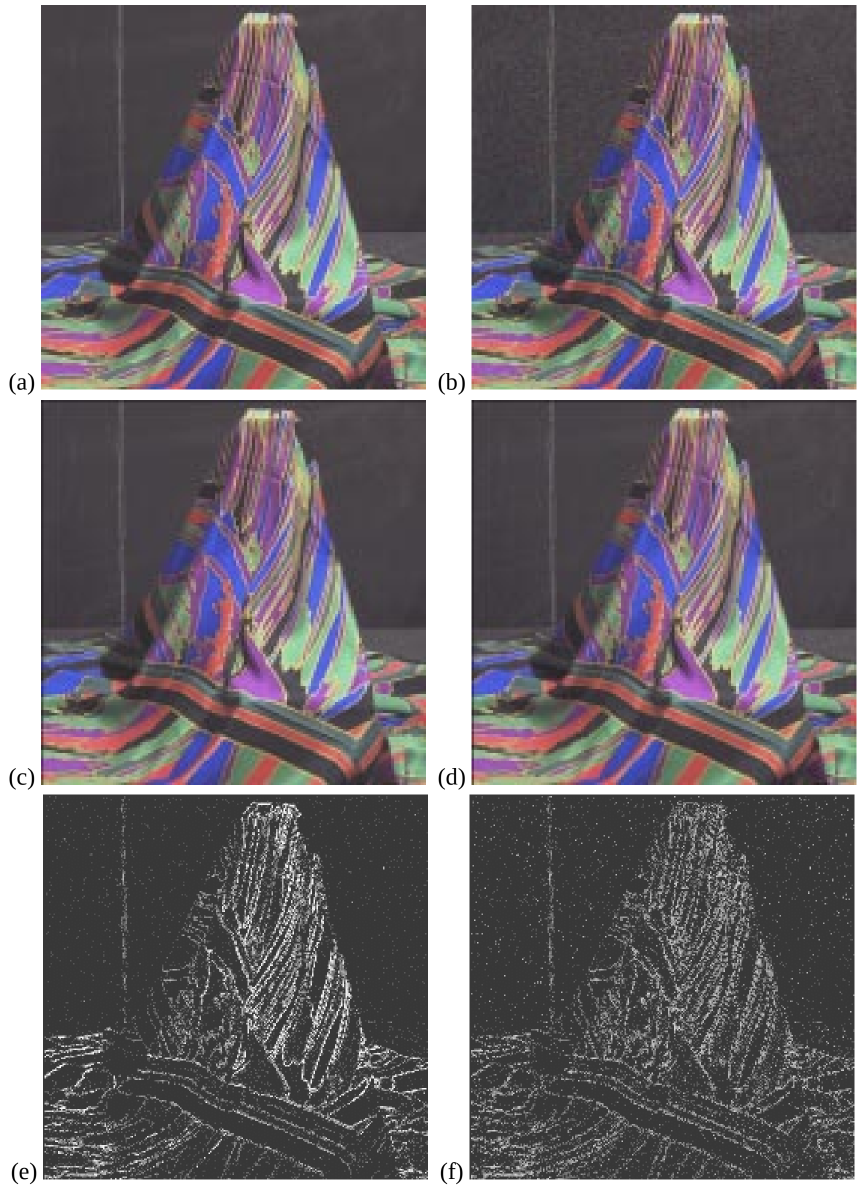

Figure 3: Results of median filtering: (a) the original color image "SHAWL", (b) the disturbed color image, (c) the result of vector median filtering, (d) the result of applying the adaptive scalar median filter to the color image, and (intensified by the factor three) (e) the difference image between the original image and the vector median filtered image and (f) the difference image between the original image and the adaptive scalar filtered image.