MORPHOLOGICAL OPERATIONS FOR COLOR IMAGE PROCESSING

Mary L. Comer and Edward J. Delp
Computer Vision and Image Processing Laboratory
School of Electrical Engineering
Purdue University
West Lafayette, Indiana

Corresponding Author:
Professor Edward J. Delp
School of Electrical Engineering
1285 Electrical Engineering Building
Purdue University
West Lafayette, IN 47907-1285
Telephone: (317) 494-1740
Fax: (317) 494-6440
ace@ecn.purdue.edu

Abstract – The use of mathematical morphology in low and mid-level image processing and computer vision applications has allowed the development of a class of techniques for analyzing shape information in monochrome images. In this paper these techniques are extended to color images. We investigate two approaches for "color morphology": a vector approach, in which color vectors are ranked using a multivariate ranking concept known as reduced ordering, and a component-wise approach, in which grayscale morphological operations are applied to each of the three color component images independently. New vector morphological filtering operations are defined, and a set-theoretic analysis of these vector operations is presented. We also present experimental results comparing the performance of the vector approach and the component-wise approach for two applications: multiscale color image analysis and noise suppression in color images.

EDICS: IP 1.7

This work was partially supported by the National Science Foundation, under Grant No. CDR-8803017 to the Engineering Research Center for Intelligent Manufacturing Systems, and a National Science Foundation Graduate Fellowship.

1. INTRODUCTION

Mathematical morphology has been shown to be useful for the processing and analysis of binary and grayscale images [1, 2, 3]. Morphological filtering of an image involves transforming the image into another image using a function or set, known as the structuring element, which acts as a probe sensitive to geometrical information. Geometrical features of the image that are similar in shape and size to the structuring element are preserved, while other features are extracted or suppressed. Morphology has been used to perform noise suppression, texture analysis, shape analysis, edge detection, skeletonization, and multiscale filtering for applications such as medical imaging, geological image processing, automated industrial inspection, image compression, and ECG signal analysis [1, 4, 5, 6, 7, 8, 9, 10, 11]. In this paper we discuss the extension of mathematical morphology to color imagery.

Many techniques developed for use with monochrome images can be extended to color imagery by applying the algorithm to each of the color component images separately. An important question arises: Is component-wise spatial filtering sufficient? Algorithms that exploit spectral correlations could provide better performance and be more computationally efficient. For example, several authors have addressed this issue and proposed linear least-squares and minimum mean-square error restoration algorithms developed specifically for multispectral images [12, 13, 14, 15, 16, 17]. A general framework for linear filtering of multichannel signals in the frequency domain was presented in [18].

Several nonlinear filtering algorithms for multispectral image enhancement have been proposed [19, 20, 21]. These algorithms are based on the concept of ranking multivariate data [22]. In [19] the vector median filter was developed as an extension of the median filter for scalar-valued signals. The vector median filter examines each pixel of a color image as a vector, rather than individual scalar components. The vector median of a collection of vectors is the vector from the collection which has the minimum aggregate distance from all other vectors in the collection [19, 22]. The output of the vector median filter at a given pixel of a color image is the vector median of the color

vectors inside the filter window.

In [20] a class of ranked-order based filters for multichannel images was presented. At each pixel, a region of confidence is determined based on the order statistics of the vectors at neighboring pixels. If the vector at the given pixel does not lie inside this region of confidence, then it is assumed to be an outlier, and it is replaced by the vector in the region of confidence which is at a minimum distance. Thus, these ranked-order vector filters remove "atypical" samples, which makes them suitable for noise suppression in color images.

Vector directional filters for multichannel image processing were introduced in [21]. These filters process vector-valued signals in two steps. First, vectors are processed based on direction, or angle, resulting in the removal of vectors with atypical directions. Then, magnitude processing is performed using any classical grayscale image processing filter.

In this paper we describe new approaches for applying mathematical morphology to color images. We introduce new vector morphological filters which are based on vector ranking concepts [22, 23], and present a set-theoretic analysis relative to these vector operations. We compare the performance of the vector morphological filters and component-wise morphological filters for the applications of multiscale image analysis and noise suppression.

2. MATHEMATICAL MORPHOLOGY

Mathematical morphology is based on set theory. A morphological operation defined on a binary image is referred to as binary morphology. This involves representing the image as a set $X \subseteq \mathbf{R}^2$ or $\mathbf{Z}^2$ (depending on whether the image is defined on a discrete or continuous lattice—an image defined on a discrete lattice will be referred to as a discrete-space image and an image defined on a continuous lattice will be referred to as a continuous-space image), where $\mathbf{R}$ is the set of real numbers and $\mathbf{Z}$ is the set of integers. Points in the image foreground are members of X and points in the background are members of the complement of X, designated X^c. The image is transformed by another set, known as the structuring element. The shape and size of the structuring element

determine the resultant image [1, 2, 3, 24, 25, 26].

There are four basic binary morphological operations: dilation, erosion, opening, and closing, represented by the symbols $\oplus$, $\ominus$, $\circ$, and $\bullet$, respectively. The four operations are defined as follows:

$$X \oplus H = \{(x,y) : H_{(x,y)} \cap X \neq \emptyset\} \quad (1)$$

$$X \ominus H = \{(x,y) : H_{(x,y)} \subseteq X\} \quad (2)$$

$$X \circ H = (X \ominus H) \oplus H \quad (3)$$

$$X \bullet H = (X \oplus H) \ominus H \quad (4)$$

where X is the original image, $H \subseteq \mathbf{R}^2$ or $\mathbf{Z}^2$ is the structuring element, and $H_{(x,y)}$ is the translate of the set H by the vector $(x,y) \in \mathbf{R}^2$ or $\mathbf{Z}^2$.

Grayscale morphological operations are an extension of binary morphological operations to grayscale images. For grayscale operations, the image will be represented by the function $f(x,y)$, where $(x,y) \in \mathbf{R}^2$ or $\mathbf{Z}^2$, or simply f, and the structuring element will be the function $h(x,y)$, or h. Grayscale dilation, erosion, opening, and closing are defined as follows:

$$(f \oplus h)(x,y) = \sup_{(r,s)\in H}\{f(x-r, y-s) + h(r,s)\} \quad (5)$$

$$(f \ominus h)(x,y) = \inf_{(r,s)\in H}\{f(x+r, y+s) - h(r,s)\} \quad (6)$$

$$f \circ h = (f \ominus h) \oplus h \quad (7)$$

$$f \bullet h = (f \oplus h) \ominus h \quad (8)$$

where sup{} and inf{} denote the supremum and infimum operators, respectively, and $H \subseteq \mathbf{R}^2$ or $\mathbf{Z}^2$ is the support of $h(x,y)$. A special class of grayscale morphological filters, referred to as function-and-set-processing (FSP) filters [3], results when $h(x,y) = 0$ for every $(x,y) \in H$. The resulting operations in Equations 5 through 8 are then written as $f \oplus H$, $f \ominus H$, $f \circ H$, and $f \bullet H$.

Alternative definitions and detailed descriptions of the operations defined in this section can be

found in [1, 2, 3, 25, 26].

3. COLOR MORPHOLOGY

The extension of mathematical morphology to color images is not straightforward. In [27] Serra discusses the generalization of morphology to its most basic elements, and concludes that the axioms can be reduced to three key ideas: an order relationship (for example, set inclusion for binary morphology), a supremum or an infimum pertaining to that order, and the possibility of admitting an infinity of operands. The first two of these, the order relationship and the supremum (or infimum), are missing in color images, because there is no unambiguous way to order two or more colors. The fact that these fundamental concepts of morphology do not apply to color images makes it difficult to define "color morphology". However, it is possible that some of the techniques can be extended to color images.

The problem of ordering multivariate data is not unique to mathematical morphology. Although there is no natural means for total ordering of multivariate samples, much work has been done to define concepts such as median, range, and extremes in multivariate analysis. Barnett proposed the classification of these sub-ordering principles into four groups: marginal ordering, reduced ordering, partial ordering, and conditional ordering [22]. The two types of sub-ordering used in this paper are marginal ordering and reduced ordering.

In marginal ordering ranking takes place within one or more of the marginal sets of samples [22], i.e., scalar ranking is performed within each channel. Thus, to order a collection of color vectors using marginal ordering the components in each spectral band are ordered independently of the components in other spectral bands. Morphological operations which are defined using marginal ordering will be referred to in this paper as component-wise operations. The component-wise color dilation of $\mathbf{f}(x,y) = [f_R(x,y), f_G(x,y), f_B(x,y)]^T$ by the structuring element $\mathbf{h}(x,y) =$

$[h_R(x,y), h_G(x,y), h_B(x,y)]^{\mathrm{T}}$ in RGB color space is defined as[1]

$$(\mathbf{f} \oplus_c \mathbf{h})(x,y) = [(f_R \oplus h_R)(x,y), (f_G \oplus h_G)(x,y), (f_B \oplus h_B)(x,y)]^{\mathrm{T}} \tag{9}$$

where the symbol $\oplus_c$ represents component-wise dilation and the dilations on the right-hand side of Equation 9 are grayscale dilations, and the functions $h_R(x,y)$, $h_G(x,y)$, and $h_B(x,y)$ represent structuring elements with regions of support H_R, H_G, and H_B, respectively. Component-wise color erosion, opening, and closing are defined similarly. A component-wise morphological filtering operation can also be implemented in any color space other than RGB.

Because the component images are filtered separately with the component-wise filter, there is a possibility of altering the spectral composition of the image, e. g., the color balance and object boundaries. For example, there is a possibility with this approach that an object could be removed or enhanced in one or two of the R, G, and B components, but not in all of them. This effect near spatial edges in an image is referred to as "edge jitter" [19].

To illustrate this problem, the test image shown on the left in Figure 1 was opened using a component-wise morphological filter with a 15×15 square structuring element.[2] The output of the opening filter is shown on the right in Figure 1. In the opened image the two smallest squares have changed hue. This effect would be unacceptable for many applications. For example, in object recognition color and shape both play important roles. If the component-wise filter is applied to a color image as a step in object recognition, then the effect shown in Figure 1, in which the spectral composition of image objects has been altered, may produce errors in further processing of the color image obtained from the filter.

A different way to examine the problem of color morphology is to treat the color at each pixel as a vector. To motivate this approach, consider a color image with only two colors, representing an object and a background region. Let $\mathbf{f}$ be a color image which consists of the two colors $\mathbf{f}_1 =$

[1] See Appendix A for a discussion of color images and the color space conversions used in this paper.

[2] Due to the prohibitive cost of color printing and the inherent distortions associated with the halftone printing process, only luminance images will be shown in this paper. The corresponding color images will be made available via anonymous ftp to skynet.ecn.purdue.edu (Internet address 128.46.154.48) in the directory /pub/dist/delp/color-morph.

$[R_1, G_1, B_1]^{\mathrm{T}}$ and $\mathbf{f}_2 = [R_2, G_2, B_2]^{\mathrm{T}}$. One way to analyze geometrical features of this image, which is perhaps more natural than the component-wise approach, is to view the pixels which represent the object as a set, and the pixels which represent the background as the complement of this set, as is done with a binary image. The justification for this approach is that both binary images and two-color color images represent scenes with two regions, and the geometrical information in both cases is determined by which pixels belong to the object and which belong to the background. Thus, we define the sets X and X^c as

$$X = \{(x, y) : \mathbf{f}(x, y) = \mathbf{f}_1\} \tag{10}$$

$$X^c = \{(x, y) : \mathbf{f}(x, y) = \mathbf{f}_2\} \tag{11}$$

In this case color dilation, erosion, opening, and closing would be defined the same way as binary dilation, erosion, opening, and closing, with X as the image foreground and X^c as the background. In this vector approach the data at each pixel in the filtered image represent either an object pixel or a background pixel, rather than, for example, the red component from the object and the blue and green components from the background, as is possible with the component-wise approach.

To extend the vector approach to color images with more than two colors, it is necessary to define an order relation which orders the colors as vectors, rather than ordering the individual components. This will be done using reduced ordering. In reduced ordering each multivariate observation is reduced to a single value, which is a function of the component values for that observation, with the multivariate samples ranked according to this single value [22]. To illustrate this type of ordering, let $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ be a collection of multivariate samples, where each $\mathbf{x}_i$ is a vector in $\mathbf{R}^p$. The n samples are to be ordered using a reduced ordering scheme. The first step is to map each $\mathbf{x}_i$ to a scalar value $d_i = d(\mathbf{x}_i)$ where $d : \mathbf{R}^p \to \mathbf{R}$. After d_i has been obtained for each i, the vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_n$ are ordered based on $d_1, d_2, \ldots, d_n$ as follows:

$$\mathbf{x}_{(1)} \leq \mathbf{x}_{(2)} \leq \cdots \leq \mathbf{x}_{(n)} \tag{12}$$

where $\mathbf{x}_{(r)}$ is the vector with corresponding scalar value $d_{(r)}$, and $d_{(r)}$ is the rth smallest element of the set $\{d_1, d_2, \ldots, d_n\}$.

We now use reduced ordering as described above to define new vector morphological filtering operations for color images. The structuring element for the vector morphological operations defined here is the set H, and the scalar-valued function used for the reduced ordering is $d : \mathbf{R}^3 \rightarrow \mathbf{R}$. The operation of vector dilation is represented by the symbol $\oplus_v$. The value of the vector dilation of $\mathbf{f}$ by H at the point (x, y) is defined as

$$(\mathbf{f} \oplus_v H)(x, y) = \mathbf{a} \tag{13}$$

where

$$\mathbf{a} \in \{\mathbf{f}(r, s) : (r, s) \in H_{(x,y)}\} \tag{14}$$

and

$$d(\mathbf{a}) \geq d(\mathbf{f}(r, s)) \; \forall \, (r, s) \in H_{(x,y)} \tag{15}$$

Similarly, vector erosion is represented by the symbol $\ominus_v$, and the value of the vector erosion of $\mathbf{f}$ by H at the point (x, y) is defined as

$$(\mathbf{f} \ominus_v H)(x, y) = \mathbf{b} \tag{16}$$

where

$$\mathbf{b} \in \{\mathbf{f}(r, s) : (r, s) \in H_{(x,y)}\} \tag{17}$$

and

$$d(\mathbf{b}) \leq d(\mathbf{f}(r, s)) \; \forall \, (r, s) \in H_{(x,y)} \tag{18}$$

Vector opening is defined as the cascade of vector erosion and vector dilation, and vector closing is defined as the cascade of vector dilation and vector erosion. With the above definitions for vector morphological operations we must impose the restriction that the set H is a finite set, because if H is not finite then it is possible that no value of $\mathbf{a}$ satisfies Equations 14 and 15 or no value of $\mathbf{b}$ satisfies Equations 17 and 18.

With these definitions the output vector at each point in the image is, by definition, one of the vectors in the original image, so there is no possibility of introducing new color vectors into the image. It is possible that two different color values of $\mathbf{a}$ could satisfy Equations 14 and 15 or two

different color values of $\mathbf{b}$ could satisfy Equations 17 and 18. In this case the output of the vector filter can be chosen based on positions in the structuring element window.

Often the metric used to perform reduced ordering is some type of distance metric [22]. The output of the vector filter will depend not only on the input image and the structuring element, but also on the scalar-valued function used to perform the reduced ordering. For many image processing applications it might make sense to use a characteristic of the human visual system, such as luminance, as a metric for reduced ordering. We investigate different choices for the scalar-valued function in later sections of the paper.

Since mathematical morphology is based on set theory, it is important to investigate the new vector morphological operators defined above in terms of set operations. Serra discusses the need to analyze grayscale morphological operations not only in terms of transformations of grayscale functions, but also in terms of set transformations on the cross sections of those grayscale functions [1]. In fact, grayscale morphology was originally developed by representing functions as sets, using either cross sections or the umbra representation of a function, and applying binary morphological operations to those sets [1]. Similarly, if we define the concept of cross-sections of a color image, then vector morphological operations can be analyzed in terms of set transformations on these cross-sections.

We first review the set-theoretic analysis of FSP morphological operations for grayscale images. For a grayscale image f (discrete-space or continuous-space) the set

$$X_t(f) = \{(x, y) \in D : f(x, y) \geq t\}, t \in V \tag{19}$$

where $V \subseteq \mathbf{R}$ or $\mathbf{Z}$ is the range of the function f, is known as the cross section of f at level t [1, 3]. If the function f is upper semi-continuous then the image can be reconstructed from its cross sections by

$$f(x, y) = \sup\{t \in V : (x, y) \in X_t(f)\} \tag{20}$$

The following equations show the relationship between grayscale dilation and erosion of the

image f by the set H and binary dilation and erosion of the cross sections of f by H [3]:

$$X_t(f \oplus H) = X_t(f) \oplus H \iff (f \oplus H)(x,y) = \sup_{(r,s)\in H_{(x,y)}} \{f(r,s)\} \tag{21}$$

$$X_t(f \ominus H) = X_t(f) \ominus H \iff (f \ominus H)(x,y) = \inf_{(r,s)\in H_{(x,y)}} \{f(r,s)\} \tag{22}$$

Thus, with grayscale dilation and erosion defined as in Equations 5 and 6, the cross section at level t of $f \oplus H$ is equal to the binary dilation of the cross section at level t of f by the set H and the cross section at level t of $f \ominus H$ is equal to the binary erosion of the cross section at level t of f by the set H.

To derive analogous equations relating vector morphological operations to binary morphological operations, we propose the following definition for cross sections of a color image: The set

$$X_t(\mathbf{f}) = \{(x,y) : d(\mathbf{f}(x,y)) \geq t\} \tag{23}$$

is the cross section of the color image $\mathbf{f}$ at level t with respect to the function $d : \mathbf{R}^3 \rightarrow \mathbf{R}$. Reconstruction of a color image $\mathbf{f}$ from its cross sections is possible only if each value of d has a unique color vector associated with it. In this case, reconstruction of $\mathbf{f}$ is given by

$$d(\mathbf{f}(x,y)) = sup\{t \in V : (x,y) \in X_t(\mathbf{f})\} \tag{24}$$

Then, $\mathbf{f}(x,y)$ is the color vector corresponding to $d(\mathbf{f}(x,y))$.

The following proposition relates the vector operations of Equations 13 through 18 to the operations of binary morphology.

Proposition 1 *Subject to the constraints*

$$(\mathbf{f} \oplus_v H)(x,y) \in \{\mathbf{f}(r,s) : (r,s) \in H_{(x,y)}\} \tag{25}$$

and

$$(\mathbf{f} \ominus_v H)(x,y) \in \{\mathbf{f}(r,s) : (r,s) \in H_{(x,y)}\} \tag{26}$$

the following equations provide necessary and sufficient conditions for the vector dilation and

erosion of f to be equivalent to binary dilations and erosions of the cross-sections of f:

$$X_t(\mathbf{f} \oplus_v H) = X_t(\mathbf{f}) \oplus H \iff d((\mathbf{f} \oplus_v H)(x,y)) \geq d(\mathbf{f}(r,s))\ \forall (r,s) \in H_{(x,y)} \tag{27}$$

and

$$X_t(\mathbf{f} \ominus_v H) = X_t(\mathbf{f}) \ominus H \iff d((\mathbf{f} \ominus_v H)(x,y)) \leq d(\mathbf{f}(r,s))\ \forall (r,s) \in H_{(x,y)} \tag{28}$$

Proof. First, we assume that $X_t(\mathbf{f} \oplus_v H) = X_t(\mathbf{f}) \oplus H$ and prove the right-hand side of Equation 27. Using this assumption, we have

$$\{(x,y) : d((\mathbf{f} \oplus_v H)(x,y)) \geq t\} = \{(x,y) : H_{(x,y)} \cap X_t(\mathbf{f}) \neq \emptyset\} \tag{29}$$

$$= \{(x,y) : \exists (r,s) \in H_{(x,y)} \ni d(\mathbf{f}(r,s)) \geq t\} \tag{30}$$

Hence,

$$d((\mathbf{f} \oplus_v H)(x,y)) \geq t \iff \exists (r,s) \in H_{(x,y)} \ni d(\mathbf{f}(r,s)) \geq t \tag{31}$$

Let

$$t_{max} = \max_{(r,s) \in H_{(x,y)}} \{d(\mathbf{f}(r,s))\} \tag{32}$$

Then

$$d((\mathbf{f} \oplus_v H)(x,y)) \geq t_{max} \geq d(\mathbf{f}(r,s))\ \forall (r,s) \in H_{(x,y)} \tag{33}$$

Thus,

$$d((\mathbf{f} \oplus_v H)(x,y)) \geq d(\mathbf{f}(r,s))\ \forall (r,s) \in H_{(x,y)} \tag{34}$$

Now we assume that $d((\mathbf{f} \oplus_v H)(x,y)) \geq d(\mathbf{f}(r,s))\ \forall (r,s) \in H_{(x,y)}$ and prove the left-hand side of Equation 27. Let $(x,y) \in X_t(\mathbf{f}) \oplus H$. Then

$$H_{(x,y)} \cap X_t(\mathbf{f}) = H_{(x,y)} \cap \{(r,s) : d(\mathbf{f}(r,s)) \geq t\} \neq \emptyset \tag{35}$$

which means that $\exists (r,s) \in H_{(x,y)}$ such that $d(\mathbf{f}(r,s)) \geq t$, and hence,

$$d((\mathbf{f} \oplus_v H)(x,y)) \geq t \tag{36}$$

by the assumption that the right-hand side of Equation 27 holds. Thus, $(x,y) \in X_t(\mathbf{f} \oplus_v H)$, and

$$X_t(\mathbf{f}) \oplus H \subseteq X_t(\mathbf{f} \oplus_v H) \tag{37}$$

Now, let $(x, y) \in X_t(\mathbf{f} \oplus_v H)$. Then

$$d((\mathbf{f} \oplus_v H)(x, y)) \geq t \tag{38}$$

which means that

$$\exists (r, s) \in H_{(x,y)} \ni d(\mathbf{f}(r, s)) \geq t \tag{39}$$

because of the constraint of Equation 25. Hence,

$$H_{(x,y)} \cap X_t(\mathbf{f}) \neq \emptyset \tag{40}$$

which implies that

$$(x, y) \in \{(r, s) : H_{(r,s)} \cap X_t(\mathbf{f}) \neq \emptyset\} \tag{41}$$

Thus $(x, y) \in X_t(\mathbf{f}) \oplus H$, and

$$X_t(\mathbf{f} \oplus_v H) \subseteq X_t(\mathbf{f}) \oplus H \tag{42}$$

Next, we assume that $X_t(\mathbf{f} \ominus_v H) = X_t(\mathbf{f}) \ominus H$ and prove the right-hand side of Equation 28. We have

$$\{(x, y) : d((\mathbf{f} \ominus_v H)(x, y)) \geq t\} = \{(x, y) : H_{(x,y)} \subseteq X_t(\mathbf{f})\} \tag{43}$$

$$= \{(x, y) : d(\mathbf{f}(r, s)) \geq t \forall (r, s) \in H_{(x,y)}\} \tag{44}$$

Hence,

$$d((\mathbf{f} \ominus_v H)(x, y)) \geq t \iff d(\mathbf{f}(r, s)) \geq t \forall (r, s) \in H_{(x,y)} \tag{45}$$

Then, for any $t_0 \in \mathbf{R}$,

$$d((\mathbf{f} \ominus_v H)(x, y)) = t_0 \Longrightarrow d(\mathbf{f}(r, s)) \geq t_0 \forall (r, s) \in H_{(x,y)} \tag{46}$$

and, thus,

$$d((\mathbf{f} \ominus_v H)(x, y)) \leq d(\mathbf{f}(r, s)) \, \forall (r, s) \in H_{(x,y)} \tag{47}$$

Now we assume that $d((\mathbf{f} \ominus_v H)(x, y)) \leq d(\mathbf{f}(r, s)) \ \forall (r, s) \in H_{(x,y)}$ and prove the left-hand side of Equation 28. Let $(x, y) \in X_t(\mathbf{f}) \ominus H$. Then $H_{(x,y)} \subseteq X_t(\mathbf{f})$, which implies that for every $(r, s) \in H_{(x,y)}$, $d(\mathbf{f}(r, s)) \geq t$, and hence, using the constraint of Equation 26,

$$d((\mathbf{f} \ominus_v H)(x, y)) \geq t \tag{48}$$

Thus $(x, y) \in X_t(\mathbf{f} \ominus_v H)$. Now, let $(x, y) \in X_t(\mathbf{f} \ominus_v H)$. Then

$$d((\mathbf{f} \ominus_v H)(x, y)) \geq t \tag{49}$$

and hence,

$$\forall (r, s) \in H_{(x,y)} d(\mathbf{f}(r, s)) \geq t \tag{50}$$

which means that

$$H_{(x,y)} \subseteq X_t(\mathbf{f}) \tag{51}$$

and thus, $(x, y) \in X_t(\mathbf{f}) \ominus H$. □

Proposition 1 is important because it provides the following interpretation of the vector morphological operations defined in Equations 13 through 18: Vector dilation of $\mathbf{f}$ by H is equivalent to thresholding $\mathbf{f}$ at each level $t \in \mathbf{R}$ to obtain a set containing all pixels representing objects with d value greater than or equal to t and performing a binary dilation by H on this set, and then recombining the dilated sets to form the output color image.

In the following sections we investigate the use of the component-wise and vector morphological filters for multiscale image analysis and noise suppression. We present experimental results comparing the performance of the two types of filters.

4. MULTISCALE IMAGE ANALYSIS

The representation of image objects at multiple scales is important in many computer vision and image processing applications. A multiscale representation of an image consists of a set of images which are derived by filtering the original image with a family of filters of varying scale, or spatial extent. A multiresolution representation is obtained from a multiscale representation by subsampling the image at each level in the multiscale representation, where images at larger scales are subsampled by a greater factor than images at smaller scales.[3]

Multiscale and multiresolution image representations have become important for several reasons.

[3]This terminology is taken from the computer vision literature [28, 29]. It should be noted that this is different from the terminology used in the signal processing literature [30], in which filtering a signal to change its frequency content affects the resolution of the signal and changing the sampling rate (for a discrete-time signal) affects the scale of the signal.

First, in analyzing an image for computer vision applications, the appearance of an object in the image depends not only on the properties of the object itself, but also on the resolution of the image. Since it is often not possible to define *a priori* an optimal resolution for analyzing images, scale-space methods and multiscale representations have been proposed for computer vision problems, such as stereo matching, object recognition, and image segmentation [31, 32, 33, 34, 29, 35].

Multiresolution image representations are also important because of their usefulness in the area of image coding [36, 37, 38, 10]. In a multiresolution representation the lowest resolution image can be encoded at a low bit rate because it contains a much smaller number of pixels than the original image. Also, at each resolution level the difference between the image at that resolution and the image at the next resolution is obtained, and these difference images can be encoded at reduced bit rates due to their low entropies [36].

A third reason for processing images at multiple resolutions is that this can lead to reduced computational complexity in comparison with algorithms that process images at a single scale [39, 40]. By processing an image first at a low resolution, where there are fewer pixels to process, and then using higher resolution information to refine the solution, significant savings in computational complexity can be obtained.

The type of filtering used to obtain a multiscale representation of an image determines the properties of the resultant multiscale representation (and thus the multiresolution representation, also). Linear filters, such as wavelet filters, quadrature mirror filters, and Gaussian filters, are often used [41, 38, 36]. Multiscale image analysis using morphological filtering has been suggested for applications such as shape-size distributions, image compression, and edge enhancement [4, 28, 29, 10, 9, 42]. Although the linear filtering approach has the advantage that a multiscale representation obtained using linear filtering can be viewed as a space-frequency representation and its frequency content can be studied using Fourier analysis, the morphological filtering approach is more appropriate for quantifying shape information at different scales [28, 29].

In this section we compare the component-wise and vector approaches for obtaining multiscale

representations of color images. We concentrate in particular on multiscale FSP operations. The multiscale FSP opening and closing of a discrete-space grayscale image f by the finite, connected set $H \subseteq \mathbf{Z}^2$ at scale n are defined as [28]

$$f \circ nH = (f \ominus nH) \oplus nH \tag{52}$$

and

$$f \bullet nH = (f \oplus nH) \ominus nH \tag{53}$$

for $n = 0, 1, 2, \ldots$, where

$$nH = H \oplus H \oplus \cdots \oplus H \quad (n-1 \ \textit{dilations}) \tag{54}$$

The multiscale FSP opening and closing of a continuous-space grayscale image f by the compact, connected set $H \subseteq \mathbf{R}^2$ at scale r are defined as

$$f \circ rH = (f \ominus rH) \oplus rH \tag{55}$$

and

$$f \bullet rH = (f \oplus rH) \ominus rH \tag{56}$$

for $r \in \mathbf{R}$, $r \geq 0$, where

$$rH = \{rh : h \in H\} \tag{57}$$

We can perform a multiscale opening or closing on a color image by replacing the grayscale erosions and dilations of Equations 52, 53, 55, and 56 by either component-wise color dilations and erosions or vector dilations and erosions. From the example shown in Figure 1, it is clear that the component-wise filter can have some unexpected effects in terms of spectral filtering. This raises the question of how to include spectral information in a multiscale color image representation. The vector filter has an advantage over the component-wise filter in addressing this issue because it will not alter the spectral composition of image objects. To illustrate the importance of this advantage for multiscale image analysis, Figure 3 shows the component-wise multiscale opening in RGB space of a color image, and Figure 4 shows the vector multiscale opening of the same image, using luminance as the scalar-valued function for the reduced ordering. The structuring element

H used to create these multiscale representations is shown in Figure 2. The vector filter provides a more appropriate multiscale representation of the image than the component-wise filter. The component-wise multiscale representation would be particularly unsatisfactory if the image was to be represented by its multiscale edges [43]. In this case a color edge detector applied at each scale would not produce an accurate representation of the image structure. Although the component-wise filter could be applied in a color space other than RGB, the undesirable effect illustrated in Figure 3 could still occur, since the individual component images would still be filtered independently.

Since the output of the vector filter depends on the scalar-valued function used for reduced ordering, the selection of this function provides flexibility in incorporating spectral information into the multiscale image representation. For example, certain linear combinations of the tristimulus values can be used. This would be written as

$$d(\mathbf{f}(x,y)) = a_R f_R(x,y) + a_G f_G(x,y) + a_B f_B(x,y) \tag{58}$$

if the image is filtered in the RGB color space. For the case $a_R = 0.299$, $a_G = 0.587$, and $a_B = 0.114$, $d(\mathbf{f}(x,y))$ becomes the luminance image. Multiscale opening in this case would suppress bright objects at each scale. A multiscale representation obtained using the luminance image as the scalar-valued function and the structuring element shown in Figure 2 is shown in Figure 5.

The values of a_R, a_G, and a_B can also be selected to enhance or suppress specific colors. For example, if $a_R = 1$, $a_G = 0$, and $a_B = 0$, then the effect of a multiscale opening would be to suppress objects with high red content. Similarly, if $a_R = 0$, $a_G = 1$, and $a_B = 0$ then green objects would be suppressed by a multiscale opening and if $a_R = 0$, $a_G = 0$, and $a_B = 1$ then blue objects would be suppressed. This would lead to a family of images parameterized by shape, size, and "color", which could be useful for an application such as object recognition. Figure 6 shows a multiscale representation (using the same original image and structuring element as those used for Figure 5) obtained using the red image as the scalar-valued function. It can be seen that the difference in the values of a_R, a_G, and a_B used for the two multiscale representations shown in Figures 5 and 6 strongly influences the resultant representation.

5. IMAGE ENHANCEMENT

In this section we investigate the use of color morphology for noise suppression. First, the method used to simulate noisy color images is discussed, and then experimental results using the component-wise and vector filters are presented.

5.1 Simulation of Noisy Color Images

The noise model used here is different from the one used in [20, 44], in which three independent random variables were added to each pixel, one to each of the three spectral components. Our model allows us to vary the levels of noise correlation among the three spectral component images, using a method which is less ad hoc than that used in [21].

There are several reasons that it is important to consider spectrally correlated noise. First, the original image signal (the input to the imaging system) may contain noise which is correlated across the spectral bands. Second, if the three component images are obtained using the same sensor array with different color filters , then at each pixel the three vector components are obtained using the same sensor(s), which could lead to spectrally correlated noise. A third reason to consider spectrally correlated noise is that certain transformations of the image data will result in noise which is spectrally correlated. For example, consider an image in a color space $\mathrm{X_1X_2X_3}$ that has been transformed from another color space $\mathrm{Y_1Y_2Y_3}$. If the image was corrupted with additive noise while in the $\mathrm{Y_1Y_2Y_3}$ space, then at a given pixel the noisy $\mathrm{Y_1Y_2Y_3}$ color vector would be given by $[Y_1 + n_{Y_1}, Y_2 + n_{Y_2}, Y_3 + n_{Y_3}]^{\mathrm{T}}$, where Y_1, Y_2, and Y_3 are the tristimulus values at the given pixel, and n_{Y_1}, n_{Y_2}, and n_{Y_3} are random variables which represent the additive noise at the given pixel. If the transformation matrix from $\mathrm{Y_1Y_2Y_3}$ space to $\mathrm{X_1X_2X_3}$ space has elements a_{ij}, $i = 1, 2, 3$; $j = 1, 2, 3$, then the transformed color vector is

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} Y_1 + n_{Y_1} \\ Y_2 + n_{Y_2} \\ Y_3 + n_{Y_3} \end{bmatrix} = \begin{bmatrix} X_1 + n_{X_1} \\ X_2 + n_{X_2} \\ X_3 + n_{X_3} \end{bmatrix} \tag{59}$$

where X_1, X_2, and X_3 are the $X_1X_2X_3$ tristimulus values at the given pixel for the non-noisy image, and $n_{X_1} = a_{11}n_{Y_1} + a_{12}n_{Y_2} + a_{13}n_{Y_3}$, $n_{X_2} = a_{21}n_{Y_1} + a_{22}n_{Y_2} + a_{23}n_{Y_3}$, and $n_{X_3} = a_{31}n_{Y_1} + a_{32}n_{Y_2} + a_{33}n_{Y_3}$ are random variables which represent the additive noise in the X_1, X_2, and X_3 image planes, respectively. Hence, in this situation, the noise correlation between spectral planes would in general be nonzero.

To simulate noisy color images with spectrally correlated noise, the classical process used to whiten correlated random variables can be reversed [45]. At each pixel, a vector $\mathbf{Z}$, of three uncorrelated, unit-variance, random samples is generated. These samples are then mapped, through a linear transformation, to a vector $\mathbf{N}$ with three samples with variances and correlation coefficients which can be selected as desired. The first step is to specify the desired covariance matrix for the three random variables representing the noise. If the noise is to be added to an image which is in the RGB color space, then this matrix can be written as:

$$\Sigma = \begin{bmatrix} \sigma_R^2 & \rho_{RG}\sigma_R\sigma_G & \rho_{RB}\sigma_R\sigma_B \\ \rho_{RG}\sigma_R\sigma_G & \sigma_G^2 & \rho_{GB}\sigma_G\sigma_B \\ \rho_{RB}\sigma_R\sigma_B & \rho_{GB}\sigma_G\sigma_B & \sigma_B^2 \end{bmatrix} \tag{60}$$

where σ_R^2, σ_G^2, and σ_B^2 are the variances of the red, green, and blue noise components, respectively, and ρ_{RG}, ρ_{RB}, and ρ_{GB} are the spectral correlation coefficients for the red and green, red and blue, and green and blue noise components, respectively. It should be noted that the noise could be added in a color space other than RGB color space.

The transformation from $\mathbf{Z}$ to $\mathbf{N}$, where $\mathbf{N}$ is a three-dimensional random vector with covariance matrix Σ, is needed. To find this transformation, the transformation from $\mathbf{N}$ to $\mathbf{Z}$ is first found. This is the whitening transformation, which converts correlated samples to uncorrelated, unit-variance samples. It has the form

$$\mathbf{Z} = \Lambda^{-\frac{1}{2}}\Phi^{\mathrm{T}}\mathbf{N} \tag{61}$$

where Λ and Φ are the eigenvalue and eigenvector matrices of Σ. This means that $\mathbf{N}$ can be obtained as

$$\mathbf{N} = \Phi^{\mathrm{T}}\Lambda^{\frac{1}{2}}\mathbf{Z} \tag{62}$$

Using the method described above, the spectral correlation of the noise can be specified. To simulate the noise spatially, an ϵ-mixture of Gaussian noise is added to the R, G, and B components of a color image [11]. The probability density function of this noise is given by

$$f_{\mathbf{N}}(\mathbf{n}) = \epsilon \frac{1}{(2\pi)^{3/2}|\Sigma_1|^{1/2}} \exp[-\frac{1}{2}\mathbf{n}^{\mathrm{T}}\Sigma_1^{-1}\mathbf{n}] +$$

$$(1-\epsilon)\frac{1}{(2\pi)^{3/2}|\Sigma_2|^{1/2}} \exp[-\frac{1}{2}\mathbf{n}^{\mathrm{T}}\Sigma_2^{-1}\mathbf{n}], \tag{63}$$

where $\mathbf{N} = [N_1, N_2, N_3]^{\mathrm{T}}$ is the random noise vector, and Σ_1 and Σ_2 are covariance matrices corresponding to impulsive and non-impulsive noise, respectively. Thus, with probability ϵ, a noise vector with covariance matrix Σ_1 is added at a given pixel. The variances $\sigma^2_{R_1}, \sigma^2_{G_1}$, and $\sigma^2_{B_1}$ for this noise are large. This allows spatially impulsive noise to be simulated. With probability $1-\epsilon$, a noise vector with covariance matrix Σ_2 is added. The variances $\sigma^2_{R_2}, \sigma^2_{G_2}$, and $\sigma^2_{B_2}$ for this noise are small. This allows for simulation of non-impulsive noise.

5.2 Experimental Results

This section presents results of experiments involving noise suppression in color images. The grayscale morphological filter used for all of these experiments was the 2DCO filter introduced by Stevenson and Arce [24] and extended by Song and Delp (See Figure 7) [25, 26]. This filter consists of a cascade of two stages each of which consists of multiple morphological operators of one type (opening or closing). In each stage multiple operators are applied to the input image using different structuring elements. The output of each stage is selected according to the morphological operator used: maximum for the output of the opening stage, minimum for the output of the closing stage. If the first stage consists of closing operators and the second stage consists of opening operators, then for a grayscale input image f and a set of structuring elments $\{h_1, h_2, \ldots, h_n\}$, the output y of the first stage is given by

$$y = \min\{f \bullet h_1, f \bullet h_2, \ldots, f \bullet h_n\} \tag{64}$$

The output z of the second stage is given by

$$z = \max\{y \circ h_1, y \circ h_2, \ldots, y \circ h_n\} \tag{65}$$

The purpose of each stage is to preserve the geometrical features in the image that match any one of the given structuring elements. For the experiments presented in this section, four structuring elements consisting of lines with different directions were used [24].

The 2DCO component-wise filter consists of applying the grayscale 2DCO filter to each of the three component images independently. To reduce computational complexity it is also possible to filter only the luminance image in the YIQ space. Hunt and Kubler suggest filtering only the luminance image as a simple approach for deblurring color imagery [12]. For morphological filtering this approach is equivalent to component-wise filtering as in Equation 9 with $H_I = H_Q = \{(0,0)\}$ and $h_I(0,0) = h_Q(0,0) = 0$.

The 2DCO vector filter is defined as a two-stage filter where the output $\mathbf{g}(x,y)$ of the first stage with input $\mathbf{f}(x,y)$ and structuring elements $\{H_1, H_2, \ldots, H_n\}$ is given by

$$\mathbf{g}(x,y) \in \{(\mathbf{f} \bullet_v H_1)(x,y), (\mathbf{f} \bullet_v H_2)(x,y), \ldots, (\mathbf{f} \bullet_v H_n)(x,y)\} \tag{66}$$

and

$$d(\mathbf{g}(x,y)) \leq d((\mathbf{f} \bullet_v H_i)(x,y)) \; \forall i = 1, 2, \ldots, n \tag{67}$$

The output $\mathbf{p}(x,y)$ of the second stage is given by

$$\mathbf{p}(x,y) \in \{(\mathbf{g} \circ_v H_1)(x,y), (\mathbf{g} \circ_v H_2)(x,y), \ldots, (\mathbf{g} \circ_v H_n)(x,y)\} \tag{68}$$

and

$$d(\mathbf{p}(x,y)) \geq d((\mathbf{g} \circ_v H_i)(x,y)) \; \forall i = 1, 2, \ldots, n \tag{69}$$

Noise was added to a color image in RGB space as discussed in the previous section, for two different values of spectral correlation between component images. Figure 8 shows the original image used for the noise suppression experiments. The noisy image shown in the upper left of Figure 9 has parameters $\epsilon = 0.05$; $\sigma_{R_1} = \sigma_{G_1} = \sigma_{B_1} = 100$, for the impulsive noise and $\sigma_{R_2} = \sigma_{G_2} = \sigma_{B_2} = 10$

for the non-impulsive noise; and $\rho_{RG} = \rho_{RB} = \rho_{GB} = 0.95$ for both impulsive and non-impulsive noise. The noise between component planes in this image is highly correlated.

The result of a component-wise 2DCO filter applied in RGB space is shown in Figure 9.[4] Figure 9 also shows the result of filtering the luminance image, Y, and converting the filtered Y and unfiltered I and Q to RGB space for display. This method performs as well as the component-wise filtering in RGB space. Because the noise is highly correlated across spectral bands, the chrominance images, which depend on the spectral composition, were not affected significantly by the noise.

Also shown in Figure 9 is the result of the application of the vector 2DCO filter in RGB space. The Euclidean norm

$$d(\mathbf{f}(x,y)) = [(f_R(x,y))^2 + (f_G(x,y))^2 + (f_B(x,y))^2]^{1/2} \tag{70}$$

was used as the metric for reduced ordering. This filtering method has results similar to the component-wise filtering in RGB space.

The second noisy image used in the experiments is shown in Figure 10. In this image, the values of ϵ and the variances are the same as those used for Figure 8. However, this image has $\rho_{RG} = \rho_{RB} = \rho_{GB} = 0.45$ for both types of noise. Thus, the spectral correlation of the noise is lower.

The result of component-wise filtering in RGB space is shown in the upper right of Figure 10. It can be seen that this filter has the same level of performance for this image as it did for the image of Figure 8.

The lower left image in Figure 10 shows the result of filtering only the luminance and converting to RGB for display. In this case, this method does not perform as well as the component-wise RGB filter. This method is less effective for noise which has lower spectral correlation.

Also shown in Figure 10 is the result of the application of the vector filter to the image with lower

[4] We are not aware of any generally accepted metric for measuring distortion in noisy color images. Therefore, evaluation of our results is subjective.

noise spectral correlation. This method also does not perform as well as the component-wise RGB filter for noise with lower spectral correlation. It is noted in [19] that the vector median filter also does not perform as well as the component-wise median filter when the noise in the different vector components is independent. In general, with spectrally uncorrelated noise the vector approach will not perform as well as the component-wise approach due to the restriction that the output of the vector filter must be one of the input vectors inside the filter window.

6. CONCLUSION

We have developed and compared two approaches to mathematical morphology for color images — the component-wise approach and the vector approach. It was shown that the component-wise filters can alter the spectral composition of image objects, producing undesirable artifacts. Multiscale color image representations illustrating these artifacts were shown. The new vector filters do not have this disadvantage, because all color vectors in the filtered image are selected from the color vectors in the original image. For noise suppression in color images the component-wise approach performed better than the vector approach for noise with low spectral correlation. For noise with high spectral correlation the component-wise and vector approaches had similar performance.

APPENDIX: COLOR IMAGES AND COLOR SPACES

It is well-known that a normal human observer can match any color stimulus by combining light from three primary sources in the correct proportions [46]. Therefore, any color can be represented by three values which correspond to the amounts of the three primaries required to match the color. These three values are known as the tristimulus values. In a digital color image, each pixel has associated with it a vector with three components which represent the tristimulus values of the color at that spatial location in the image. If the tristimulus values of a color image $\mathbf{f}(x, y)$ represent amounts of red, green, and blue primaries, then the image can be modeled as

$$\mathbf{f}(x, y) = [f_R(x, y), f_G(x, y), f_B(x, y)]^{\mathrm{T}} \tag{71}$$

where $f_R(x, y)$ is the red tristimulus value, $f_G(x, y)$ is the green value, and $f_B(x, y)$ is the blue value, at spatial location (x, y).

For many important applications it is more convenient to work in a color space other than RGB space. The YIQ color space is a linear transformation of the RGB space which consists of a luminance signal, Y, and two chrominance signals, I and Q. This is the standard system used for NTSC color television transmission in the United States [47]. The equations used to transform a set of tristimulus values R, G, and B in RGB space to tristimulus values Y, I, and Q in YIQ space are

$$\begin{bmatrix} Y \\ I \\ Q \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ 0.596 & -0.274 & -0.322 \\ 0.211 & -0.523 & 0.312 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \tag{72}$$

The YIQ color space is used for some of the experiments presented in Section 5.

REFERENCES

[1] J. Serra, *Image Analysis and Mathematical Morphology*. New York: Academic Press, 1982.

[2] R. M. Haralick, S. R. Sternberg, and X. Zhuang, "Image analysis using mathematical morphology," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 9, no. 4, pp. 532–550, July 1987.

[3] P. A. Maragos and R. W. Schafer, "Morphological filters–Part I: Their set-theoretic analysis and relations to linear shift-invariant filter," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-35, no. 8, pp. 1153–1169, August 1987.

[4] P. A. Maragos, "Morphological systems for multidimensional signal processing," *Proceedings of the IEEE*, vol. 78, no. 4, pp. 690–709, April 1990.

[5] S. R. Sternberg, "Biomedical image processing," *IEEE Computer Magazine*, pp. 22–34, January 1983.

[6] C. H. G. Wright, E. J. Delp, and N. C. Gallagher, Jr., "Nonlinear target enhancement for the hostile nuclear environment," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, no. 1, pp. 122–145, January 1990.

[7] A. M. Darwish and A. K. Jain, "A rule based approach for visual pattern inspection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 10, pp. 56–68, January 1988.

[8] D. Schonfeld and J. Goutsias, "Optimal morphological pattern restoration from noisy binary images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 1, pp. 14–29, January 1991.

[9] F. Sun and P. Maragos, "Experiments on image compression using morphological pyramids," *Proceedings of the SPIE Visual Communications and Image Processing Conference IV* vol. 1199, November 1989, pp. 1303–1312.

[10] L. A. Overturf, M. L. Comer, and E. J. Delp, "Color image coding using morphological pyramid decomposition," *to appear in IEEE Transactions on Image Processing*, 1995.

[11] C. H. Chu and E. J. Delp, "Impulsive noise suppression and background normalization of electrocardiogram signals using morphological operators," *IEEE Transactions on Biomedical Engineering*, vol. 36, no. 2, pp. 262–273, February 1989.

[12] B. R. Hunt and O. Kubler, "Karhunen-Loeve multispectral image restoration, part 1: Theory," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-32, no. 3, pp. 592–600, June 1984.

[13] N. Ohyama, M. Yachida, E. Badique, J. Tsujiuchi, and T. Honda, "Least squares filter for color image restoration," *Journal of the Optical Society of America A*, vol. 5, no. 1, pp. 19–24, January 1988.

[14] N. P. Galatsanos and R. T. Chin, "Digital restoration of multichannel images," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. ASSP-37, no. 3, pp. 415–421, March 1989.

[15] A. M. Tekalp and G. Pavlovic, "Multichannel image modeling and Kalman filtering for multispectral image restoration," *Signal Processing*, vol. 19, no. 3, pp. 221–232, March 1990.

[16] N. P. Galatsanos, A. Katsaggelos, R. T. Chin, and A. D. Hillery, "Least squares restoration of multichannel images," *IEEE Transactions on Signal Processing*, vol. 39, no. 10, pp. 2222–2236, October 1991.

[17] N. P. Galatsanos and R. T. Chin, "Restoration of color images by multichannel Kalman filtering," *IEEE Transactions on Signal Processing*, vol. 39, no. 10, pp. 2237–2252, October 1991.

[18] A. K. Katsaggelos, K. T. Lay, and N. P. Galatsanos, "A general framework for frequency domain multichannel signal processing," *IEEE Transactions on Image Processing*, vol. 2, no. 3, pp. 417–420, July 1993.

[19] J. Astola, P. Haavisto, and Y. Neuvo, "Vector median filters," *Proceedings of the IEEE*, vol. 78, no. 4, pp. 678–689, April 1990.

[20] R. Hardie and G. Arce, "Ranking in $\mathbf{R}^p$ and its use in multivariate image estimation," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 1, no. 2, pp. 197–209, June 1991.

[21] P. E. Trahanias and A. N. Venetsanopolous, "Vector directional filters–a new class of multichannel image processing filters," *IEEE Transactions on Image Processing*, vol. 2, no. 4, pp. 528–534, October 1993.

[22] V. Barnett, "The ordering of multivariate data," *Journal of the Royal Statistical Society A*, vol. 139, pp. 318–355, 1976.

[23] M. L. Comer and E. J. Delp, "An empirical study of morphological operators in color image enhancement," *Proceedings of the SPIE Conference on Image Processing Algorithms and Techniques III*, vol. 1657, February 1992, San Jose, California, pp. 314–325.

[24] R. L. Stevenson and G. R. Arce, "Morphological filters: Statistics and further syntactic properties," *IEEE Transactions on Circuits and Systems*, vol. 34, no. 11, pp. 1292–1305, 1987.

[25] J. Song and E. J. Delp, "The analysis of morphological filters with multiple structuring elements," *Computer Vision, Graphics, and Image Processing*, vol. 50, pp. 308–328, June 1990.

[26] J. Song and E. J. Delp, "A study of the generalized morphological filter," *Circuits, Systems, and Signal Processing*, vol. 11, no. 1, pp. 229–252, 1991.

[27] J. Serra, *Image Analysis and Mathematical Morphology Volume 2: Theoretical Advances*. London: Academic Press, 1988.

[28] P. A. Maragos, "Pattern spectrum and multiscale shape representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 701–716, July 1989.

[29] M. Chen and P. Yan, "A multiscaling approach based on morphological filtering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 694–700, July 1989.

[30] M. Vetterli and C. Herley, "Wavelets and filter banks: Theory and design," *IEEE Transactions on Signal Processing*, vol. 40, no. 9, pp. 2207–2232, September 1992.

[31] A. Rosenfeld and G. J. Vanderbrug, "Coarse-fine template matching," *IEEE Transactions on Systems, Man, and Cybernetics*, pp. 104–107, February 1977.

[32] A. P. Witkin, "Scale-space filtering," *Proceedings of the International Joint Conference on Artificial Intelligence*, 1983, pp. 1019–1022.

[33] A. L. Yuille and T. A. Poggio, "Scaling theorems for zero-crossings," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, January 1986.

[34] F. Mokhtarian and A. Mackworth, "Scale-based description and recognition of planar curves and 2D shapes," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, January 1986.

[35] C. Bouman and B. Liu, "Multiple resolution segmentation of textured images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 2, pp. 99–113, February 1991.

[36] P. J. Burt and E. H. Adelson, "The Laplacian pyramid as a compact image code," *IEEE Transactions on Communications*, vol. 31, no. 4, pp. 532–540, April 1983.

[37] M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies, "Image coding using wavelet transform," *IEEE Transactions on Image Processing*, vol. 1, no. 2, pp. 205–220, April 1992.

[38] J. W. Woods and S. D. O'Neil, "Subband coding of images," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 34, pp. 1278–1288, October 1986.

[39] P. J. Burt, *The Pyramid As a Structure for Efficient Computation*, pp. 6–35. Berlin: Springer-Verlag, 1984.

[40] D. Terzopoulos, "Image anaysis using multigrid relaxation methods," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, pp. 129–139, March 1986.

[41] S. G. Mallat, "A theory for multiresolution signal decomposition: The wavelet representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 7, pp. 674–693, July 1989.

[42] A. Toet, "A morphological pyramidal image decomposition," *Pattern Recognition Letters*, vol. 9, no. 1, pp. 255–261, May 1989.

[43] S. Mallat and S. Zhong, "Characterization of signals from multiscale edges," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 14, no. 7, pp. 710–732, July 1992.

[44] I. Pitas and P. Tsakalidis, "Multivariate ordering in color image filtering," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 1, no. 3, pp. 247–259, September 1991.

[45] K. Fukunaga, *Introduction to Statistical Pattern Recognition.* Boston: Academic Press, 1990.

[46] D. L. MacAdam, *Color Measurement: Theme and Variations.* Berlin: Springer-Verlag, 1985.

[47] R. W. G. Hunt, *The Reproduction of Colour in Photography, Printing, & Television.* England: Fountain Press, 1987.

Figure 1: Left: original image; right: result of component-wise color opening of original image with 15x15 square structuring element.

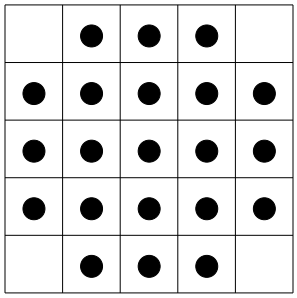

Figure 2: Structuring element used for multiscale smoothing experiments.

Figure 3: Component-wise multiscale opening. Upper left: original image; upper right: result of component-wise opening, $n = 3$; lower left: result of component-wise opening, $n = 4$; lower right: result of component-wise opening, $n = 5$.

Figure 4: Vector multiscale opening with $d(\mathbf{f}(x, y)) = f_Y(x, y)$ (luminance image). Upper left: original image; upper right: result of vector opening, $n = 3$; lower left: result of vector opening, $n = 4$; lower right: result of vector opening, $n = 5$.

Figure 5: Vector multiscale opening with $d(\mathbf{f}(x,y)) = f_Y(x,y)$. Upper left: original image; upper right: result of vector opening, $n = 3$; lower left: result of vector opening, $n = 4$; lower right: result of vector opening, $n = 6$.

Figure 6: Vector multiscale opening with $d(\mathbf{f}(x,y)) = f_R(x,y)$. Upper left: original image; upper right: result of vector opening, $n = 3$; lower left: result of vector opening, $n = 4$; lower right: result of vector opening, $n = 6$.

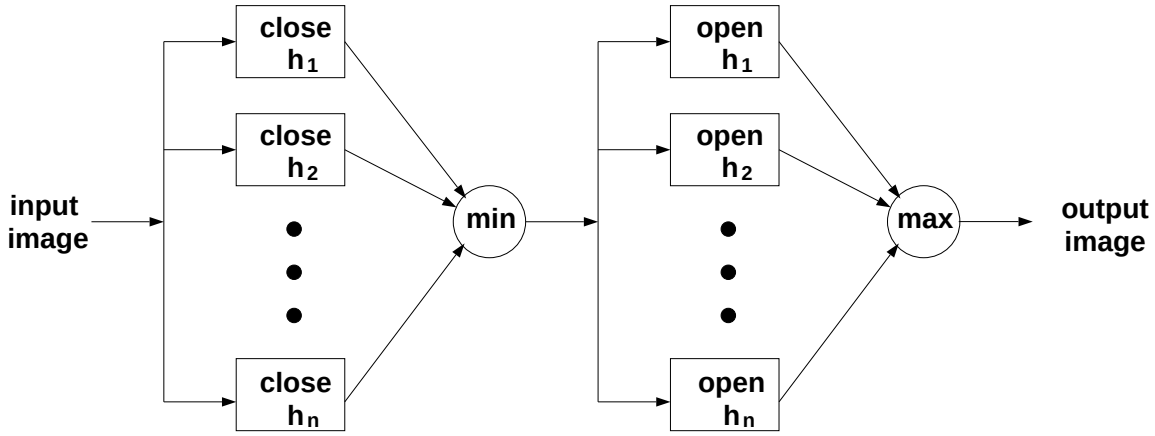

Figure 7: Block diagram of 2DCO filter for grayscale images.

Figure 8: Original image used for noise suppression experiments.

Figure 9: Upper left: noisy version of original image, high spectral correlation; upper right: result of component-wise filter applied in RGB space; lower left: result of filtering only Y in YIQ space; lower right: result of vector filter.

Figure 10: Upper left: noisy version of original image, low spectral correlation; upper right: result of component-wise filter applied in RGB space; lower left: result of filtering only Y in YIQ space; lower right: result of vector filter.