Adaptive Fuzzy Systems for Multichannel Signal Processing

KONSTANTINOS N. PLATANIOTIS, MEMBER, IEEE,
DIMITRIOS ANDROUTSOS, STUDENT MEMBER, IEEE, AND
ANASTASIOS N. VENETSANOPOULOS, FELLOW, IEEE

Invited Paper

Processing multichannel signals using digital signal processing techniques has received increased attention lately due to its importance in applications such as multimedia technologies and telecommunications. The objective of this paper is twofold: 1) to introduce adaptive filtering techniques to the reader who is just beginning in this area and 2) to provide a review for the reader who may be well versed in signal processing. The perspective of the topic offered here is one that comes primarily from work done in the field of multichannel (color) image processing. Hence, many of the techniques and works cited here relate to image processing with the emphasis placed primarily on filtering algorithms based on fuzzy concepts, multidimensional scaling, and order statistics-based designs. It should be noted, however, that multichannel signal processing is a very broad field and thus contains many other approaches that have been developed from different perspectives, such as transform domain filtering, classical least-square approaches, neural networks, and stochastic methods, just to name a few.

In this paper, we present a general formulation based on fuzzy concepts, which allows the use of adaptive weights in the filtering structure, and we discuss different filter designs. The strong potential of fuzzy adaptive filters for multichannel signal applications, such as color image processing, is illustrated with several examples.

Keywords—Fuzzy systems, image processing, multichannel signal processing, neural networks.

I. INTRODUCTION

The availability of a wide set of multichannel information sources in application areas, such as color image processing, multispectral remote sensing imagery, biomedicine, robotics, and industrial inspection, has stimulated a renewed interest in developing efficient and cost-effective processing techniques for multichannel signals. In recent years, significant advances have been made in the development of multichannel signal processing techniques. Such techniques are used in a variety of tasks, such as color image filtering and processing video sequences, enhancement of multispectral remote sensing data, seismic deconvolution for oil exploration, and boundary detection in vector fields.

The most common signal processing task is noise filtering. Filtering is the process of estimating a signal degraded, in most cases, by additive random noise. This task is an essential part of any signal processing system, especially when the final product is used for human interpretation, such as visual inspection or for automatic analysis [1], [2]. Several filtering techniques have been proposed over the years. Among them are linear processing techniques, whose mathematical simplicity and the existence of a unifying theory make their design and implementation easy [3]. Their simplicity, in addition to their satisfactory performance in a variety of practical applications, has made them methods of choice for many years. However, most of these techniques operate assuming a Gaussian model for the statistical characteristics of the underlying process, and thus they try to optimize the parameters of a system suitable for such a model.

Many signal processing problems cannot be efficiently solved by using linear techniques. For example, an area where linear processing techniques fail is in image processing, where conventional linear techniques cannot cope with the nonlinearities of the image formation model and do not take into account the nonlinear nature of the human visual system. Image signals are composed of flat regional parts and abruptly changing areas, such as edges, which carry important information for visual perception. Filters having good edge and image detail preservation properties are highly suitable for image filtering and enhancement. Unfortunately, most of the linear signal processing tech-

niques tend to blur edges and to degrade lines, edges, and other fine image details.

Recently, fuzzy techniques have been investigated to provide a bridge between linear and nonlinear techniques. This paper summarizes efforts devoted to the development of fuzzy systems suitable for filtering multichannel signals. The approach discussed here is an adaptive one. It integrates well-known theories in the areas of nonlinear filtering, multidimensional scaling, robust statistics, and fuzzy sets so as to form a new composite model. At the heart of the approach is a filter whose weights (parameters) are adaptively determined on the basis of local signal context. The filtering procedure is seen as the estimation of the prototype for a given set of input signals in a processing window. Thus, filtering is the process of replacing a noisy signal by a prototypical value, such that some metric which is a function of the filter output and its neighbors is minimized.

The organization of the paper is as follows. In Section II, we provide an overview of the literature and research in the area. In Section III, we present the problem under consideration and introduce the general framework for an adaptive fuzzy solution. We also discuss properties and design characteristics of the proposed fuzzy system. In Section IV, we discuss distance and similarity measures, which can be used to define dissimilarity or similarity between multichannel, vector-like signals. In Section V, we review membership functions based on the distance and similarity measures surveyed in the paper. Properties, characteristics, and implementation issues are also discussed in detail. A number of filtering systems that are supported by the general framework discussed in this work are also presented in this section. Section VI deals with the problem of color image processing, an important area of multichannel signal processing, where we present experimental results and comparisons with other standard methods. Finally, Section VII summarizes the conclusions and describes future research.

II. LITERATURE REVIEW

In applications where the noise characteristics are unknown or vary with time, the use of conventional linear filters to smooth out the noise does not provide satisfactory solutions. In such cases adaptive filters become very important. Most adaptive filters have the structure of finite impulse response filters with time-varying coefficients [4]. Due to their flexibility and self-adjusting nature, they have found uses in many practical applications ranging from telephone echo canceling to radar signal processing to biomedical signal enhancement [5].

However, the need to treat increasingly complex nonlinear systems coupled with the availability of increasing computing power has led to a reevaluation of conventional filtering methodologies. New algorithms and techniques, which can take advantage of the increase in computing power and can handle more realistic assumptions, are needed. To this end, nonlinear signal processing techniques are introduced. Nonlinear techniques, theoretically, are able to suppress non-Gaussian and signal-dependent noise, to preserve important signal elements, such as edges and fine details, and eliminate degradations occurring during signal formation or transmission through nonlinear channels. Despite impressive growth in the past two decades, new theoretical results, new tools, and emerging applications, nonlinear filtering techniques still lack a unifying theory that encompasses existing nonlinear processing techniques. Instead, each class of nonlinear operators possesses its own mathematical tools which can provide a reasonably good analysis of its performance. Consequently, a multitude of nonlinear signal processing techniques have appeared in the literature.

At present the following classes of nonlinear processing techniques can be identified:

1) polynomial-based techniques [6], [7];
2) homomorphic techniques [1], [8];
3) order statistic-based techniques [1], [9], [10];
4) techniques based on mathematical morphology [11]–[14];
5) neural network-based techniques [15]–[18].

Polynomial filters, especially second-order Volterra filters (quadratic filters), represent the most natural extension of linear filters. Volterra filters can be seen as linear filters with higher order polynomial extensions. They have been used for nonlinear channel modeling in telecommunications as well as in multichannel geophysical signal processing and color image processing.

Homomorphic filters and their extensions represent one of the first classes of nonlinear filters and have been used extensively in digital signal and image processing. This filter class has found various practical applications, such as multiplicative and signal-dependent noise removal, multichannel satellite image processing, and identification of fingerprints. Homomorphic filters use nonlinearities (mainly the logarithm) to transform nonlinearly related signals into additive signals and then process them using linear filters. The output of the linear filter is then transformed by the inverse nonlinearity.

One of the most popular families of nonlinear filters for noise removal are order-statistics filters. The theoretical basis of these filters is that of robust statistics [1]. These filters utilize algebraic ordering of a windowed set to compute the output signal. There exist several filters which are members of this class, such as the vector median filter (VMF) [9]. Order-statistics multichannel filters have found extensive applications in multichannel signal processing and color image filtering.

Morphological filters utilize geometric, rather than analytical, features of signals. These filters have found applications in image processing and analysis. Specifically, areas of application include biomedical signal processing, nonlinear filtering, edge detection, and image enhancement.

Neural network-based techniques have been extensively used over the past ten years for multichannel signal pro-

cessing. They have been successfully applied in image processing and analysis, brain research, signal classification, speech recognition, and of course noise reduction. Their attractive generalization properties, their ability to perform complex mappings from a set of noise signals to the noise-free signal, and their parallel implementation make them the method of choice in many digital signal processing applications.

Adaptive nonlinear schemes have been considered as successful alternatives to adaptive linear techniques, especially in the case of nonstationary signals corrupted by non-Gaussian noise [2], [19], [20]. Adaptive versions of order-statistics filters, Volterra filters, and neural network structures have been used in a variety of signal processing applications ranging from image restoration to biomedical signal enhancement. As for linear adaptive filters, their performance is based on the reliable estimation of signal and noise characteristics. Local statistics estimation is employed by decision-directed nonlinear filters, a class of adaptive filters, which has been extensively used in image processing applications [21].

Recently, a number of adaptive techniques based on fuzzy logic principles have also been proposed. Such techniques have been used mainly for high-level analysis of signals and images, computer vision applications, systems control, pattern recognition, and decision modeling. Different approaches ranging from fuzzy clustering to fuzzy entropy and decision making under fuzzy constraints have been used for scene detection, object recognition, and decision-directed image analysis. More recently, fuzzy techniques have been used for low-level signal and image processing tasks, such as non-Gaussian noise elimination, nonlinear/non-Gaussian stochastic estimation, image enhancement, video coding, signal sharpening, and edge detection [22]–[37].

A number of fuzzy techniques adopt a window-based, rule-driven approach leading to data-dependent fuzzy filters, which are constructed by fuzzy rules in order to remove additive noise while preserving important signal characteristics, such as edges. Using a bank of IF-THEN-ELSE fuzzy rules, the fuzzy filter directly yields the filtered output taking into account selected patterns in the neighborhood of the element to be processed. Since the antecedents of fuzzy rules can be composed of several local characteristics, it is possible for the fuzzy filter to adapt to local data. Local correlation in the data is utilized by applying the fuzzy rules directly on the signal elements which lie within the operational window. Thus the output of the fuzzy filter depends on the fuzzy rule and the defuzzification process, which combines the effects of the different rules into an output value.

Through the utilization of linguistic terms, a fuzzy rule-based approach to signal processing allows for the incorporation of human knowledge and intuition into the design, which cannot be achieved via traditional mathematical modeling techniques. However, there is no optimal way to determine the number and type of fuzzy rules required for the fuzzy image operation. Usually, a large number of rules are necessary and the designer has to compromise between

quality and number of rules, since even for a moderate processing window a large number of rules are required [24], [25], [28].

Data-dependent filters adopting fuzzy reasoning have been proposed to overcome these difficulties. These designs combine fuzzy concepts, such as membership functions, fuzzy rules, and fuzzy aggregators with nonlinear filters, such as the α-trimmed mean filter and the weighted average mean filter, in order to remove Gaussian and non-Gaussian noise while preserving useful signal characteristics, such as edges in image signals.

Some methods utilize fuzzy rule-based systems to extend the classical structure of a weighted linear filter. The fuzzy weights are evaluated by fuzzy rules whose inputs are local features that extract information from the vicinity of the signal value to be processed. Hybrid fuzzy operators that combine nonlinear filters devoted to the suppression of impulsive spikes with linear filters have also been proposed [37]. Other approaches utilize fuzzy rules to combine the outputs of different filters. For example, in [27] the fuzzy system combines the outputs of five classical nonlinear filters depending on values of local features. A similar approach is also discussed in [38]. In addition, based on the adoption of a fuzzy positive Boolean function, a new class of operators named fuzzy stack filters have been proposed [39]. These operators extend the smoothing capabilities of the classical stack filters and can provide efficient and cost-effective solutions provided that an adequate set of training signals is available. Recently, neurofuzzy filters and evolutionary optimization techniques have been combined in the hopes of deriving a nonlinear filter which can cancel noise and preserve signal details at the same time [29].

As is the case of nonlinear techniques in general, the fuzzy signal processing techniques available today lack a unifying theory. Cross fertilization among the different fuzzy techniques, as well as with other nonlinear techniques, has appeared promising. For example, mathematical morphology and fuzzy concepts have been blended together in the case of fuzzy stack operators. Also, fuzzy designs and order-statistic filters have been efficiently integrated into one class even though they come from completely different origins [34], [35].

The plethora of available fuzzy techniques poses some application difficulties. Since most of them are designed to perform well in a specific application, their performance deteriorates rapidly under different operation scenarios. Thus, an adaptive fuzzy system, which performs equally well in a wide variety of applications, is of great importance.

III. THE GENERAL FRAMEWORK

A. The Problem Under Consideration

Consider the following commonly used model for a multichannel signal corrupted by additive noise:

$$\boldsymbol{x} = \boldsymbol{y} + \boldsymbol{n}_o \qquad (1)$$

where $\boldsymbol{y}$ denotes the m channel uncorrupted signal vector, $\boldsymbol{x}$ is the corresponding noisy vector to be filtered, and $\boldsymbol{n}_o$ an additive noise vector.

The signal processing literature has been dominated by the assumption of the Gaussian model for the statistical characteristics of the noise. Optimal filtering solutions for the problem at hand can be devised based on this assumption. However, the type and degree of noise corruption depends on many factors, such as the measurement apparatus, the signal levels, and the experimental procedure under consideration. The appearance of the noise and its effect on the signal are related to its frequency characteristics. In many cases, the noise characteristics vary within the same application. Such cases include channel noise in image transmission as well as atmospheric noise corrupting multichannel satellite images.

Although the Gaussian model is often justified in practice by the central limit theorem, we often encounter noise processes that exhibit impulsive behavior and are more accurately modeled by heavy-tailed, non-Gaussian distributions [40]–[42]. Impulsive behavior can be characterized in terms of short duration, high-energy spikes attaining large amplitudes with probability higher than that predicted by a Gaussian density model [43]. Even if the level of non-Gaussian noise contamination is small, the performance of a filtering system optimized under the Gaussian assumption can suffer from drastic degradation. In such a case, the performance of classical adaptive filtering schemes, such as recursive least squares (RLS) or simple weighted average filters, is seriously degraded [44]. Thus, there is a need for a flexible and efficient filter class for non-Gaussian environments that can appear in practice [45].

B. The Adaptive Fuzzy System

Let us define $\boldsymbol{x}(l)$ as the multichannel sample to be processed at time index l and let W be a set of ($n = 2N + 1$) neighboring samples which belong to the window W centered on $\boldsymbol{x}_l$

$$W = [\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n]$$
$$= [\boldsymbol{x}_{l-N}, \ldots, \boldsymbol{x}_{l-1}, \boldsymbol{x}_l, \boldsymbol{x}_{l+1}, \ldots, \boldsymbol{x}_{l+N}]. \quad (2)$$

Since the most commonly used method to decrease the level of random noise present in the signal is smoothing, an averaging operation is required in order to replace the noisy vector at the window center with a suitable representative vector (prototypical value). The general form of the system presented 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 [34], [46]–[50]:

$$\hat{\boldsymbol{y}} = \sum_{j=1}^{n} w_j \boldsymbol{x}_j = \sum_{j=1}^{n} \frac{f(\mu_j)}{\sum_{j=1}^{n} f(\mu_j)} \boldsymbol{x}_j \quad (3)$$

where $f(\mu_j) = \mu_j^{\lambda}$ is a function adaptively determined on the basis of local context with μ_j the membership function of input $\boldsymbol{x}_j$ and λ a parameter such that $\lambda \in [0, \infty)$.

In this adaptive design the weights provide the degree to which an input vector contributes to the output of the filter. The relationship between the signal at the window center (vector under consideration) and each signal within the window should be reflected in the decision for the weights of the filter. Through the normalization procedure, two constraints necessary to ensure that the output is an unbiased estimator are satisfied, namely:

1) each weight is a positive number, $w_j \geq 0$;
2) the summation of all the weights is equal to unity $\Sigma_{j=1}^{n} w_j = 1$.

The weights of the filter are determined adaptively using functions of a distance criterion between the input vectors. These weighting coefficients are based on the distance between the center of the window (the vector under consideration) and all other vector samples inside the filter window. The weights can be considered to be a membership function with respect to the specific window component. The adaptive algorithm evaluates a membership function based on a given vector signal and then uses the membership values to calculate the final filtered output.

To explain the concepts behind the filter only the definition of the fuzzy set is required. Other definitions, such as fuzzy rule bases and fuzzy control, are not essential to this work and thus are omitted. Assume that $\mathbf{X}$ is a universe of discourse with elements $\boldsymbol{x}$. Then, a fuzzy set $\boldsymbol{A}$ in $\boldsymbol{x}$ is a set of ordered pairs $\boldsymbol{A} = [\boldsymbol{x}/\mu_A(\boldsymbol{x})|\boldsymbol{x} \in \boldsymbol{X}]$, where $\mu_A(\boldsymbol{x})$ is the membership function or grade of membership of $\boldsymbol{x}$ in $\boldsymbol{A}$ which maps $\boldsymbol{X}$ into a membership space $\boldsymbol{M}$. The range of the membership function is a subset of the nonnegative real numbers whose supremum is finite. In practical applications $\boldsymbol{M}$ is normalized to the interval $[0, 1]$.

The design summarized here qualifies as an adaptive fuzzy system since it utilizes sample input data and inference procedures (here in the form of transformed distance metrics) to define a fuzzy system at each time instant. Through the adaptation mechanism utilized, the system structure changes over time, resulting in a time-varying mapping between input values and filtered output. This temporal mapping defines an adaptive fuzzy system. As was argued in [51], adaptation, or learning, is essentially parameter changing. Thus, by changing the weights in (3), we have developed an adaptive fuzzy system capable of learning new associations between input patterns and new functional dependencies. In the framework described here, this can be accomplished without the use of linguistic fuzzy rules or local statistics estimation. Features extracted from local data, such as distances among neighboring input vectors, are used to define the fuzzy weights.

The noise smoothing problem is seen as a problem of prototype estimation given a set of signal inputs. In this sense, filtering is the process of replacing the noise-corrupted multichannel signal at the window center by a prototype signal, such that the differences between this prototype and all its neighbors inside the window are minimized in some sense. This operation is, essentially,

a defuzzification procedure. It determines the most appropriate signal value (a vector signal in the case of multichannel inputs), to represent a collection of elements whose membership functions have been constructed over a universe of discourse.

Although a number of different defuzzification strategies exist, the centroid defuzzification approach, known as the center of gravity (CoG), is often utilized in practice. The CoG method generates a defuzzified value which is at the center of the values of a fuzzy set. Its defuzzified output actually corresponds to the membership-graded weighted mean of the square (Euclidean) distance.

To clarify this, let us consider a fuzzy set A that is defuzzified as:

$$A = \left(\frac{\mu_1}{x_1}, \frac{\mu_2}{x_2}, \ldots, \frac{\mu_n}{x_n} \right) \qquad (4)$$

where μ_n is the membership function associated with the input value x_n. If a quadratic cost function is considered

$$K(\overline{x}) = \sum_{i=1}^{n} |x_i - \overline{x}|^T \mu_i |x_i - \overline{x}| \qquad (5)$$

the CoG defuzzified value is obtained when $K(\overline{x})$ is minimized by differentiation

$$\overline{y} = \frac{\sum_{i=1}^{n} x_i \mu_i}{\sum_{i=1}^{n} \mu_i}. \qquad (6)$$

Simple inspection of the CoG defuzzified value obtained reveals the similarity with the adaptive filtered output of (3). We can therefore claim that the output of our adaptive filter can be considered as the output of the CoG defuzzification strategy with the noisy multichannel signals as members of a fuzzy set and the membership functions $\mu_i, i-1, 2, \ldots, n$ defined over them.

In such a design, the overall performance of the processing system is determined by the defuzzification procedure selected. The quadratic cost function discussed above can be generalized to include any arbitrary function of μ. Under such a scenario, we assume that the cost function associated with the selection of the defuzzified value to represent the fuzzy set A is

$$K(\overline{x}) = \sum_{i=1}^{n} |x_i - \overline{x}|^T f(\mu_i) |x_i - \overline{x}| \qquad (7)$$

where $f(\mu_i)$ is a function of the associated membership function. By minimizing the above quadratic form, a defuzzified (crisp) value can be obtained as

$$\overline{y} = \frac{\sum_{i=1}^{n} x_i f(\mu_i)}{\sum_{i=1}^{n} f(\mu_i)} \qquad (8)$$

which is identical to the form used to generate the filtered output in the adaptive design of (3).

If the power function $f(\mu_i) = \mu_i^\lambda$ with $\lambda \in [0, \infty)$ is used, the defuzzified value can be obtained through the following equation:

$$\overline{y} = \frac{\sum_{i=1}^{n} x_i \mu_i^\lambda}{\sum_{i=1}^{n} \mu_i^\lambda}. \qquad (9)$$

It can easily be seen that, in the generalized defuzzification rule of (9), if $\lambda = 1$ the widely used CoG strategy can be obtained.

The defuzzified vector-valued signal obtained through the CoG strategy is a vector-valued signal, which was not part of the original set of input vectors. However, there are signal processing applications, such as image filtering, where it is desirable for the filter output to be one of the samples in the input window. As an example, the VMF [9] is always constrained, by definition, to be one of the input samples. Thus, if the output of the adaptive fuzzy system is required to be a member of the original input set, a different defuzzification strategy should be used. By defining $\mu_{(\max)}$ to be the largest membership value, the adaptive weights in (3) can be rewritten as follows [90]–[92]:

$$w_j = \frac{\mu_j^\lambda}{\sum_{j=1}^{n} \mu_j^\lambda} = \frac{\left(\frac{\mu_j}{\mu_{(\max)}} \right)^\lambda}{\sum_{j=1}^{n} \left(\frac{\mu_j}{\mu_{(\max)}} \right)^\lambda}. \qquad (10)$$

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

$$w_j = \begin{cases} 1, & \text{if } \mu_j = \mu_{(\max)} \\ 0, & \text{if } \mu_j \neq \mu_{(\max)}. \end{cases} \qquad (11)$$

Equation (11) represents the maximum defuzzifier strategy. If 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 an adaptive fuzzy system is defined as

$$\hat{y} = x_j, \quad \mu_j = \mu_{(\max)}. \qquad (12)$$

In this case, the fuzzy adaptive filter behaves as a mode-like selection filter [45], since by construction its output is always one of the samples inside the processing window. This selection property is shared by well-known nonlinear filters, such as the VMF and the myriad filter [40]. However, unlike these filters, which are optimized for specific noise models (the Laplacian and Cauchy model, respectively), the fuzzy filter can be optimized for any noise model by tuning its membership function. Thus existing selection filters can be generalized.

C. Determining the Parameters

The most crucial step in the design of the adaptive fuzzy system lies in determining the membership function to be used in the construction of its weights. The difficulties associated with the meaning and measurement of the membership function hinder the applicability of fuzzy techniques to many practical applications. It is important to clarify where the membership function arises, how is it used and measured, and how it can be manipulated in order to provide meaningful results. Since there are different interpretations of fuzziness, the meaning of the membership function changes depending on the application or methodology adopted. In general, apart from the formal definition, a membership function can be seen as a "graded membership" in a set. Depending on the interpretation of fuzziness, various solutions to the problem of membership definition and graded membership can be obtained. Our analysis considers the membership function as a function of similarity. Viewing membership values as similarity indicators is often used in prototype theory where membership is a notion of being similar to a representative of a category [52]. Thus a membership function value can be used to quantify the degree of similarity of an element to the set in question. The assumption behind this approach is that there exists a perfect (ideal) example of the set which belongs to the set to the full degree. The valuation of membership for the rest of the elements in the set can be regarded as the comparison of a given input x_i with the ideal input x_r, which results in a distance $d(x_i, x_r)$.

If we wish to adopt a view of similarity, we must scale the membership function accordingly. If the input under consideration has all the features of the ideal prototype then the distance should be zero, and this object should belong to the set to the full degree. On the other hand, if no similarity between the ideal prototype and the input exists, the distance should be infinite and this notion should be reflected in the membership value.

Assuming that a certain degree of membership is assigned to each element in the set, this membership function can be defined as

$$\mu_i = \frac{1}{1 + f(d(x_i, x_r))}. \tag{13}$$

Based on the definition above, $\mu_i \to 0$ as $d(x_i, x_r) \to \infty$ and $\mu_i = 1$ when $d(x_i, x_r) = 0$. Equation (13) is only a transformation rule from one numerical representation into another. To complete the process, the exact form of the distance function has to be specified. Depending on the specific distance measure that is applied to the input data, a different fuzzy membership function can be devised.

However, the definition of a distance (or similarity) measure requires an appropriate metric space on which the different distance (similarity) measures will be defined and evaluated. Although the notion of distance is very natural in the case of scalar signals [one-dimensional (1-D) signals], it cannot be extended in a straightforward way for the case of vector signals. In a vector space, different measures can be used to quantify similarity or dissimilarity

among multichannel inputs. Therefore, before we discuss the exact form of the membership function and how it will be integrated in our adaptive framework, we need to define and quantify distance, or similarity, among multichannel signals.

IV. Measuring Similarity

Let us assume that two m-D signals x_i and x_j are available. The most commonly used measure to quantify distance between these two vectors is the generalized weighted Minkowski metric, which is defined as follows [53]:

$$d_p(i, j) = c \left(\sum_{k=1}^{m} \xi_k | (x_i^k - x_j^k) | \right)^p \right)^{1/p} \tag{14}$$

where m is the dimension of the vector x_i and x_i^k is the kth element of x_i. The nonnegative scaling parameter c is a measure of the overall discrimination power. The exponent p defines the nature of the distance metric. The most popular cases occur when $p = 1$ (city-block distance) and when $p = 2$ (Euclidean distance) [54], [55]. The chess-board distance corresponds to $p = \infty$. In this case, the distance between the two m-D vectors is considered equal to the maximum distance among their components. The parameter ξ_k measures the proportion of attention allocated to the dimensional component k and thus $\Sigma_k \xi_k = 1$.

Vectors having a range of values greater than a desirable threshold can be scaled down by the use of the weighting function ξ.

An alternative method of implementation for vector signals taking nonnegative values, such as color vectors in the RGB space, is to build the weighting into the distance measure. Such an example is the Canberra metric [56], [57]

$$d_c(i, j) = \sum_{k=1}^{m} \frac{|x_i^k - x_j^k|}{|x_i^k + x_j^k|} \tag{15}$$

where m is the dimension of the vector x_i and x_i^k is the kth element of x_i. The summand is defined to be zero if both x_i^k and x_j^k are zero.

Of course, there are many other measures by which a distance function can be constructed. Depending on the nature of the problem and the constraints imposed by the design, one method may be more appropriate than the other [58]. Furthermore, measures other than distance can be used to measure similarity between vector signals, as the next section will attest. Any nonparametric function $s(x_i, x_j)$ can be used to compare the two multichannel signals x_i and x_j. This can be done by utilizing a symmetric function whose value is large when x_i and x_j are similar. An example of such a function is the normalized inner product defined as

$$s_1(x_i, x_j) = \frac{x_i x_j^T}{|x_i||x_j|} \tag{16}$$

which corresponds to the cosine of the angle between the two vectors x_i and x_j. The cosine of the angle (or the magnitude of the angle) discussed here is used to quantify their

similarity in orientation. Therefore, in applications where the orientation difference between two vector signals is of importance, the normalized inner product or equivalently the angular distance

$$\theta = \cos^{-1}\left(\frac{\boldsymbol{x}_i \boldsymbol{x}_j^T}{|\boldsymbol{x}_i||\boldsymbol{x}_j|}\right) \qquad (17)$$

can be used instead of the L_p metric functions to quantify the dissimilarity between the two vectors [59], [60]. As an example, we consider color images where the color signals appear as three-dimensional (3-D) vectors in the RGB color space. It was argued in [59] that similar colors have almost parallel orientations. On the other hand, significantly different colors point in different overall directions in the 3-D color space. Thus the angular distance which quantifies the orientation difference between two color signals is a meaningful measure of their similarity.

It is obvious that a generalized similarity measure model which can effectively quantify differences among multi-channel signals should take into consideration both the magnitude and the orientation of each vector signal. The distance or similarity measures discussed thus far utilize only part of the information carried by the vector signal. It is anticipated that a generalized measure based on both the magnitude and the orientation of the vectors will provide a robust solution to the problem of similarity between two vectors. To this end, a new similarity measure was introduced [46]. The proposed measure defines the similarity between two vectors $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ as follows:

$$s_2(\boldsymbol{x}_i, \boldsymbol{x}_j) = w_i\left(\frac{\boldsymbol{x}_i \boldsymbol{x}_j^T}{|\boldsymbol{x}_i||\boldsymbol{x}_j|}\right) w_j\left(1 - \frac{||\boldsymbol{x}_i| - |\boldsymbol{x}_j||}{\max(|\boldsymbol{x}_i|, |\boldsymbol{x}_j|)}\right). \qquad (18)$$

As can be seen, this similarity measure takes into consideration both the direction and the magnitude of the vector inputs. The first part of the measure is equivalent to the angular distance defined previously and the second part is related to the normalized difference in magnitude. Thus if the two vectors under consideration have the same length, the second part of (18) becomes unity and only the directional information is used. On the other hand, if the vectors under consideration have the same direction in the vector space (collinear vectors) the first part (orientation) is unity and the similarity measure of (18) is based only on the magnitude difference. In addition, the weights w_i and w_j can be adjusted to stress either component more or less, depending on the design constraints.

This measure can be considered as a member of the generalized "content model" family of measures, which can be used to define the similarity between multidimensional signals.

The main idea behind the content model family is that similarity between two vectors is regarded as the degree of common content in relation to the total content of the two vectors [61]–[65]. Therefore, given the common quantity, commonality C_{ij}, and the total quantity, totality T_{ij}, the similarity between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is defined as

$$s(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{C_{ij}}{T_{ij}}. \qquad (19)$$

Based on the general framework of (19), different similarity measures can be obtained by utilizing different commonality and totality concepts.

Assume that given the two input signals $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, the angle between them is θ and their magnitudes are $|\boldsymbol{x_i}|$ and $|\boldsymbol{x_j}|$, respectively. As before, the magnitudes of the vectors represent the intensity and the angle between the vectors quantifies the orientation difference between them.

Based on these elements, commonality can be defined as the sum of the projections of one vector over the other (e.g., $h_i = |\boldsymbol{x_i}| \cos(\theta)$) and totality as the sum of their magnitudes [61]. Therefore, their similarity model can be written as

$$s_3(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{h_i + h_j}{|\boldsymbol{x_i}| + |\boldsymbol{x_j}|} = \frac{|\boldsymbol{x_i}| \cos(\theta) + |\boldsymbol{x_i}| \cos(\theta)}{|\boldsymbol{x_i}| + |\boldsymbol{x_j}|}$$
$$= \cos(\theta). \qquad (20)$$

Although the content model in [61] is equivalent to the normalized inner product (cosine of the angle) similarity model of (16), different similarity measures can be devised if we define commonality and/or totality between the two vectors differently. Experimental studies have revealed that there is a systematic deviation between empirically measured similarity values and those obtained through the utilization of the model in [61], especially in applications where the magnitudes of the vectors are of importance. To compensate for the discrepancy, the totality T_{ij} was redefined as the vector sum of the two vectors under consideration. In such a case, similarity was defined as

$$s_4(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{h_i + h_j}{(|\boldsymbol{x_i}|^2 + |\boldsymbol{x_j}|^2 + 2|\boldsymbol{x_i}||\boldsymbol{x_j}| \cos(\theta))^{1/2}}. \qquad (21)$$

In the special case of vectors with equal magnitudes, the similarity measure is solely based on the orientation differences between the two vectors, and it can be written as $s_4(\boldsymbol{x}_i, \boldsymbol{x}_j) = (\cos(\theta)/\cos((\theta/2))$.

These are not the only similarity measures which can be devised based on the content-model approach. For example, it is also possible to define commonality between two vectors as a vector algebraic sum of their projections, instead of a simple sum. That gives a mathematically lower value of commonality than the one used in the models reported earlier. Using the two totality measures we can comprise two new similarity measures as

$$s_5(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{(|h_i|^2 + |h_j|^2 + 2|h_i||h_j| \cos(\theta))^{1/2}}{|\boldsymbol{x}_i| + |\boldsymbol{x}_j|} \qquad (22)$$

or

$$s_5(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{\cos(\theta)(|\boldsymbol{x}_i|^2 + |\boldsymbol{x}_j|^2 + 2|\boldsymbol{x}_i||\boldsymbol{x}_j| \cos(\theta))^{1/2}}{|\boldsymbol{x}_i| + |\boldsymbol{x}_j|}. \qquad (23)$$

If only the orientation similarity between the two vectors is of interest, assuming that $|\boldsymbol{x_i}| = |\boldsymbol{x_j}|$, the above similarity measure can be rewritten as $s_5(\boldsymbol{x_i}, \boldsymbol{x_j}) = \cos(\theta) \cos(\theta/2)$.

If, on the other hand, we define the totality T_{ij} as the algebraic sum of the original vectors and define commonality C_{ij} as the algebraic sum of the corresponding projections, the resulting similarity measure can be expressed as

$$s_6(\boldsymbol{x}_i, \boldsymbol{x}_j) = \frac{\cos(\theta) p_{(i,j)}}{p_{(i,j)}} = \cos(\theta) \qquad (24)$$

with $p_{(i,j)} = (|\boldsymbol{x}_i|^2 + |\boldsymbol{x}_j|^2 + 2|\boldsymbol{x}_i||\boldsymbol{x}_j| \cos(\theta))^{1/2}$, which is the same expression obtained through the utilization of the inner product in (16).

Other members of the content-based family of similarity measures can be obtained by modifying either the commonality, the totality, or both. Equation (19) can be seen as a guideline for the construction of specific models where the common part and the total part are specified. As a general observation, we can claim that when the totality and commonality were derived according to the same principle, e.g., sum of vectors, the cosine of the angle between the two vectors can be used to quantify similarity. On the other hand, when commonality and totality were derived according to different principles, similarity was defined as a function of both the angle between the vectors and their corresponding magnitudes.

Content-based measures can also be used to define dissimilarity among vector signals. This is the approach taken in [65], where the emphasis is on what is uncommon to the two vectors instead of on what is common. The difference between the two vectors divided by the total part was assumed to be the measure of their dissimilarity. It was suggested in [65] that the part not in common is specified as the distance between the two vector termini with the totality defined as the vector sum of the two vectors under consideration. Furthermore, assuming that similarity and distance are complimentary, [65] proposed the following similarity measure:

$$s_7(\boldsymbol{x}_i, \boldsymbol{x}_j) = 1 - \frac{\left(|\boldsymbol{x}_i|^2 + |\boldsymbol{x}_j|^2 - 2|\boldsymbol{x}_i||\boldsymbol{x}_j| \cos(\theta)\right)^{1/2}}{\left(|\boldsymbol{x}_i|^2 + |\boldsymbol{x}_j|^2 + 2|\boldsymbol{x}_i||\boldsymbol{x}_j| \cos(\theta)\right)^{1/2}} \qquad (25)$$

where the numerator of the ratio represents the distance between the two vector termini, e.g., vector difference, and the denominator is an indication of the totality. The different nonmetric similarity measures described here, or similarity measures such as those discussed in [66] can be used instead of the Minkowski-type distance measures to quantify distance among a vector under consideration and the ideal prototype in our membership function mechanism, as discussed earlier.

V. THE MEMBERSHIP FUNCTION

Having discussed the different measures to quantify distance (or similarity) between two vector inputs, we turn attention to the problem of membership function specification. The generic form of our function was given in [67]

as

$$\mu_i = \frac{1}{1 + f(d_i)} \qquad (26)$$

where $f(\cdot)$ is a function of the distance between the vector signal $\boldsymbol{x}_i$ and the ideal prototype $\boldsymbol{x}_r$. Membership functions are either monotonically increasing functions from zero to one, monotonically decreasing from one to zero, or can be divided into monotonically increasing or decreasing parts. Each increasing or decreasing part is specified by a crossover or dispersion point. The particular function $f(d_i)$ used in (26) will determine the actual shape of the membership function [67]–[71]. The approach of [67] suggests that since the relationship between distances measured in physical units and perception is generally exponential, an exponential type of function should be used in the generic membership function [67]. The resulting type of a sigmoidal function deduced from this proposition can be defined as

$$\mu_i(\boldsymbol{x}_i) = \frac{1}{1 + \exp\left(\dfrac{-c(s(\boldsymbol{x}_i, \boldsymbol{x}_r) - a_1)}{\sigma_1}\right)} \qquad (27)$$

with $\lim_{s \to \infty} \mu_i = 1$ and $\lim_{s \to 0} \mu_i = 0$ for a monotonically increasing function, where $s(\boldsymbol{x}_i, \boldsymbol{x}_r)$ is any similarity function defined in (18)–(25) and

$$\mu_i(\boldsymbol{x}_i) = \frac{1}{1 + \exp\left(\dfrac{-c(d_p(\boldsymbol{x}_i, \boldsymbol{x}_r) - a_2)}{\sigma_2}\right)} \qquad (28)$$

with $\lim_{d \to \infty} \mu_i = 0$ and $\lim_{d \to 0} \mu_i = 1$ for a monotonically decreasing function, where $d_p(\boldsymbol{x}_i, \boldsymbol{x}_r)$ is any member of the generalized Minkowski family of metrics.

The resulting membership function has the S shape (sigmoidal) required by Zadeh's design [72]. Due to the lack of break points, sigmoidal functions are best suited to represent natural, continuous behavior. The crossover point (the point assigned a membership value of 0.5), is also the inflection point between the convex and the concave part in the sigmoidal function defined in (27)–(28). By construction, the function never reaches absolute truth (or falsehood) values due to the asymptotic behavior to both values. If this constitutes a problem for a particular application domain it can be resolved by introducing the appropriate break points at $x = a_1 + \sigma_1$ ($x = a_1 - \sigma_1$) and $x = a_2 + \sigma_2$ ($x = a_2 - \sigma_2$).

In addition, the membership function needs a parameter c, which controls its dispersion characteristics. Dispersion is defined as the range between the crossover points and the nearest entry which receives the maximum value of one. The dispersion value regulates the fuzzification process and is a design parameter which can be tuned to modify the fuzziness of the membership function. The parameters for crossover points and dispersion are the minimum requirements for determining a fuzzy membership function.

In cases where a distance measure is used to quantify dissimilarity between the vector under consideration and the ideal prototype, the decreasing form of the function is

utilized. If a similarity measure is used instead, we consider the monotonically increasing version of the membership function.

Although not supported by the general form of the membership function introduced in [67], the exponential (Gaussian) kernel was used by the authors as an approximation to the membership function, especially when Minkowski metrics ($d_p(\boldsymbol{x}_i, \boldsymbol{x}_r)$) are used to quantify distance between multichannel signals (color vectors in particular).

In this case the proposed membership function can be defined as

$$\mu_i(\boldsymbol{x}_i) = \exp\left(\frac{-d_p(\boldsymbol{x}_i, \boldsymbol{x}_r)^\lambda}{\beta}\right) \quad (29)$$

where λ is a positive constant and β is a distance threshold. The membership model proposed, although defined empirically, complies with certain psychometrical experiments [73]. According to [73], quantification of similarity of an unknown stimulus to a known prototype can be expressed as a simple exponential decay or Gaussian function of a normalized distance in a psychological space. The two parameters are design parameters. That is, their actual values vary with the application. The above parameters correspond to the denominational and exponential fuzzy generators controlling the amount of fuzziness. A gradient method can be applied to optimize the parameters of the membership function. However, since the training signal should be expressed as a nonlinear form of the parameters to be controlled, the convergence speed can be very slow or a local minimum may exist in some cases. In addition, the training signal should be selected to match the characteristics of the input signal to be processed by the fuzzy adaptive system. Alternative methodologies to adjust the parameters of the membership function include neural network techniques and evolutionary computation, to name a few [23].

A. The Generalized Membership Function

The model of (27)–(28) satisfies the requirement imposed by the adaptive fuzzy framework. However, the membership function in its present form is computationally expensive since it involves the evaluation of the exponential function, and more importantly its parameters cannot be evaluated easily in practical applications. Therefore, other functions which can retain the same characteristics and are easier to implement are needed. Such a membership form was proposed in [74]. This function is continuously increasing (decreasing), satisfies the same boundaries conditions, complies with the generic membership form of (13), and retains the properties of the S-shaped membership function. However, unlike the function in (27)–(28), it can be written as a rational function of two polynomials.

In the new formulation, for any input value z, the membership function, by construction, can be completely characterized by only four parameters: 1) and 2) the interval $[a, b]$ of the input parameter z; 3) the sharpness λ of the membership function; and 4) the inflection point ν of the S-shaped function. Based on these parameters a membership function can be defined as

$$\mu(z) = \frac{(1-\nu)^{\lambda-1}(z-a)^\lambda}{(1-\nu)^{\lambda-1}(z-a)^\lambda + \mu^{\lambda-1}(b-z)^\lambda} \quad (30)$$

for a monotonically increasing function, and

$$\mu(z) = \frac{(1-\nu)^{\lambda-1}(b-z)^\lambda}{(1-\nu)^{\lambda-1}(b-z)^\lambda + \mu^{\lambda-1}(z-a)^\lambda} \quad (31)$$

for a monotonically decreasing function, with the inflection point ν defined via $\mu(z_\nu) = \nu$ and $z_\nu = (b-a)\nu + a$ or $z_\nu = (a-b)\nu + b$ for the case of monotonically increasing or decreasing functions, respectively.

The sharpness of the function (an indicator of increasing/decreasing membership) can be defined respectively as

$$\lambda = f'(z - \nu)(b - a). \quad (32)$$

This membership function, which is universally applicable, can be utilized by considering the distance or similarity value as the input to the membership function. Assume that $d_i = d_p(\boldsymbol{x}_i, \boldsymbol{x}_r)$ and $s_i = s_i(\boldsymbol{x}_i - \boldsymbol{x}_r)$ are appropriate distance or similarity measures between the vector under consideration and the ideal prototype, similar to those discussed in the previous section, we may rewrite the membership function needed in the fuzzy system of (3) as follows:

$$\mu_i = \frac{(1-\nu)^{\lambda-1}(d_{\max} - d_i)^\lambda}{(1-\nu)^{\lambda-1}(d_{\max} - d_i)^\lambda + \mu^{\lambda-1}(d_i - d_{\min})^\lambda} \quad (33)$$

with $d_i \in [d_{\max}, d_{\min}]$.

Alternatively, a monotonically increasing function can be defined based on a similarity measure $s_i = s_i(\boldsymbol{x}_i - \boldsymbol{x}_r)$ as follows:

$$\mu_i = \frac{(1-\nu)^{\lambda-1}(s_i - s_{\min})^\lambda}{(1-\nu)^{\lambda-1}(s_i - s_{\min})^\lambda + \mu^{\lambda-1}(s_{\max} - s_i)^\lambda} \quad (34)$$

with $s_i \in [s_{\max}, s_{\min}]$.

For the case of $\lambda = 1$ the linear form of the membership function is obtained

$$\mu_i = \frac{s_i - s_{\min}}{s_{\max} - s_{\min}} \quad (35)$$

for a monotonically increasing function, and

$$\mu_i = \frac{d_{\max} - d_i}{d_{\max} - d_{\min}} \quad (36)$$

for a monotonically decreasing function, which corresponds to the nearest-neighbor rule used in [46], [49], and [50] to define membership functions.

The membership function is critically dependent on the similarity measure and the reference point selected. The ideal reference signal is the actual value of the multidimensional signal in the specific location under consideration. This signal, however, is not available. In addition, the noisy vector at the same location is not the appropriate choice since any input vector inside the window can be an outlier. Therefore, to make the procedure more robust and to ensure that the fuzzy system will provide accurate

results, we eliminate the need for a reference point by evaluating the membership function used to weight each input vector $\boldsymbol{x}_i$ in (3) on the aggregate distance between the vector $\boldsymbol{x}_i$ under consideration and all the other vectors inside the processing window. Thus, the vector with the smallest overall distance (or maximum similarity) is now assigned the maximum membership value. Needless to say the membership function selected is now evaluated on the aggregated distances and not on the distance between the vector and the ideal prototype.

It is obvious that such a design does not depend on a reference point and thus is more robust to occasional outliers. However, the computational complexity of the algorithm increases as a result of the need to evaluate a number of distances (similarities) in the processing window. Any distance metric or similarity function discussed in the previous section can be used in the formulation of the aggregate distance. For example, assume that the noisy vector $\boldsymbol{x}_i$ inside the processing window W is considered. Its aggregated distance from all other vectors inside the window is given as

$$d_i = \sum_{j=1}^{n} d_2(\boldsymbol{x}_i, \boldsymbol{x}_j) \qquad (37)$$

$$d_2(i,j) = \left(\sum_{k=1}^{m} (x_i^k - x_j^k)^2 \right)^{1/2} \qquad (38)$$

if the Euclidean metric has been selected to measure dissimilarity between two vector signals. This aggregate distance value is used as an input to the membership functions of (29) or (33) that will be used to determine the fuzzy weights in the multichannel filter.

Similarly, the vector angle criterion defines the scalar measure

$$d_{ai} = \sum_{j=1}^{n} A(\boldsymbol{x}_i, \boldsymbol{x}_j) \qquad (39)$$

as the distance associated with the noisy vector $\boldsymbol{x}_i$ inside the processing window of length n, when the angle between two vectors

$$A(\boldsymbol{x}_i, \boldsymbol{x}_j) = \cos^{-1} \left(\frac{\boldsymbol{x}_i \boldsymbol{x}_j^T}{|\boldsymbol{x}_i||\boldsymbol{x}_j|} \right) \qquad (40)$$

is used to measure dissimilarity. The approach suggested here eliminates the need for a reference point and generalizes the concept of membership function as a similarity indicator. In the suggested formulation, the valuation of membership is regarded not as a comparison with an ideal point but as a comparison to the rest of the elements to be included in the fuzzy set.

An alternative, although suboptimal, procedure has been proposed by the authors. In this approach, a robust estimate of the location, usually evaluated in a smaller subset of the input vector set, is utilized as the reference vector. The selection of this robust reference vector depends on the signal characteristics. Usually the median is the preferable choice since it smooths out impulsive noise and preserve

edges and details. Moreover, unlike scalars, the centermost vector in a set of vectors can be defined in more than one way. Thus, the VMF, the basic vector directional filter (BVDF), or the marginal median filter (MAMF) operating in a processing window centered around the current vector input can be used to provide the requested reliable reference point [47].

The proposed adaptive fuzzy filter can be viewed as a double-window two-stage estimator in which we can distinguish between two operations. First, the original signal is filtered by a multichannel median filter in a small processing window in order to reject possible outliers, then an adaptive fuzzy filter with data dependent weights is applied to provide the final estimates. Thus, the overall filter can be viewed as a combined multichannel operator which incorporates simple nonlinear statistical estimators such as the VMF into adaptive designs based on fuzzy membership functions.

To summarize, we have outlined a possible interpretation of the membership function and discussed how membership functions can be built based on similarity concepts. A generalized model for building membership functions was utilized here. The different similarity or distance measures discussed here can be used as input values to this membership function model. The possibility of tuning the design parameters, namely distance (similarity) metrics and membership functions provides the adaptive fuzzy system of (3) with a rich variety of modes of operation that range from simple selection type filters, such as the VMF to hybrid filters, such as the α-trimmed filter.

Now that we have completed the presentation of the fuzzy, adaptive framework we can proceed with the presentation of the different multichannel filters which can be devised within this framework.

B. Adaptive Fuzzy Filters

The filter structure reviewed in this paper can support different filter families often used in multichannel signal processing. To illustrate the point we will discuss next the derivation of commonly used filters through the adaptive fuzzy structure.

In particular, we shall start with mode-like filters such as the VMF, which is the most widely used windowed nonlinear, multichannel filter. The vector median of the input set is defined as the minimal vector according to the aggregate, reduced ordering technique (R-ordering) of [35], [48], [59], and [75]–[77].

Let us assume that given a set of (n) noisy input signals $\boldsymbol{x}_j, j = 1, 2, \ldots, n$ inside a processing window the scalar quantity $d_i = \sum_{j=1}^{n} d_1(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is the aggregated distance associated with the noisy color vector $\boldsymbol{x}_i$, where $d_1(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is the city-block metric.

Assume further that an ordering of the d_i's $d_{(1)} \leq d_{(2)} \leq \cdots \leq d_{(n)}$ implies the same ordering to the corresponding $\boldsymbol{x}_i$'s $\boldsymbol{x}_{(1)} \leq \boldsymbol{x}_{(2)} \leq \cdots \leq \boldsymbol{x}_{(n)}$, where $\boldsymbol{x}_{(i)}$ refers to the ith ordered sample. The VMF defines the vector $\boldsymbol{x}_{(1)}$ as the filter output. This selection arises because vectors which

diverge greatly from the data population usually appear in higher indexed locations in the ordered sequence [9].

VMF is a selection filter in which the output is always constrained by definition to one of the samples inside the processing window. It can easily be seen that the VMF can be derived as a special case of the fuzzy adaptive system of (3). Indeed, by adopting the maximum defuzzifier of (12) in conjunction with aggregated city-block metrics to drive the membership function of (36) the same filtered output can be obtained. Simple inspection of (36) reveals that since $d_{\min} = d_{(1)}$ by construction, the membership function for the corresponding input $\boldsymbol{x}_{(1)}$ is $\mu_{(1)} = (d_{\max} - d_{\min}/d_{\max} - d_{\min}) = 1$ and thus $\boldsymbol{x}_{(1)}$ is selected through (12) as the filtered output.

The VMF is not the only multichannel filter which can be used for processing vector signals. A new type of vector processing filter, the vector directional filter (VDF) was proposed recently for multichannel, color signals [78]. The VDF operates on the direction of the color image signals, aiming at eliminating vectors with atypical directions in the vector space. To achieve its objective, the VDF utilizes the angular distance of (39) to order the input vectors inside a processing window. As a result, a set of input vectors with approximately the same direction in the vector space is produced as the output set. Since the vectors in this set are approximately collinear, a magnitude processing operation can be applied in a second step to produce the requested filtered output.

The BVDF is a selection, nonlinear filter which parallelizes the VMF operation. However, it employs the angle between the two color vectors as the distance criterion. The output of the BDVF is the vector from the input set which minimizes the sum of the angles with the other vectors. In other words, the BVDF chooses the vector most centrally located without considering the magnitudes of the input vectors. It may perform well when the vector magnitudes are of no importance and the direction of the vectors is the dominant factor. However, this is usually not the case. In signal processing applications, including color image processing, the magnitudes of the vectors should also be considered. To improve the performance of the BVDF, a generalized filter structure was proposed [59]–[78]. The new filter, appropriately called the generalized vector directional filter (GVDF), generalizes the BVDF in the sense that its output is a superset of the single BVDF output. Instead of a single output, the GVDF outputs the set of vectors whose angle from all other vectors is small as opposed to the BVDF, which outputs the vector whose angle from all the other vectors is minimum. Thus, the GVDF's produced output initially consists of a set of (φ), $(\varphi < n)$ input vectors with approximately the same direction in the color space. Then, in the magnitude processing module, a final single vector output is produced by considering only magnitude information. As before, the BVDF can be derived through the generalized filtering structure of (3). Simply the vector angle criterion should be used in conjunction with the maximum defuzzifier and the membership function of (36).

It is not necessary for the designer to use all the inputs inside the filtering window to produce the final output in the fuzzy filter. If necessary, only a part of the input set can be used. The input vectors can be ordered according to their respective fuzzy membership strengths. In the sequence, only a subset of them are used to form the filtered output. Thus the fuzzy filter can be rewritten as

$$
\hat{\boldsymbol{y}} = \frac{\displaystyle\sum_{j=1}^{\varphi} w_{(j)} \boldsymbol{x}_j}{\displaystyle\sum_{j=1}^{\varphi} w_{(j)}} \tag{41}
$$

where $w_{(j)}$ represents the jth ordered fuzzy membership function, and $w_{(\varphi)} \leq w_{(\varphi-1)} \leq \cdots \leq w_{(1)}$, with $w_{(1)}$ being the fuzzy coefficient with the largest membership strength. The present form of the filter is a special case of the general filter introduced by (3). The $f(\mu_j)$ used here is the ordering of the fuzzy weights. The above form of the algorithm constitutes a fuzzy generalization of the α-trimmed filters [2]. Through the fuzzy transformation, the weights to be sorted are scalar values. In this way the non-linear ordering process does not introduce any significant computational burden. Depending on the distance criterion and the associated fuzzy form that the designer chooses, a number of different α-trimmed filters can be obtained. If the vector angle criterion is selected, the GVDF can be derived through (41). The question which arises is how to select the appropriate number of input vectors that will be included in the final output. There is no standard procedure to determine the number of inputs that are trimmed and not included in the averaging process. There are two different ways to determine the number of vectors that have to be included in the final set that produces the fuzzy output. The first option is through the selection of a fixed number of inputs vectors. The filter designer selects (φ) input vectors that correspond to the (φ) fuzzy weights with the largest values. On the other hand, the number of vectors can be determined in an adaptive fashion. A simple approach is to include all the vectors associated with a fuzzy weight larger than a given threshold. To this end, we select a threshold value of $t = (1/n)$ where n are the vectors inside the operational window.

The last example considered here is another selection filter known as the directional-distance filter (DDF) which was proposed in [78]. The DDF retains the structure of the BVDF but utilizes a new distance criterion to order the vectors inside the processing window. Based on the observation that the BVDF and the VMF differ only in the quantity that is minimized, a new distance criterion was utilized by the designers of the DDF in the hopes of deriving a filter which combines the properties of both these filters. Specifically, in the case of the DDF, the distance inside W is defined as

$$
\beta_i = \sum_{j=1}^{n} A(\boldsymbol{x}_i, \boldsymbol{x}_j) \sum_{j=1}^{n} \|\boldsymbol{x}_i, \boldsymbol{x}_j\| \tag{42}
$$

where $A(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is the directional (angular) distance defined in (9) with the second term in (11) to account for the differences in magnitude in terms of the L_1 metric. As for any other mode-type, multichannel, nonlinear filter, it is assumed that an ordering of the β_i distance $\beta_{(1)} \leq \beta_{(2)} \leq \cdots \leq \beta_{(n)}$ implies the same ordering to the corresponding input vectors $\boldsymbol{x}_i$'s: $\boldsymbol{x}_{(1)} \leq \boldsymbol{x}_{(2)} \leq \cdots \leq \boldsymbol{x}_{(n)}$. Thus, the DDF defines the minimum-order vector as its output: $\boldsymbol{x}_{DDF} = \boldsymbol{x}_{(1)}$. By incorporating the proposed distance function in the linear membership function of (36), the same result is returned by the adaptive fuzzy filter as when the maximum defuzzifier is employed.

Although a number of different adaptive designs were discussed here, all of them have some common design characteristics and exhibit similar behavior. We summarize a number of them in a series of comments.

1) All the adaptive vector processing filters perform smoothing of all the vectors which are from the same region as the vector at the window center. It is reasonable to make their fuzzy weights proportional to the difference (similarity), in terms of a distance measure, between a given vector and its neighbors inside the operational window. At edges, or in areas with high details, the filter only smooths inputs on the same side of the edge as the centermost vector, since vectors with relatively large distance values will be assigned smaller weights and will contribute less to the final filter output. Thus through the utilization of the fuzzy adaptive designs we are able to not only preserve the signal characteristics but also to reduce the computational effort by avoiding prefiltering operations, such as edge or line detection operations. The proposed adaptive framework combines elements from almost all known classes of nonlinear filters. Namely, it combines Minkowski-type distances (used in order-statistics-based estimators) or nonmetric content-based similarity measures (used in ranked-type estimators), averaging outputs (used in linear filtering), with data-dependent coefficients used in adaptive designs and membership functions used in fuzzy systems.

2) In the framework described above, there is no requirement for fuzzy rules or local statistics estimates. Features extracted from local data, here in the form of distances or similarities, are used as inputs to the membership function. The fuzzy filters discussed in this section do not utilize the distance measures to order the noisy input signals. Instead, they are used to provide selected features in a reduced space: features used as inputs for the adaptive weights.

3) The filtering algorithms differ in their computational complexity. It should be noted at this point that the computational complexity of a given filter is a realistic measure of its practicality and usefulness, since it determines the required computing power and the associated processing time required for its implementation. The computational complexity analysis of the adaptive designs requires knowledge of the

Table 1
Noise Models

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

membership function used to calculate the adaptive weights and the exact form of the selected distance (similarity) measure used. The computationally intensive part of the adaptive scheme rests in calculating the distance. This part, however, is common to all vector processing designs. Thus from a practical standpoint, the remarkably flexible structure of (16) yields realizations of different filters that can meet a number of design constraints including hardware and computational complexity.

VI. COLOR IMAGE PROCESSING

The adaptive fuzzy filters discussed here can be used to process multichannel signals in a variety of practical applications, such as color image processing, medical imaging, remote sensing applications, geophysical signal processing, and military communications. Due to numerous practical applications, color images comprise an important class of multichannel signals, and thus they can serve as an excellent illustrative application.

The international standard *CIE 1931* defines color curves based on tristimulus values of human capabilities and conditions of view [79]–[82]. The basis of the trichromatic theory of color vision is that it is possible to match an arbitrary color by superimposing appropriate amounts of three primary colors. Thus, in the different color spaces, each pixel of an image is represented by three values which can be considered as a vector, transforming the color image to a vector field in which each vector's direction and length is related to the pixel's chromatic properties. Being a two-dimensional, three-channel signal, a color image requires increased computation and storage during processing, as compared to a greyscale image.

We have conducted a set of experiments in order to evaluate the fuzzy adaptive designs and compare their performance against the performance of other filters, such as the arithmetic mean filter (AMF), a simple weighted average filter, the VMF, the DDF, and the hybrid filters of [83].

The noise attenuation properties of the different filters are examined by utilizing the color images "Lenna" and "Peppers." The test images have been contaminated using various noise source models in order to assess the performance of the filters under different scenarios (see Table 1). Image transmission noise exists in many practical applications. There are various sources that can generate this type of noise including many man made phenomena,

Table 2
Filters Compared

Notation	Filter	Reference
BVDF	Basic Vector Directional Filter	[59],[60]
CBRF	Content-based Rank Filter, eq. (25)	[50]
GVDF	Generalized Vector Directional Filter	[59],[78]
	with an α-trimmed magnitude module, ($\alpha = 1.5$)	
DDF	Directional-Distance Filter	[78]
VMF	Vector Median Filter	[9]
FVDF	Fuzzy Vector Directional Filter	[34]
	with weights determined through (29), $\lambda = 1$, $\beta = 2$	
ANNF	Adaptive Nearest Neighbor Filter, eqs. (36), (37), (38)	[35]
ANNMF	Double window Adaptive Nearest Neighbor Filter, eqs. (36), (38), (40)	[47]
HF	Hybrid Directional Filter	[83]
AHF	Adaptive Hybrid Directional Filter	[83]
CANNF	Adaptive Nearest Neighbor Filter eqs. (36) and (15)	
CANNMF	Double window adaptive nearest neighbor filter, eqs. (36) and (15)	
CBANNF	Adaptive Nearest Neighbor Filter, eqs. (36), (37) and (25)	
CBANNMF	Double window adaptive nearest neighbor filter, eqs. (36) and (25)	

such as car ignition systems, industrial machines in the vicinity of the receiver, switching transients in power lines, and various unprotected switches. In addition, natural causes, such as lightning in the atmosphere and ice cracking in the Antarctic region, can also affect the transmission process. This transmission noise, also known as salt-and-pepper noise in greyscale imaging, is modeled after an impulsive distribution. However, a problem in the study of the effect of the noise in the image processing process is the lack of a multivariate impulsive noise model. A number of simplified models have been introduced recently to assist in the performance evaluation of the different color image filters.

The impulsive noise model considered here is as follows:

$$\boldsymbol{n}_{Io} = \begin{cases} \boldsymbol{x}, & \text{with probability } (1-p) \\ (d, x_2, x_3)^T, & \text{with probability } p_1 p \\ (x_1, d, x_3)^T, & \text{with probability } p_2 p \\ (x_1, x_2, d)^T, & \text{with probability } p_3 p \\ (d, d, d)^T, & \text{with probability } p_\Sigma p \end{cases} \quad (43)$$

where n_{Io} is the noisy signal, $\boldsymbol{x} = (x_1, x_2, x_3)^T$ is the noise free color vector, d is the impulse value, and

$$p_\Sigma = 1 - p_1 - p_2 - p_3 \quad (44)$$

with $\Sigma_{i=1}^3 p_i \leq 1$ the degree of impulsive noise contamination. Impulse d can have either positive or negative values. We further assume that $d \gg s_1 \gg s_2 \gg s_3$ and that the delta functions are situated at ($+255$, -255). Thus, when an impulse is added or subtracted, forcing the pixel value outside the [0, 255] range, clipping is applied to force the

corrupted noise value into the integer range specified by the 8-bit arithmetic.

In many practical situations an image is often corrupted by both additive Gaussian noise due to faulty sensors and transmission noise introduced by environmental interference or faulty communication. An image can therefore be thought of as being corrupted by mixed noise according to the following model:

$$\boldsymbol{y} = \begin{cases} \boldsymbol{x} + \boldsymbol{n}_o(\boldsymbol{x}), & \text{with probability } (1 - p_I) \\ \boldsymbol{n}_{Io}, & \text{otherwise} \end{cases} \quad (45)$$

where $s(\boldsymbol{x})$ is the noise free three-variate color signal with the additive noise $n(\boldsymbol{x})$ modeled as zero mean white Gaussian noise and $n_{Io}(\boldsymbol{x})$ transmission noise modeled as multivariate impulsive noise with p_I the degree of impulsive noise contamination [84], [85].

The original images as well as their noisy versions are represented in the RGB color space. This color coordinate system is considered to be objective since it is based on the physical measurements of the color attributes. The filters operate on the images in the RGB color space.

Since it is impossible to discuss all the fuzzy adaptive filters resulting from the theory introduced here, we instead construct five different filters based on our designs. These filters are compared in terms of performance with other widely used multichannel filters (see Table 2). In particular, we introduce a simple rank-order filter based on the distance measure of [50] [hereafter referred to as content-based rank filter (CBRF)], which can be seen as an adaptive fuzzy system with the defuzzification rule of (12). We also include the fuzzy vector directional filter (FVDF) which is based on

Fig. 1. "Peppers" corrupted by 4% impulsive noise.

Fig. 3. BVDF of Fig. 1 using 3×3 window.

Fig. 2. VMF of Fig. 1 using 3×3 window.

Fig. 4. HF of Fig. 1 using 3×3 window.

the defuzzification strategy of (6), the membership formula of (29), and the aggregated distance of (39) evaluated over the filtering window W. The adaptive nearest-neighbor filter (ANNF) based on the defuzzification strategy of (6), the membership function formula of (36), and the distance measure of (38) are also included in the set. Further, we utilized the same defuzzification formula and the same membership function, along with the aggregated distance of (38) to derive the double window nearest neighbor filter ANNMF. By using the Canberra distance and the distance measure of (25) instead of the angular distance, four new filters have been devised, namely the CANNF, CANNMF, CBANNF, and the CBANNMF (see Table 2).

A number of different objective measures can be utilized to assess the performance of the different filters. All of them provide some measure of closeness between two

digital images by exploiting the differences in the statistical distributions of the pixel values [48]. The most widely used measure is the normalized mean square error (NMSE) defined as

$$\text{NMSE} = \frac{\sum_{i=0}^{N1} \sum_{j=0}^{N2} \|(\boldsymbol{y}(i,j) - \hat{\boldsymbol{y}}(i,j)\|^2}{\sum_{i=0}^{N1} \sum_{j=0}^{N2} \|(\boldsymbol{y}(i,j)\|^2} \qquad (46)$$

where $N1$ and $N2$ are the image dimensions, and $\boldsymbol{y}(i,j)$ and $\hat{\boldsymbol{y}}(i,j)$ denote the original image vector and the estimation at pixel (i,j), respectively.

In many application areas, such as multimedia, telecommunications [e.g., high-definiton television (HDTV)], production of motion pictures, the printing industry, and

Fig. 5. AHF of Fig. 1 using 3×3 window.

Fig. 6. FVDF of Fig. 1 using 3×3 window.

Fig. 7. ANNMF of Fig. 1 using 3×3 window.

Fig. 8. CANNMF of Fig. 1 using 3×3 window.

graphic arts, greater emphasis is given to perceptual image quality. Consequently, the perceptual closeness (alternatively perceptual difference or error) of the filtered image to the uncorrupted original image is ultimately the best measure of the efficiency of any color image filtering method. There are basically two major approaches used for assessing the perceptual error between two color images. In order to make a complete and thorough assessment of the performance of the various filters, both approaches are used in this paper.

The first approach is to make an objective measure of the perceptual error between two color images. This leads us to the question of how to estimate the perceptual error between two color vectors. Precise quantification of the perceptual error between two color vectors is one of the most important and open research problems. RGB

is the most popular color space used conventionally to store, process, display, and analyze color images. However, the human perception of color cannot be described using the RGB model. Therefore, measures such as the NMSE defined in the RGB color space are not appropriate to quantify the perceptual error between images. Thus it is important to use color spaces which are closely related to the human perceptual characteristics and suitable for defining appropriate measures of perceptual error between color vectors. A number of such color spaces are used lately in areas such as computer graphics, motion pictures, graphic arts, and the printing industry. Among these, perceptually uniform color spaces are the most appropriate to define simple yet precise measures of perceptual error. The Commission Internationale de l'Eclairage (CIE) standardized two color spaces, $L^*u^*v^*$ and $L^*a^*b^*$, as perceptually

Fig. 9. "Lenna" corrupted by Gaussian noise $\sigma = 15$ mixed with 2% impulsive noise.

Fig. 11. BVDF of Fig. 9 using 3×3 window.

Fig. 10. VMF of Fig. 9 using 3×3 window.

Fig. 12. HF of Fig. 9 using 3×3 window.

uniform. The $L^*u^*v^*$ color space is chosen for our analysis because it is simpler in computation than the $L^*a^*b^*$ color space, without sacrificing perceptual uniformity.

The conversion from nonlinear RGB color space (the nonlinear RGB values are the ones stored in the computer and applied to the CRT of the monitor to generate the image) to the $L^*u^*v^*$ color space is explained in detail in [80]. Nonlinear RGB values of both the uncorrupted original image and the filtered image are converted to corresponding $L^*u^*v^*$ values for each of the filtering methods under consideration. In the $L^*u^*v^*$ space, the L^* component defines the lightness and the u^* and v^* components together define the chromaticity. In a uniform color space, such as $L^*u^*v^*$, the perceptual color error between two color vectors is defined as the Euclidean

distance between them given by

$$\Delta E_{Luv} = \left[(\Delta L^*)^2 + (\Delta u^*)^2 + (\Delta v^*)^2 \right]^{1/2} \quad (47)$$

where ΔE_{Luv} is the color error and ΔL^*, Δu^*, and Δv^* are the difference in the L^*, u^*, and v^* components, respectively, between the two color vectors under consideration. Once the ΔE_{Luv} for each pixel of the images under consideration is computed, the normalized color difference (NCD) is estimated according to the following formula:

$$\text{NCD} = \frac{\sum_{i=0}^{N1} \sum_{j=0}^{N2} \|\Delta E_{Luv}\|}{\sum_{i=0}^{N1} \sum_{j=0}^{N2} \|E_{Luv}^*\|} \quad (48)$$

Fig. 13. AHF of Fig. 9 using 3×3 window.

Fig. 15. ANNMF of Fig. 9 using 3×3 window.

Fig. 14. FVDF of Fig. 9 using 3×3 window.

Fig. 16. CANNMF of Fig. 9 using 3×3 window.

where $E_{Luv}^* = [(L^*)^2 + (u^*)^2 + (v^*)^2]^{1/2}$ is the norm or magnitude of the uncorrupted original image pixel vector in the $L^*u^*v^*$ space.

Although quantitative measures such as ΔE_{uv} and NCD are close approximations to the perceptual error they cannot exactly characterize the quite complex attributes of the human perception. Therefore an alternative subjective approach was used to estimate the perceptual error [86].

The second approach, the easiest and simplest, is the subjective evaluation of the resulting images when they are viewed, simultaneously, under identical viewing conditions by a set of observers. To this end, we compare the performance of the different filters in noise attenuation using two RGB images.

Filtering results for the test image "Peppers" (Fig. 1) are depicted in Figs. 2–8 while filtering results for the

test image "Lenna" (Fig. 9) are depicted in Figs. 10–16. A visual comparison of the images clearly favors the adaptive designs over existing techniques.

Having defined the measures for the assessment of performance by various filters, we proceed to the assessment itself.

One of the obvious observations from the results in Tables 3–10 is the effect of window size on the performance of the filter. In the case of rank-type filters, such as the VMF, the BVDF, the CBVF, the DDF, as well as the hybrid directional filter (HF) and the adaptive hybrid directional filter (AHF), the bigger window size (5×5) gives considerably better results for the removal of Gaussian noise (noise model 1), while decreasing the performance for the removal of impulsive noise (noise model 2). Although a similar pattern follows for the adaptive fuzzy filters, the

Table 3
NMSE($\times 10^{-2}$) for the RGB "Lenna" Image, 3×3 Window

Filter	Noise Model			
	1	2	3	4
None	4.2083	5.1694	3.6600	9.0724
BVDF	2.8962	0.3448	0.4630	1.1354
CBRF	1.3990	0.1863	0.5280	1.5168
GVDF	1.4600	0.3000	0.6334	1.9820
DDF	1.5240	0.3255	0.6483	1.6791
VMF	1.6000	0.1900	0.5404	1.6790
FVDF	0.7335	0.2481	0.4010	1.0390
ANNF	0.8510	0.2610	0.3837	1.0860
ANNMF	0.6591	0.1930	0.3264	0.7988
HF	1.3192	0.2182	0.5158	1.6912
AHF	1.0585	0.2017	0.4636	1.4355
CANNF	0.8360	0.2497	0.3471	1.0481
CANNMF	0.6001	0.1891	0.3087	0.7137
CBANNF	0.8398	0.2349	0.3935	1.0119
CBANNMF	0.6011	0.1894	0.3087	0.7149

Table 5
NMSE($\times 10^{-2}$) for the RGB "Peppers" Image, 3×3 Window

Filter	Noise Model			
	1	2	3	4
None	5.0264	6.5257	3.2890	6.5076
BVDF	3.9267	1.5070	0.8600	1.4911
CBRF	1.9622	0.4650	0.4354	0.4711
GVDF	1.8640	0.4550	0.3613	0.4562
DDF	3.5090	0.5886	0.5336	0.5893
VMF	1.8440	0.3763	0.3260	0.3786
FVDF	1.4550	0.4246	0.3412	0.4046
ANNF	1.1230	0.5110	0.3150	0.5180
ANNMF	0.9080	0.3550	0.3005	0.3347
HF	1.5892	0.4690	0.3592	0.4781
AHF	1.4278	0.4246	0.3566	0.4692
CANNF	1.1382	0.4696	0.3492	0.4699
CANNMF	0.8994	0.4526	0.4284	0.4545
CBANNF	1.2246	0.4546	0.4566	0.4548
CBANNMF	0.8964	0.4546	0.4300	0.4548

Table 4
NMSE($\times 10^{-2}$) for the RGB "Lenna" Image, 5×5 Window

Filter	Noise Model			
	1	2	3	4
None	4.2083	5.1694	3.6600	9.0724
BVDF	2.800	0.7318	0.6850	1.3557
CBRF	0.9258	0.3180	0.4890	1.0061
GVDF	1.0800	0.5400	0.4590	1.1044
DDF	1.0242	0.5126	0.6913	1.3048
VMF	1.1700	0.5800	0.5172	1.0377
FVDF	0.7549	0.3087	0.4076	0.9550
ANNF	0.6260	0.4210	0.4360	0.7528
ANNMF	0.5445	0.2505	0.3426	0.6211
HF	0.7700	0.3841	0,4890	1.1417
AHF	0.6762	0.3772	0.4367	0.7528
CANNF	0.5950	0.4028	0.4091	0.7380
CANNMF	0.5208	0.3017	0.3671	0.5802
CBANNF	0.5925	0.3943	0.4045	0.7111
CBANNMF	0.5201	0.3014	0.3662	0.5795

Table 6
NMSE($\times 10^{-2}$) for the RGB "Peppers" Image, 5×5 Window

Filter	Noise Model			
	1	2	3	4
None	5.0264	6.5257	3.2890	6.5076
BVDF	4.2698	2.7920	1.6499	4.1350
CBRF	1.4639	0.7090	0.6816	0.7161
GVDF	1.2534	0.6977	0.6600	0.7030
DDF	2.1440	0.7636	0.7397	0.7612
VMF	1.3390	0.6740	0.6563	0.6812
FVDF	2.1120	0.7310	0.6971	0.7178
ANNF	1.0027	0.5230	0.5200	0.6210
ANNMF	0.8050	0.4471	0.4047	0.4458
HF	1.0040	0.9970	0.7684	0.9970
AHF	1.1167	0.9841	0.7632	0.9841
CANNF	1.0281	0.7393	0.6718	0.7426
CANNMF	0.8687	0.6355	0.6405	0.6420
CBANNF	1.0145	0.7281	0.6677	0.7310
CBANNMF	0.8634	0.6338	0.6313	0.6371

effect of the window size on performance is less dramatic as compared to the rank-type filters.

Analysis of the results summarized here reveals the effect that the distance (or similarity) measure can have on the filter output. Even filters which are based on the same concept, such as VDF, CVDF, CBVF, ANNF, and CANNF, have different performance simply because a different dis-

tance measure is utilized to quantify dissimilarity among the color vectors. Similarly, double window adaptive filters have better smoothing abilities, outperforming the other filters under consideration when a Gaussian noise or mixed noise model is assumed.

For the case of impulsive noise, the VMF gives the best performance among the rank-type filters according

Table 7
NCD for the RGB "Lenna" Image, 3 × 3 Window

Filter	Noise Model			
	1	2	3	4
None	0.1149	0.0875	0.7338	0.1908
BVDF	0.0508	0.0082	0.0210	0.0708
CBRF	0.0467	0.0051	0.0169	0.0524
GVDF	0.0462	0.0079	0.0191	0.0489
DDF	0.0398	0.0073	0.0179	0.0426
VMF	0.0432	0.0053	0.0238	0.0419
FVDF	0.0377	0.0049	0.0144	0.0394
ANNF	0.0338	0.0061	0.0149	0.0412
ANNMF	0.0316	0.0047	0.01374	0.0402
HF	0.03824	0.0061	0.0147	0.0486
AHF	0.0347	0.0593	0.0139	0.0442
CANNF	0.0222	0.0057	0.0090	0.0255
CANNMF	0.0175	0.0046	0.0081	0.0193
CBANNF	0.0229	0.0055	0.0089	0.0250
CBANNMF	0.0175	0.0046	0.0081	0.01934

Table 8
NCD for the RGB "Lenna" Image, 5 × 5 Window

Filter	Noise Model			
	1	2	3	4
None	0.1149	0.0875	0.7338	0.1908
BVDF	0.0408	0.0084	0.0267	0.0631
CBRF	0.0284	0.0070	0.0130	0.0310
GVDF	0.0220	0.0089	0.0189	0.0474
DDF	0.0279	0.0079	0.0171	0.0368
VMF	0.0193	0.0062	0.0236	0.0344
FVDF	0.0218	0.0057	0.0129	0.0339
ANNF	0.0202	0.0071	0.0120	0.0329
ANNMF	0.0181	0.0059	0.0123	0.0318
HF	0.0199	0.0097	0.0123	0.01205
AHF	0.0188	0.0941	0.0120	0.0322
CANNF	0.0129	0.0078	0.0085	0.0153
CANNMF	0.0126	0.0063	0.0080	0.0134
CBANNF	0.0130	0.0077	0.0084	0.0150
CBANNMF	0.0126	0.0063	0.0080	0.0134

Table 9
NCD for the RGB "Peppers" Image, 3 × 3 Window

Filter	Noise Model			
	1	2	3	4
None	0.2414	0.0854	0.0831	0.0859
BVDF	0.1916	0.0774	0.0668	0.0775
CBRF	0.1579	0.0560	0.0541	0.0561
GVDF	0.1463	0.0631	0.0596	0.0639
DDF	0.2113	0.0678	0.0657	0.0679
VMF	0.1624	0.0559	0.0533	0.0558
FVDF	0.1217	0.0585	0.0558	0.0591
ANNF	0.1135	0.0642	0.0578	0.0643
ANNMF	0.0997	0.0575	0.0565	0.0579
HF	0.1406	0.0609	0.0553	0.0605
AHF	0.1346	0.0605	0.0557	0.0601
CANNF	0.1137	0.0610	0.0561	0.0610
CANNMF	0.1009	0.0571	0.0560	0.0574
CBANNF	0.1132	0.0605	0.0558	0.0606
CBANNMF	0.1007	0.0569	0.0559	0.0570

Table 10
NCD for the RGB "Peppers" Image, 5 × 5 Window

Filter	Noise Model			
	1	2	3	4
None	0.2414	0.0854	0.0831	0.0859
BVDF	0.186235	0.1056	0.0867	0.1047
CBRF	0.1281	0.0657	0.0646	0.0659
GVDF	0.1384	0.0941	0.0870	0.0946
DDF	0.1613	0.0706	0.0695	0.0706
VMF	0.1301	0.0662	0.0648	0.0663
FVDF	0.1310	0.0658	0.0644	0.0659
ANNF	0.0917	0.0760	0.0698	0.0760
ANNMF	0.0895	0.0657	0.0652	0.0658
HF	0.1118	0.0798	0.0697	0.0798
AHF	0.1070	0.0795	0.0699	0.0792
CANNF	0.0896	0.0652	0.0651	0.0659
CANNMF	0.0896	0.0652	0.0651	0.0659
CBANNF	0.0988	0.07246	0.06837	0.0725
CBANNMF	0.0893	0.0649	0.06452	0.0651

to the results, as well as the theory, and it is thus used as a benchmark to evaluate the fuzzy adaptive designs. The proposed fuzzy filters perform close to the VMF and outperform existing adaptive designs, such as the HF or the AHF with respect to NMSE and NCD, and for both window sizes.

For the case of pure Gaussian noise, the VMF gave the worst results. The results summarized in Tables 3–10 indicate that the adaptive fuzzy filters perform exceptionally well in this situation.

For the mixture of Gaussian and impulsive noise (noise models 3 and 4), the adaptive fuzzy filters consistently outperform any of the existing listed filters, both rank type or adaptive with respect to NMSE and NCD.

Herein lies the real advantage of the adaptive fuzzy designs. In real applications, the noise model is unknown *a*

priori. Nevertheless, the most common noise types encountered in real situations are Gaussian, impulsive, or a mixture of both. Therefore, the use of the presented fuzzy adaptive filters suggests near-optimal performance for the removal of any kind of noise encountered in practical applications. On the contrary, application of a "noise-mismatched" filter, such as the VMF for Gaussian noise, can have profound consequences leading to unacceptable results.

In conclusion, from the results listed in the tables, it can be easily seen that the adaptive designs provide consistently good results in all types of noise, outperforming the other multichannel filters under consideration. The adaptive fuzzy designs discussed here attenuate both impulsive and Gaussian noise. The versatile design of (3) allows for a number of different filters which can provide solutions to many types of different filtering problems. Simple adaptive fuzzy designs, such as the ANNF or the CANNF, 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 fuzzy system is the required distance calculation. However, this step is common in all multichannel algorithms considered here. In summary, the design is simple, does not increase the numerical complexity of the multichannel algorithm, and delivers excellent results for complicated multichannel signals, such as real color images.

VII. CONCLUSIONS

The paper presented a multichannel signal processing methodology based on fuzzy concepts. The framework combines nonlinear filters, fuzzy membership functions, and distance (similarity) criteria. Several filters can be considered special cases of this framework. The behavior of these adaptive designs was analyzed and their performance was compared with that of the most commonly used filters for a problem of great practical importance, namely color image processing. These filters not only have a rigid theoretical foundation but promising performance in a variety of noise characteristics. Indeed, the simulation results included and the subjective evaluation of the filtered color images indicate that the proposed fuzzy adaptive filters compare favorably with other commonly used techniques.

In conclusion, this paper summarizes the present state of knowledge on the subject of adaptive fuzzy systems and their application to the problem of multichannel signal processing. Particular emphasis was given to the formulation of the problem and the filter design procedure. Other issues, such as the realization of the different methodologies presented here, and implementation as well as verification issues, were not discussed in this paper. To assess fully the applicability of the presented techniques, further analysis is required on algorithms and architectures, which may be used for the realization of the adaptive fuzzy systems. Issues such as speed, modularity, the effect of finite precision arithmetic, cost, and software transportability should be addressed.

Multichannel signal processing is a rich and expanding field. Numerous new and advanced areas have appeared which have increased the importance of the tools introduced and analyzed here. Multimedia signal processing [87], visual data processing and analysis [88], multimodal signal processing [89], telecommunications, digital audio restoration, satellite imagery, seismic deconvolution, and biomedicine are only some of the areas in which the methodologies proposed here can be applied.

From a long-term research perspective, there is a need to establish a coherent theoretical foundation for nonlinear filtering algorithms. New algorithms and methodologies which may result in even more effective filtering structures suitable for intelligent processing of multimedia signal processing demand investigation. The framework presented here can serve as an initial point for further research and development in the area and ultimately help in the development of new results and products in the near future.

REFERENCES

[1] I. Pitas and A. N. Venetsanopoulos, *Nonlinear Digital Filters: Principles and Applications*. Norwell, MA: Kluwer, 1990.
[2] ——, "Order statistics in digital image processing," *Proc. IEEE*, vol. 80, pp. 1893–1923, Dec. 1992.
[3] A. Antoniou, *Digital Filters: Analysis and Design*. New York: McGraw-Hill, 1979.
[4] S. Haykin, *Adaptive Filter Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1991.
[5] E. C. Ifeachor and B. W. Jervis, *Digital Signal Processing: A Practical Approach*. New York: Addison-Wesley, 1996.
[6] M. Schetzen, *The Voltera and Wiener Theories of Nonlinear Filters*. New York: Wiley, 1982.
[7] S. Thurnnhofer and S. K. Mitra, "A general framework for quadratic Volterra filters for edge enhancement," *IEEE Trans. Image Processing*, vol. 5, pp. 950–964, June 1996.
[8] A. V. Oppenheim, R. W. Schafer, and T. G. Stockham, "Nonlinear filtering of multiplied and convolved signals," *Proc. IEEE*, vol. 56, pp. 1264–1291, 1968.
[9] J. Astola, P. Haavisto, and Y. Neuvo, "Vector median filter," *Proc. IEEE*, vol. 78, pp. 678–689, Apr. 1990.
[10] M. Gabbouj, E. J. Coyle, and N. C. Gallager, "An overview of median and stack filtering," *Circuits, Syst. Signal Processing*, vol. 11, no. 1, pp. 7–45, Jan. 1992.
[11] J. Serra, *Image Analysis and Mathematical Morphology*. New York: Academic, 1982.
[12] E. R. Dougherty and J. Astola, *An Introduction to Nonlinear Image Processing*. Bellingham: SPIE Press, 1994.
[13] P. Maragos and R. W. Schafer, "Morphological systems for multidimensional signal processing," *Proc. IEEE*, vol. 78, pp. 690–710, Apr. 1990.
[14] P. Maragos, "Differential morphology and image processing," *IEEE Trans. Image Processing*, vol. 5, pp. 922–937, June 1996.
[15] B. Kosko, *Neural Networks for Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1991.
[16] L. Yin, J. Astola, and Y. Neuvo, "A new class of nonlinear filters: Neural filters," *IEEE Trans. Signal Processing*, vol. 41, pp. 1201–1222, Mar. 1993.
[17] S. Haykin, *Neural Networks: A Comprehensive Foundation*. New York: MacMillan, 1994.
[18] S. Haykin, P. Lee, and E. Derbez, "Optimum nonlinear filtering," *IEEE Trans. Signal Processing*, vol. 45, pp. 2774–2786, Nov. 1997.
[19] T. Haweel and P. Clarkson, "A class of order statistic LMS algorithms," *IEEE Trans. Signal Processing*, vol. 40, pp. 44–53, Jan. 1992.
[20] C. Kotropoulos and I. Pitas, "Adaptive LMS L-filters for noise suppression in images," *IEEE Trans. Image Processing*, vol. 5, pp. 1596–1609, Dec. 1996.

[21] J. S. Lee, "Digital image enhancement and noise filtering by local statistics," *IEEE Trans. Pattern Recognition Machine Intell.*, vol. 2, pp. 165–168, Mar. 1980.

[22] H. M. Kim and B. Kosko, "Fuzzy prediction and filtering in impulsive noise," *Fuzzy Sets Syst.*, vol. 77, no. 1, pp. 15–33, Jan. 1996.

[23] F. Russo, "Fuzzy sets in instrumentation: Fuzzy signal processing," *IEEE Trans. Instrum. Meas.*, vol. 45, pp. 683–689, Apr. 1996.

[24] ——, "Nonlinear fuzzy filters: An overview," in *Proc. Signal Processing VIII*, 1996, pp. 1709–1712.

[25] F. Russo and G. Ramponi, "Nonlinear fuzzy operators for image processing," *Signal Processing*, vol. 38, no. 4, pp. 429–440, 1994.

[26] A. Taguchi and T. Kimura, "Data-dependent filtering based on if-then rules and else rules," in *Proc. Signal Processing VIII*, 1996, pp. 1713–1716.

[27] A. Taguchi, T. Kimura, and M. Meguro, "Shaping a noise image by image rules," *SPIE Proc. Nonlinear Image Processing IX*, vol. 3304, 1998, pp. 144–149.

[28] F. Russo, "Nonlinear filtering of noisy images using neuro-fuzzy operators," in *Proc. IEEE Conf. Image Processing*, vol.III, 1997, pp. 412–415.

[29] F. Russo and G. Ramponi, "A fuzzy filter for images corrupted by impulsive noise," *IEEE Signal Processing Lett.*, vol. 3, pp. 168–170, June 1996.

[30] K. Arakawa, "Median filter based on fuzzy rules and its application to image restoration," *Fuzzy Sets Syst.*, vol. 77, no. 1, pp. 3–13, Jan. 1996.

[31] K. Arakawa and Y. Arakawa, "Digital signal processing using fuzzy clustering," *IEICE Trans.*, vol. E 74, no. 11, pp. 3554–3558, 1991.

[32] ——, "Proposal of median-type fuzzy filter and its optimum design," *Electron. Commun. Japan*, part 3, vol. 76, no. 7, pp. 27–35, 1993.

[33] A. Taguchi and N. Izawa, "Fuzzy center weighted median filters," in *Proc. Signal Processing VIII*, 1996, pp. 1721–1724.

[34] K. N. Plataniotis, D. Androutsos, and A. N. Venetsanopoulos, "Fuzzy adaptive filters for multichannel image processing," *Signal Processing J.*, vol. 55, no. 1, pp. 93–106, 1996.

[35] ——, "Multichannel filters for image processing," *Signal Processing: Image Commun.*, vol. 9, no. 2, pp. 143–158, 1997.

[36] X. Yang and P. S. Toh, "Adaptive fuzzy multilevel median filter," *IEEE Trans. Image Processing*, vol. 4, pp. 680–682, May 1995.

[37] S. Peng and L. Lucke, "Fuzzy filtering for mixed noise removal during image processing," *Proc. FUZZ-IEEE'94 Conf.*, Orlando, FL, pp. 83–94.

[38] Y. Choi and R. Krishnapuram, "A robust approach to image enhancement based on fuzzy logic," *IEEE Trans. Image Processing*, vol. 6, pp. 808–825, June 1997.

[39] P. T. Yu and R. C. Chen, "Fuzzy stack filters: Their definitions, fundamental properties and application in image processing," *IEEE Trans. Image Processing*, vol. 5, pp. 838–854, June 1996.

[40] S. Kalluri and G. R. Arce, "Adaptive weighted myriad filter algorithms for robust signal processing in α-stable noise environments," *IEEE Trans. Signal Processing*, vol. 46, pp. 322–334, Feb. 1998.

[41] B. Stuck and B. Kleiner, "A statistical analysis of telephone noise," *Bell Syst. Tech. J.*, vol. 53, pp. 1263–1320, Sept. 1974.

[42] R. J. Stevenson and S. M. Schweizer, "Nonlinear filtering schemes for image smoothing in mixed noise environments," *J. Math. Imaging*, vol. 2, pp. 137–154, 1992.

[43] C. L. Nikias and M. Shao, *Signal Processing with Alpha-Stable Distributions and Applications*. New York: Wiley, 1995.

[44] S. A. Kassam and V. Poor, "Robust techniques for signal processing: A survey," *Proc. IEEE*, vol. 73, pp. 433–481, Mar. 1985.

[45] S. Ambike and D. Hatzinakos, "A new filter for highly impulsive alpha-stable noise," *Proc. 1995 Int. Workshop Nonlinear Signal Processing*, Halkidiki, Greece, 1995, pp. 571–574.

[46] K. N. Plataniotis, D. Androutsos, S. Vinayagamoorthy, A. N. Venetsanopoulos, "An adaptive nearest neighbor multichannel filter," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, pp. 699–703, Dec. 1996.

[47] K. N. Plataniotis, D. Androutsos, and A. N. Venetsanopoulos, "Color image filters: The vector directional appoach," *Opt. Eng.*, vol. 36, no. 9, pp. 2375–2383, 1997.

[48] K. N. Plataniotis, D. Androutsos, S. Vinayagamoorthy, and A. N. Venetsanopoulos, "Color image processing using adaptive multichannel filters," *IEEE Trans. Image Processing*, vol. 6, pp. 933–950, July 1997.

[49] K. N. Plataniotis, D. Androutsos, V. Sri, and A. N. Venetsanopoulos, "A nearest neighbor multichannel filter," *Electron. Lett.*, vol. 31, no. 22, pp. 1910–1911, Oct. 1995.

[50] K. N. Plataniotis, D. Androutsos, and A. N. Venetsanopoulos, "Content-based color image filters," *Electron. Lett.*, vol. 33, no. 3, pp. 202–203, 1997.

[51] B. Kosko, *Neural Networks and Fuzzy Systems: A Dynamic Systems Approach to Machine Intelligence*. Englewood Cliffs, NJ: Prentice-Hall, 1992.

[52] T. Bilgic and I.B. Turksen, "Elicitation of membership functions: How far can theory take us," *Proc. 6th Int. Conf. Fuzzy Systems*, vol. III, 1997, pp. 1321–1325.

[53] R. M. Nosofsky, "Choice, similarity and the context theory of classification," *J. Experimental Psych. Learning, Memory Cognition*, vol. 10, no. 1, pp. 104–114, Jan. 1984.

[54] M. Barni, V. Cappellini, and A. Mecocci, "Fast vector median filter based on Euclidean norm approximation," *IEEE Signal Processing Lett.*, vol. 1, pp. 92–94, June 1994.

[55] J. Chaudhuri, C. A. Murthy, and B. B. Chaudhuri, "A modified metric to compare distances," *Pattern Recognition*, vol. 25, no. 5, pp. 667–677, 1992.

[56] A. D. Gordon, *Classification*. London, U.K.: Chapman & Hall, 1981.

[57] G. N. Lance and W. T. Williams, "Computer programs for hierarchical polythetic classification," *Comput. J.*, vol. 9, no. 1, pp. 60–64, Jan. 1966.

[58] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 1973.

[59] 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.

[60] P. E. Trahanias, D. G. Karakos, and A. N. Venetsanopoulos, "Directional processing of color images: Theory and experimental results," *IEEE Trans. Image Processing*, vol. 5, pp. 868–880, June 1996.

[61] G. Ekman and L. Sjoberg, "Scaling," *Annu. Rev. Psychol.*, vol. 16, pp. 451–474, 1965.

[62] L. Sjoberg, "Similarity and multidimensional ratio estimation with simulataneous qualitative and quantitative variation," *Scandinavian J. Psych.*, vol. 18, pp. 307–316, June 1977.

[63] ——, "Models of similarity and intensity," *Psych. Bulletin*, vol. 82, no. 2, pp. 191–206, 1975.

[64] B. Ekehammar, "A comparative study of some multidimensional vector models for subjective similarity," *Scandinavian J. Psych.*, vol. 13, no. 2, pp. 190–197, 1972.

[65] G. Goude, "A multidimensional scaling approach to the perception of art: I," *Scandinavian J. Psych.*, vol. 13, pp. 258–271, 1972.

[66] A. Tversky, "Features of similarity," *Psych. Rev.*, vol. 84, no. 4, pp. 327–352, 1977.

[67] P. Zysno, "Modeling membership functions," in B. Rieger, Ed., *Empirical Semantics*. Bochum, Germany: Brockmeyer, pp. 350–375, 1981.

[68] H. J. Zimmerman and P. Zysno, "Quantifying vagueness in decision models," *Europ. J. Oper. Res.*, vol. 22, pp. 148–154, 1996.

[69] H. J. Zimmermann and P. Zysno, "Latent connectives in human decision making," *Fuzzy Sets Syst.*, vol. 4, no. 1, pp. 37–51, Jan. 1980.

[70] F. S. Roberts, *Measurement Theory with Applications to Decision-Making, Utility and the Social Sciences*. Reading, MA: Addison-Wesley, 1979.

[71] H. J. Zimmermann, *Fuzzy Sets, Decision Making and Expert System*. Boston, MA: Kluwer, 1987, pp. 235–260.

[72] L. A. Zadeh, "Fuzzy sets," *Inform. Control*, vol. 8, pp. 338–353, 1965.

[73] R. N. Shepard, "Toward a universal law of generalization for psychological science," *Science*, vol. 237, pp. 1317–1323, Sept. 1987.

[74] J. Dombi, "Membership function as an evaluation," *Fuzzy Sets Syst.*, vol. 35, no. 1, pp. 1–21, Jan. 1990.

[75] V. Barnett, "The ordering of multivariate data," *J. Roy. Statistical Soc. A*, vol. 139, pt. 3, no. 1, pp. 318–348, Mar. 1976.

[76] 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, pp. 197–209, June 1991.

[77] I. Pitas and P. Tsakalides, "Multivariate ordering in color image restoration," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 1, pp. 247–260, Sept. 1991.

[78] D. G. Karakos and P. E. Trahanias, "Generalized multichannel image filtering structures," *IEEE Trans. Image Processing*, vol. 6, pp. 1038–1045, July 1997.

[79] G. Wyszecki and W. A. Stiles, *Color Science: Concepts, Methods, Quantiitative Data and Formulae*, 2nd ed. New York: Wiley, 1982.

[80] C. Poynton, *A Technical Introduction to Digital Video*. New York: Wiley, 1996.

[81] G. Sharma and H. J. Trussell, "Digital color imaging," *IEEE Trans. Image Processing*, vol. 6, pp. 901–932, May 1997.

[82] G. Sharma, M. J. Vrhel, and H. J. Trussell, "Color imaging for multimedia," *Proc. IEEE*, vol. 88, pp. 1088–1108, June 1998.

[83] M. Gabbouj and F. A. Cheickh, "Vector median-vector directional hybrid filter for color image restorartion," *Proc. EUSIPCO-96*, 1996, pp. 879–881.

[84] K. Tang, J. Astola, and Y. Neuvo, "Multichannel edge enhancement in color image processing," *IEEE Trans. Circuits and Systems for Video Technology*, vol. 4, pp. 468–479, June 1994.

[85] ——, "Nonlinear multivariate image filtering techniques," *IEEE Trans. Image Processing*, vol. 4, pp. 788–797, June 1995.

[86] P. G. Engeldrum, "A framework for image quality models," *J. Imaging Sci. Technol.*, vol. 39, no. 4, pp. 312–318, 1995.

[87] T. Chen, Guest Ed., "The past, present and future of image and multimedia signal processing," *IEEE Signal Processing Mag.*, vol. 15, pp. 21–58, Feb. 1998.

[88] T. Ebrahimi and M. Kunt "Visual data compression for multimedia applications," *Proc. IEEE*, vol. 86, pp. 1109–1125, June 1998.

[89] S. Y. Kung and J. N. Hwang, "Neural netwrorks for intelligent multimedia processing," *Proc. IEEE*, vol. 86, pp. 1244–1271, June 1998.

[90] D. P. Filev and R. G. Yager, "A generalized defuzzification method via BAD distribution," *Int. J. Intell. Syst.*, vol. 6, pp. 687–697, 1991.

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

[92] K. N. Plataniotis, D. Androutsos, and A. N. Venetsanopoulos, "Color image processing using adaptive vector directional filters," *IEEE Trans. Circuits Syst. II*, vol. 45, pp. 1414–1419, Oct. 1998.

Konstantinos N. Plataniotis (Member, IEEE) received the B.Eng. degree in computer engineering from the University of Patras, Patras, Greece, in 1988 and the M.S and Ph.D degrees in electrical engineering from the Florida Institute of Technology, Melbourne, Florida, in 1992 and 1994, respectively.

He was with the Computer Technology Institute (C.T.I), Patras, Greece, from 1989 to 1991 as a Research Associate. From 1995 to 1997, he was a Postdoctoral Fellow at the Digital Signal and Image Processing Laboratory, Department of Electrical and Computer Engineering, University of Toronto, Toronto, Ont., Canada. From September 1997 to June 1999, he was with Ryerson Polytechnic University as an Assistant Prefessot at the School of Computer Scinece. Currently, he is an Assistant Professor at the Department of Electrical and Computer Engineering, University of Toronto, Ont., Canada. His research interests include multimedia processing, adaptive systems, neural networks, fuzzy systems, and stochastic signal processing.

Dr. Plataniotis is a member of the Technical Chamber of Greece.

Dimitrios Androutsos (Student Member, IEEE) was born in Toronto, Canada, in 1969. He received the B.A.Sc., M.A.Sc., and Ph.D. degrees in electrical engineering from the University of Toronto, Toronto, Ont., Canada, in 1992, 1994, and 1999, respectively.

He is currently with the Digital Signal and Image Processing Lab at the University of Toronto. His research interests lie in color image filtering, adaptive filtering, perceptually lossless compression, signal processing for multimedia applications, and content-based image retrieval.

Anastasios N. Venetsanopoulos (Fellow, IEEE) received the Dipl.Eng. degree from the National Technical University of Athens (NTU), Athens, Greece, in 1965 and the M.S., M.Phil., and Ph.D. degrees in electrical engineering from Yale University, New Haven, CT, in 1966, 1968, and 1969, respectively.

He joined the Department of Electrical and Computer Engineering of the University of Toronto, Toronto, Ont., Canada, in September 1968 as a Lecturer, and he was promoted to Assistant Professor in 1970, Associate Professor in 1973, and Professor in 1981. Since July 1997, he has been the Associate Chair of Graduate Studies in the Department of Electrical and Computer Engineering (ECE) and Acting Chair of the ECE Department during the period January–July 1999. In 1999, he will assume the inagural position of Chair of Multimedia at the University of Toronto. He has also served as Chair of the Communications Group and Associate Chair of the Department of Electrical Engineering. He was on research leave at Imperial College of Science and Technology, the National Technical University of Athens, the Swiss Federal Institute of Technology, the University of Florence, and the Federal University of Rio de Janeiro, and he has also served as Adjunct Professor at Concordia University. He has served as lecturer in 138 short courses to industry, in continuing education programs, and as consultant to numerous organizations. He is a contributor to 25 books, a coauthor of *Nonlinear Filters in Image Processing: Principles and Applications* (Kluwer, 1990), *Artificial Neural Networks: Learning Algorithms, Performance Evaluation and Applications* (Kluwer, 1993), and *Fuzzy Reasoning in Information Decision and Control systems* (Kluwer, 1994), and has published 600 over papers in refereed journals and papers on digital signal and image processing, digital communications, and multimedia.

Dr. Venetsanopoulos has served as Chair on numerous IEEE boards, councils, and technical conference committees, including the Toronto Section (1977–1979) and the IEEE Central Canada Council (1980–1982). He was President of the Canadian Society for Electrical Engineering and Vice President of the Engineering Institute of Canada (EIC) (1983–1986). He was a Guest Editor or Associate Editor for several IEEE journals and the Editor of the *Canadian Electrical Engineering Journal* (1981–1983). He is a member of the IEEE Communications, Circuits and Systems, Computer, and Signal Processing Societies, as well as a member of Sigma Xi , the Technical Chamber of Greece, the European Association of Signal Processing, and the Association of Professional Engineers of Ontario (APEO) and Greece. He is also a Fellow of the EIC, "for contributions to electrical engineering" and was awarded an Honorary Doctorate from the National Technical University of Athens in October 1994. In October 1996, he was awarded the Excellence in Innovation Award of the Information Technology Research Centre of Ontario and Royal Bank "for innovative work in color image processing and its industrial applications."