Transactions Briefs

Color Image Processing Using Adaptive Vector Directional Filters

K. N. Plataniotis, D. Androutsos, and A. N. Venetsanopoulos

***Abstract*—A new class of filters for multichannel image processing is introduced and analyzed in this brief. This class constitutes a generalization of vector directional filters. The proposed filters use fuzzy transformations of the angles among the different vectors to adapt to local data in the image. The principle behind the new filters is explained and comparisons with other popular nonlinear filters are provided. The specific case of color image processing is studied as an important example of multichannel image processing. Simulation results indicate that the new filters offer some flexibility and have excellent performance.**

***Index Terms*—Adaptive filters, color image processing, fuzzy membership functions, vector directional filters.**

I. Introduction

Filtering of multichannel images has received increased attention recently due to its importance in processing color images. Numerous filtering techniques have been proposed to date for multichannel image processing [1]–[3]. Nonlinear filters applied to images are required to preserve edges and details, and remove impulsive and Gaussian noise. On the other hand, vector processing is one of the most effective methods available to filter noise and detect edges on multichannel images. Rank, and especially median filters, have been used extensively as multichannel image filters [4]. A new class of filters for processing vector-valued signals was introduced in [5]. The so-called vector directional filter (VDF) uses the angle between the image vectors as an ordering criterion. The VDF operates on the direction of the image vectors, separating in this way the processing of vector data into *directional processing* and *magnitude processing*. Thus, it preserves chromaticity while removing impulsive noise [6].

In this brief, a new class of VDF's is introduced. Fuzzy membership functions based on an angle criterion are adopted to determine the weights of an adaptive weighted mean filter [1], [7], [8]. Our objective is to develop a computationally efficient multichannel filter, which will have good performance without requiring any *a priori* knowledge about the signal and noise characteristics.

II. Adaptive VDFs

A. The Filter Structure

Let $y(x)$: $Z^l \rightarrow Z^m$, represent a multichannel image and let $W \in Z^l$ be a window of finite size n (filter length). Usually we consider a square window ($n = N1 \times N1$) centered around the current pixel. The noisy image pixels inside the window W are denoted as $\mathbf{x}_j$, $j = 1, 2, \cdots, n$.

Since the most commonly used methodology to decrease the level of random noise present in the signal is smoothing, we select an averaging operation in order to replace the noisy vector at the window center with a suitable representative vector. The general form of the adaptive filter proposed here is given as a fuzzy weighted average of the input vectors inside the window W. Thus, the uncorrupted multichannel signal is estimated by determining the centroid as follows:

$$\hat{\mathbf{y}} = \sum_{j=1}^{n} w_j \mathbf{x}_j = \sum_{j=1}^{n} \frac{\mu_j^\lambda}{\sum_{j=1}^{n} \mu_j^\lambda} \mathbf{x}_j \tag{1}$$

where λ is a parameter such that $\lambda \in [0, \infty)$. According to (1), the output of the proposed filter is a weighted average of all the vectors inside the operation window. The weighting coefficients are transformations of the sum of distances between the center of the window (pixel under consideration) and all input vectors inside the filter window. In multichannel filtering, it is desirable to perform smoothing on all vectors which are from the same region as the vector at the window center. Thus, the adaptive weights should represent the confidence that the vectors under consideration come from the same region. It is, therefore, reasonable to make the weights in (1) proportional to the difference in terms of a distance measure between a given vector and its neighbors inside the filtering window. The design objective is: "*Assign the maximum weight to the vector which is most centrally located inside the processing window*." Thus, the vector with the maximum weight will be the one which has the minimum distance from all the other vectors. In this way, atypical vectors due to noise or missing components (outliers), which are placed far away from the centermost vector, will be assigned smaller weights and will contribute less to the final output. The weighting transformation is essentially a membership function with respect to the specific window component. In accordance to our design objective it is reasonable to select an appropriate fuzzy transform so that the vector with the minimum distance will be assigned the maximum weight. The membership function value can be regarded as the comparison of the vector under consideration with the ideal vector which results in a distance. Thus, the degree of membership is a function of the distance, which can be defined as follows [10]:

$$\mu_j = \frac{1}{1 + d(\mathbf{x}_j)} \tag{2}$$

where $d(.)$ is the distance function yet to be determined. If the vector under consideration $\mathbf{x}_j$ has all the features of the ideal vector, the distance should be zero resulting in $\mu_j \rightarrow 1$, otherwise, if no similarity between the ideal and the vector $\mathbf{x}_j$ exists, the distance shall be ∞ with $\mu_j \rightarrow 0$.

B. Directional Distances

It is evident that the value of the membership function depends on the choice of the distance criterion selected as a measure of dissimilarity. Any distance criterion used to calculate distances among multichannel signals can be utilized. Our primary objective, however, is to apply the new filter to color images, thus a criterion suitable to

Manuscript received August 23, 1995; revised September 11, 1996. This paper was recommended by Associate Editor W.-C. Siu.

K. N. Plataniotis was with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, M5S 3G4, Canada. He is now with the Department of Mathematics, Physics and Computer Science, Ryerson Polytechnic University, Toronto, ON, M5B 2K3 Canada (e-mail: kostas@dsp.toronto.edu).

D. Androutsos and A. N. Venetsanopoulos are with the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, M5S 3G4 Canada (e-mail: zeus@dsp.toronto.edu; anv@dsp.toronto.edu).

Publisher Item Identifier S 1057-7130(98)02143-0.

TABLE I
ANGULAR DISTANCES

Absolute Distance	$a_i = \sum_{j=1}^{n} A(x_i, x_j)$
Average Absolute Distance	$\tilde{a}_i = \frac{1}{n}\sum_{j=1}^{n} A(x_i, x_j)$
Square Distance	$a2_i = \sum_{j=1}^{n} A^2(x_i, x_j)$
Average Square Distance	$\tilde{a2}_i = \frac{1}{n}\sum_{j=1}^{n} A^2(x_i, x_j)$

measure distances among color vectors is used. The distance measure selected is the *vector angle criterion*. This criterion considers the angle between two vectors as their distance. A scalar quantity

$$a_j = \sum_{i=1}^{n} A(\mathbf{x}_i, \mathbf{x}_j) \tag{3}$$

with

$$A(\mathbf{x}_i, \mathbf{x}_j) = \cos^{-1}\left(\frac{\mathbf{x}_i \mathbf{x}_j^t}{|\mathbf{x}_i||\mathbf{x}_j|}\right) \tag{4}$$

is the distance associated with the noisy vector $\mathbf{x}_j$ inside the processing window of length n. We can consider the distance in (3) as the sum of absolute angular distance for spherical data. This criterion was first introduced in [5] to measure distances among color vectors. Since in the *RGB* color space, color is defined as relative values in the trichromatic channel and not as a triplet of absolute intensity values, it was argued in [5] that the distance measure must respond to relative intensity differences (chromaticity) and not absolute intensity differences (luminance). Therefore, the orientation difference between two color vectors was selected in [5] as their distance measure because it correlates well with their spectral ratio difference. Based on the angle between two vectors, a number of other useful distance measures can be defined (see Table I). Any one of the four angular distance measures listed in Table I can be used to generate the weights in the adaptive design of (1).

Utilizing our general membership function and taking into account the fact that the relationship between a distance measured in units and perception is generally exponential, a sigmoidal (piecewise) linear membership function is appropriate [11]. In such a case the fuzzy weight μ_j associated with the vector $\mathbf{x}_j$ can take the following form:

$$\mu_j = \frac{\beta}{[1 + \exp(\alpha_j)]^r} \tag{5}$$

where $\beta,\ r = 1/\sigma$ are parameters to be determined and α_i is any angular distance measure listed in Table I.

The parameter β is a soft parameter used only to adjust the limit of the S-shaped membership function (weight scale threshold). In our experiments we assigned the value $\beta = 2$, since we decided to use membership functions that deliver an output in the interval [0, 1]. The parameter r is the smoothing parameter. Since, by definition, the angle distance measure in (4) delivers a positive number in the interval [0, π] [5], the output of the fuzzy transformation introduced above produces a membership value in $[\beta/\{1 + \exp[(n\pi)]\}^r,\ \beta/2]$. It can easily be seen that for a moderate size window, such a 3 × 3 window we can consider the above membership function as having values in the interval (0, 1], e.g., [1.4 × 10^{-12}, 1] for the angular distance of (3) with parameters $r = 1$ and $\beta = 2$.

As one should expect, the function used here does not change too much when its values are around the minimum distance (region of confidence) and does not increase quickly when the membership function's input values are around ∞, the region of rejection.

The fuzzy transformation of (5) is not unique. Other commonly used shapes, such as triangular or Gaussian-like curves can be used instead. However, despite past efforts, a unified form of fuzzy membership function has not yet been shown. Since its choice is very much problem dependent, the only applicable *a priori* rule is that the designer must confine himself to the functions which are continuous and monotonic [12].

C. The Adaptive Filters

The weighted average form of (1) it can be seen in the context of fuzzy systems as a generalized defuzzification process, where a defuzzified value is selected as output based on a probability-like distribution obtained in the fuzzification step.

In fuzzy set theory defuzzification is realized by a decision-making algorithm which selects the best crisp value based on a fuzzy set. The defuzzification process can be considered as a two step process. The first step is the conversion of the membership values associated with the fuzzy inputs into a probability-like distribution which satisfies the identity and monotonicity conditions. In the second step of the defuzzification process the defuzzified value is selected based on values from the probability-like distribution [13].

It is not hard to see that through the definitions in (1), (3)–(5) the adaptive weights $w_j = \mu_j / \sum_{j=1}^{n} \mu_j$ satisfy the above conditions. Specifically:

1) $\forall i, j$, if $\mu_i = \mu_j$ then $w_i = w_j$, (identity);
2) $\forall i, j$, if $\mu_i > \mu_j$ then $w_i > w_j$, (monotonicity).

Furthermore, through the normalization procedure, our weights w_j in (1) have the basic properties which characterize any probability distribution, namely:

1) $w_j > 0$ and $w_j \in [0, 1]$;
2) $\sum_{j=1}^{n} w_j = 1$.

Given the satisfaction of the above conditions the defuzzification strategy can be seen as a form of pseudo-expectation formula [12], [13].

Indeed, from the generalized defuzzification rule of (1) if $\lambda = 1$ the widely used "*Center of Area*" (COA) defuzzification strategy (Centroid Defuzzifier) can be obtained. According to the COA defuzzification the defuzzified value of the adaptive fuzzy filter, hereafter *Adaptive Vector Directional Filter* (AVDF), is given as:

$$\hat{\mathbf{y}} = \sum_{j=1}^{n} w_j \mathbf{x}_j = \sum_{j=1}^{n} \frac{\mu_j}{\sum_{j=1}^{n} \mu_j} \mathbf{x}_j. \tag{6}$$

The AVDF obtained through the COA strategy generates a vector valued signal, which is not included among the original set of input vectors. Controlling the parameter r in (5), we can adjust the smoothness of the output. As a general rule, smaller values of r can smooth out noisy vectors, while larger values can make the overall output as nonlinear as required to prevent details and to discard impulsive-type noise. Thus, the parameter should be properly chosen to provide a balance between smoothing and detail preservation. Since r is one-dimensional parameter, it is not difficult to determine an appropriate value for practical applications. In most cases, a few trial-and-error procedures may be enough to determine a good value. Moreover, our experience indicates that acceptable results can be obtained by assigning the value $r = 1$.

However, if the output of the adaptive fuzzy filter is required to be part of the original input set, a different defuzzification strategy should be used. Defining $\mu_{(\max)}$ as the largest membership value obtained through (5), the adaptive weights in (1) can be rewritten as follows:

$$w_j = \frac{\mu_j^\lambda}{\sum_{j=1}^{n} \mu_j^\lambda} = \frac{\frac{\mu_j^\lambda}{\mu_{(\max)}^\lambda}}{\sum_{j=1}^{n} \frac{\mu_j^\lambda}{\mu_{(\max)}^\lambda}} = \frac{\left[\frac{\mu_j}{\mu_{(\max)}}\right]^\lambda}{\sum_{j=1}^{n} \left[\frac{\mu_j}{\mu_{(\max)}}\right]^\lambda}. \tag{7}$$

Given that $\mu_j < \mu_{(\max)}$, as $\lambda \to \infty$ then

$$\begin{aligned} w_j &= 1, \quad \mu_j = \mu_{(\max)} \\ &= 0, \quad \mu_j \neq \mu_{(\max)}. \end{aligned} \tag{8}$$

Equation (8) represents the "*maximum defuzzifier*" strategy. Since given the form of the function in (3)–(5) the maximum value occurs at a single point only, the maximum defuzzifier strategy coincides with the "*mean of maxima*" (MOM) defuzzification process. Through the maximum defuzzifier the output of the adaptive fuzzy filter is defined as:

$$\hat{\mathbf{y}} = \mathbf{x}_j, \quad \mu_j = \mu_{(\max)}. \tag{9}$$

In other words, we single out the input vector associated with the maximum fuzzy weight. We name this filter *maximum adaptive vector directional filter* (MAVDF). It must be emphasized that through the fuzzy membership function, the maximum fuzzy weight corresponds to the minimum distance. If the absolute angle distance is used as the dissimilarity function the new fuzzy filter coincides with the spherical median and delivers the same output as the basic VDF (BVDF) [5], [6]. The MADVF filter defined above is a pure *chromaticity* filter in the case of color image filtering. In other words, it operates on the chromaticity component of the color vector by filtering out color vectors with large chromaticity errors. The importance of such an image processing strategy was documented in [5].

On the other hand, the AVDF defined in (6) is not only a *chromaticity* filter since it uses both the directional filtering information through the angle distances for its weights as well as the magnitude component of each one of the color vectors. This is a feature that differentiates our design from the chromaticity filters with gray-level processing components introduced in [5]. The generalized *chromaticity* filters introduced there select a subset of the color vectors and then apply gray-scale techniques only to the selected group of vectors. However, if important color information was eliminated due to errors in the chromaticity-based decision part, the filters in [5] are unable to compensate using their gray-scale processing step. That is not the case in the new design. The adaptive filter introduced in (6) does not discard any *magnitude* information based on *chromaticity* analysis. All the vectors inside the operational window contribute to the final output. Simply stated, the filter assigns weights to the *magnitude* component of each color vector modifying in this way their contribution to the output. This natural blending of *chromaticity*-based weights with *magnitude-based* input contributions makes the filter appropriate for color image processing.

III. Application to Color Images

A. Properties

It must be emphasized that the proposed filter framework can be applied in any multichannel signal and any multivariate data with a spatial domain. The most common multichannel filtering problem, however, is that of color image filtering, therefore in this section we also outline some basic properties that make the proposed filters appropriate in color image processing. We confine ourselves mainly to the MAVDF due to its amenability for mathematical treatment.

- *Property I. Invariance Under Scaling and Rotation:*
  The criterion used by the MAVDF filter to single out one vector is the sigmoidal transformation of the vector angle criterion. From (4) it can easily be seen that the angle between two vectors does not depend on the scale. Furthermore, any rotation of the coordinate system does not change the angle between two vectors. Thus, if the input to the sigmoidal remain the same, the fuzzy weight selected under scaling or rotation is the same. Therefore, the MADVF filter is invariant under scaling and rotation, but is not invariant under bias since the addition of a constant vector could change the sum of angles among the vectors.
- *Property II. Preservation of Step Edges:*
  It is easy to establish that a step edge is a root signal of the MAVDF. An illustrative example is depicted in Fig. 1. A window with size 5 is centered around the vector $\mathbf{x}_4$. Calculating the absolute angle distances among the vectors it can be seen from the diagram that $a_4 = a_5 = a_6 < a_2 = a_3$, since the vectors $\mathbf{x}_4$, $\mathbf{x}_5$, and $\mathbf{x}_6$ are parallel. Using the transformation in (8), the higher weights will be the ones associated with these vectors. It is obvious that the MAVDF will discard the vectors $\mathbf{x}_3$ and $\mathbf{x}_2$, resulting in the preservation of the vector edge.

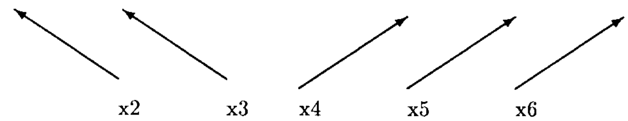


Fig. 1. MAVDF: Edge preservation.

For the adaptive filter in (6) we can not use the same methodology to justify the edge preservation property. Thus, a simple example is introduced to illustrate the effectiveness of the proposed algorithms in filtering operation near noisy edges. In the experiment we use a step edge of height 2, for a two channel invariant signal corrupted by additive mixed Gaussian noise. The signal description is:

$$\mathbf{y}(t) = \mathbf{x} + \mathbf{w}(t) \tag{10}$$

with

$$\begin{aligned} \mathbf{x} &= \begin{pmatrix} 1.5 \\ 1.75 \end{pmatrix} \qquad t \leq 45 \\ \mathbf{x} &= \begin{pmatrix} 3.5 \\ 3.75 \end{pmatrix} \qquad t > 45 \end{aligned} \tag{11}$$

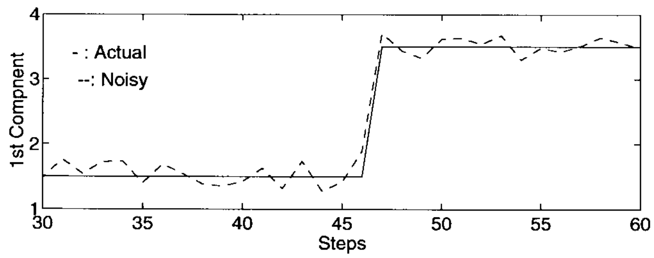
and

$$\mathbf{w}(t) = \mathbf{u}(t)\mathbf{v}_1(t) + [\mathbf{I} - \mathbf{u}(t)]\mathbf{v}_2(t) \tag{12}$$

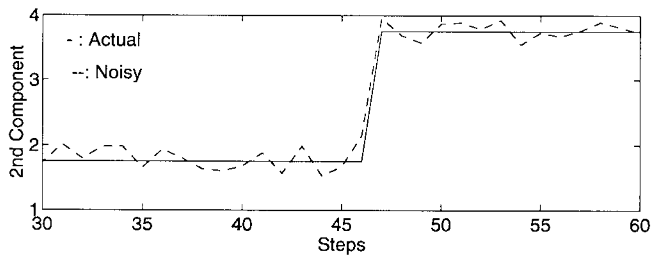
where $\mathbf{u}(t) = u(t)\mathbf{I}_{2\times 1}$ and $u(t)$ is a random number uniformly distributed over the interval [0, 1]. The $\mathbf{v}_1(t)$ results from a Gaussian distribution with zero mean and covariance $0.05\mathbf{I}_{2\times 2}$. The $\mathbf{v}_2(t)$ is Gaussian with zero mean and covariance $0.25\mathbf{I}_{2\times 2}$.

A operational window of size $n = 5$ was used in the experiment. Results are shown in Fig. 2, where the curves denote: 1) the actual signal; 2) the noisy input, and in Fig. 3, where 3) the output of the filter devised from (2) with parameters $\beta = 2$, $r = 1$, and $\alpha = a$; 4) the output of the Median filter; and 5) the output of the (arithmetic) Mean filter are depicted. From the above experiment the following conclusions can be drawn.

1) The Median algorithm works better near the sharp edges.
2) The Mean filter works better than the Median for the homogeneous signal.
3) The proposed AVDF can suppress the noise in homogeneous regions much better than the vector Median filter and can preserve edges better than the Mean filter.

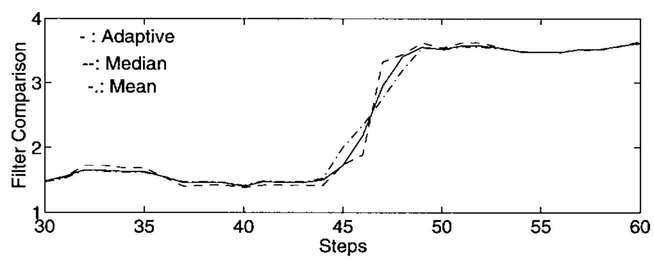


(a)

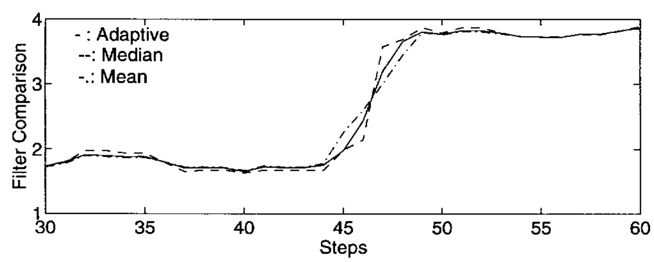


(b)

Fig. 2. Simulation experiment. (a0 Actual signal and (b) noisy input.



(a)



(b)

Fig. 3. Simulation experiment.—filtering results. (a) First component. (b) Second component.

TABLE II
ADAPTIVE VDF'S

AVDF1	Eq.(5) with $r=1$, $\alpha=a$ and $\beta=2$
AVDF2	Eq.(5) with $r=2$, $\alpha=a$ and $\beta=2$
AVDF3	Eq.(5) with $r=1$, $\alpha=\tilde{a}$ and $\beta=2$
AVDF4	Eq.(5) with $r=1$, $\alpha=a2$ and $\beta=2$
AVDF5	Eq.(5) with $r=1$, $\alpha=\tilde{a2}$ and $\beta=2$
AVDF6	Eq.(5) with $r=2$, $\alpha=\tilde{a2}$ and $\beta=2$

B. Simulation Results

The performance of our adaptive designs (see Table II) is compared against that of popular vector processing filters, such as the widely used vector median filter (VMF), the *chromaticity*-based generalized VDF (GVDF) [5] and the arithmetic mean vector filter (AMVF).

TABLE III
NOISE DISTRIBUTIONS

Number	Noise Model
1	Gaussian ($\sigma=30$)
2	impulsive (4%)
3	Gaussian ($\sigma=15$) impulsive (2%)
4	Gaussian ($\sigma=30$) impulsive (4%)

Fig. 4. "Lenna" corrupted with (4%) impulsive noise.

The test image selected for the comparison is the RGB color image "Lenna." The test image has been contaminated using various noise source models in order to assess the performance of the filters under different noise distributions (see Table III). The normalized mean square error (NMSE) has been used as quantitative measure for evaluation purposes. It is computed as

$$\mathrm{NMSE} = \frac{\sum_{i=0}^{N1}\sum_{j=0}^{N2}\|\mathbf{y}(i,j)-\hat{\mathbf{y}}(i,j)\|^2}{\sum_{i=0}^{N1}\sum_{j=0}^{N2}\|\mathbf{y}(i,j)\|^2} \tag{13}$$

where $N1$, $N2$ are the image dimensions, and $\mathbf{y}(i, j)$ and $\hat{\mathbf{y}}(i, j)$ denote the original image vector and the estimation at pixel (i, j), respectively. Table IV summarizes the results obtained for the test image "Lenna" for a 3 × 3 filter window. The results obtained using a 5 × 5 filter window are given in Table V. A "$*$" in a table entry indicates the best filter performance in the corresponding row.

In addition to the quantitative evaluation presented above, a qualitative evaluation is necessary since the visual assessment of the processed images is, ultimately, the best subjective measure of the efficiency of any method [5]. Therefore, we present sample processing results in Figs. 4–9. Fig. 4 shows the color "Lenna" image corrupted with (4%) impulsive noise. Figs. 5–7 show results of the AVDF1, AVDF2, and GVDF, respectively. Similarly, Fig. 8 shows the color "Lenna" image corrupted with Gaussian noise $(\sigma = 15)$ mixed with (2%) impulsive noise. Figs. 9–11 present again the processing of the same filters and with the same order.

C. Conclusions

From the results listed above, it can be easily seen that our filters provide consistently good results in every type of noise, outperforming the other three multichannel filters under consideration. The

TABLE IV
NMSE $(X10^{-2})$ FOR THE "LENNA" IMAGE, WINDOW 3×3

Noise	$AVDF1$	$AVDF2$	$AVDF3$	$AVDF4$	$AVDF5$	$AVDF6$	$GVDF$	VMF	$AMVF$
1	0.7335	0.8935	0.6648*	0.68880	0.6652	0.678	1.46	1.60	0.6963
2	0.2481	0.1881*	0.4920	0.3810	0.7158	0.4156	0.30	0.19	0.8186
3	0.401	0.3601*	0.5528	0.3978	0.5622	0.4181	0.6238	0.5404	0.6160
4	1.039	1.4063	1.2003	1.0316	1.2035	1.0221*	1.982	1.6791	1.298

TABLE V
NMSE $(X10^{-2})$ FOR THE "LENNA" IMAGE, WINDOW 5×5

Noise	$AVDF1$	$AVDF2$	$AVDF3$	$AVDF4$	$AVDF5$	$AVDF6$	$GVDF$	VMF	$AMVF$
1	0.7549	1.2886	0.5770*	0.6483	0.5784	0.6481	1.08	1.17	0.5994
2	0.3087	0.3472	0.5894	0.3754	0.6043	0.3802	0.3018*	0.58	0.6656
3	0.4076*	0.4841	0.5370	0.4172	0.5377	0.3660	0.459	0.5172	0.5702
4	0.9550	1.8115	0.8178*	0.8681	0.8245	0.8481	1.1044	1.0377	0.8896

Fig. 5. ADVF1 of (4) using 3×3 window.

Fig. 6. AVDF2 of (4) using 3×3 window.

different fuzzy filters attenuate both impulsive and Gaussian noise with or without outliers present in the test image. The effect of the distance measure and the smoothing parameter selected are evident in the results summarized above. Regarding the effect of the parameters the following can be concluded.

Fig. 7. GVDF of (4) using 3×3 window.

Fig. 8. "Lenna" corrupted with Gaussian noise $(\sigma = 15)$ mixed with (2%) impulsive noise.

- For small window size, namely, 3×3 window, the filters based on the average distances have worse performance than the filters based on the overall distances for all noise scenarios except for the case of Gaussian noise.

Fig. 9. ADVF1 of (8) using 3 × 3 window.

Fig. 10. AVDF2 of (8) using 3 × 3 window.

- Filters based on average distances cannot smooth out impulsive noise. Despite the fact that they have relatively good performance for a larger window size (5 × 5), they are still less effective in removing impulsive noise than the filters based on overall distances.
- By increasing the value of the smoothing parameter r better results can be obtained for impulsive-type noise. This is expected since increased r values enhances the nonlinear nature of the filter. However, the ability of the filter to smooth nonimpulsive-type noise is restricted, especially for larger windows.

As a general conclusion, the versatile design of (1) allows for a number of different fuzzy filters, which can provide solutions to many types of different filtering problems. Simple adaptive fuzzy designs, such as the AVDF1 can preserve edges and smooth noise under different scenarios, outperforming other widely used multichannel filters. If knowledge about the noise characteristics is available, the designer can tune the parameters of the adaptive filter to obtain better results.

Finally, considering the number of computations, the computationally intensive part of the adaptive algorithm is the distance calculation part. However, this step is common in all multichannel algorithms considered here. In conclusion, our design is simple, does not increase

Fig. 11. GVDF of (8) using 3 × 3 window.

the numerical complexity of the multichannel algorithm and delivers excellent results for complicated multichannel signals, such as real color images.

IV. Concluding Remarks

A new class of AVDF's for multichannel image processing has been introduced in this brief. These filters combine, in a novel way, fuzzy memberships, average filters and angle-based distances. Depending on the criterion which the designer uses to select the membership, a number of different filters can be obtained. Experimental simulation results have demonstrated the efficiency of the proposed filters. The new filters outperform other nonlinear filters, such as the VMF and the GVDF. Moreover, the new filters preserve the chromaticity component, which is very important in visual perception of color images.

References

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

[2] ——, "Order statistics in digital image processing," *Proc. IEEE*, vol. 80, pp. 1893–1923, Dec. 1992.

[3] A. N. Venetsanopoulos and K. N. Plataniotis, "Multichannel image processing," in *Proc. IEEE Workshop on Nonlinear Signal/Image Processing*, 1995, pp. 2–6.

[4] J. Astola, P. Haavisto, and Y. Neuvo, "Vector median filter," *Proc. IEEE*, vol. 78, pp. 678–689, Apr. 1990.

[5] P. E. Trahanias and A. N. Venetsanopoulos, "Vector directional filters: A new class of multichannel image processing filters," *IEEE Trans. Image Processing*, vol. 2, pp. 528–534, Oct. 1993.

[6] P. E. Trahanias, I. Pitas, and A. N. Venetsanopoulos, "Color image processing," in *Advances in 2D and 3D Digital Processing (Techniques and Applications)*, C. T. Leondes, Ed. New York: Academic, 1994, p. 41.

[7] S. G. Tzafestas and A. N. Venetsanopoulos, Eds., *Fuzzy Reasoning in Information, Decision and Control Systems.* Norwell, MA: Kluwer Academic, 1994.

[8] D. R. K Brownrig, "The weighted median filter," *Commun. ACM* vol. 27, pp. 807–818, Aug. 1984.

[9] K. N. Plataniotis, D. Androutsos, and A. N. Venetsanopoulos, "Color image processing using fuzzy vector directional filters," in *Proc. IEEE Workshop on Nonlinear Signal/Image Processing*, 1995, pp. 535–538.

[10] H. J. Zimmermann, *Fuzzy Sets, Decision Making and Expert System.* Boston, MA: Kluwer Academic, 1987.

[11] R. Helson, *Adaptation-Level Theory.* New York: Harper & Row, 1964.

[12] J. M. Mendel, "Fuzzy logic systems for engineering: A tutorial," *Proc. IEEE*, vol. 83, pp. 345–377, Mar. 1995.

[13] T. Jiang and Y. Li, "Generalized defuzzification strategies and their parameter learning procedures," *IEEE Trans. Fuzzy Syst.*, vol. 4, pp. 64–71, 1996.