ELSEVIER

Pattern Recognition Letters 17 (1996) 481–490

Pattern Recognition Letters

Reduction of color space dimensionality by moment-preserving thresholding and its application for edge detection in color images [1]

Chen-Kuei Yang [a,b], Wen-Hsiang Tsai [a,*]

[a] *Department of Computer and Information Science, National Chiao Tung University, HsinChu, Taiwan 300, ROC*
[b] *Department of Information Management, Ming Chuan College, Taipei, Taiwan 111, ROC*

Received 17 May 1995; revised 22 November 1995

Abstract

A method for reduction of color space dimensionality by moment-preserving thresholding and its application in edge detection for color images is proposed. An input color image is partitioned into $n \times n$ non-overlapping blocks. A moment-preserving thresholding technique is then applied individually to each color plane of each image block. Two sets of (R, G, B) tristimulus values are obtained from the thresholding results to form two representative color vectors for each block. The difference vector between these two representative color vectors is used as an axis onto which all the data in the block are projected to reduce the color space to one dimension. A single-spectral image block is so obtained. Due to the use of analytic formulas in the thresholding step, the proposed dimensionality reduction method is found faster than the KL expansion or vector median approaches which are also applicable for dimensionality reduction. An $(n+1) \times (n+1)$ circular window is selected to sample the resulting single-spectral image, which in turn includes the $n \times n$ square block. Some mass moments of the window data are computed and used for edge detection in the circular window. Due to the use of the larger detection window which results in smaller overlapping detection areas, the computation time for the edge detection step is reduced, compared with other similar approaches using overlapping detection windows. Experimental results show that the proposed approach is effective in reduction of color space dimensionality and edge detection in color images.

Keywords: Dimension reduction; Color edges; Moment-preserving

1. Introduction

In color image processing, the color of a pixel is usually given as three values corresponding to the tristimuli of red (R), green (G), and blue (B). Many different coordinate systems have been employed for the specification of color as discussed in (Pratt, 1991). Typically, the tristimulus values of (R, G, B) are highly correlated with one another (Pratt, 1971). In the development of efficient quantization, coding, and processing techniques for color images, it is often desirable to work with components that are uncorrelated. A lot of literature (Pratt, 1991) has been developed to transform the color coordinate system for different purposes of color image process-

* Corresponding author. Email: whtsai@cis.nctu.edu.tw

[1] This work was supported partially by National Science Council, Republic of China under Grant NSC83-0408-E009-010.

SSDI 0167-8655(95)00112-3

ing. But they still use a set of tristimulus values to specify a color. For color image processing, such approaches usually apply grey-scale image processing techniques individually to each spectral band according to the characteristics of the spectral band, and then merge the processing results in some ways. The computation is slow because color model transformation is performed and the three spectral bands are individually treated. In this study, a method for reduction of color space dimensionality is proposed. Only one spectral band is preserved. So, most existing grey-scale image techniques can be applied to color images without using the merge operation.

Edge detection is one of the basic techniques for image segmentation and compression, and various techniques for edge detection have been proposed (Pratt, 1991; Gonzalez and Wintz, 1987). Surveys of literature on edge detection can be found in (Davis, 1975; Fu and Mui, 1981; Peli and Malah, 1982; Di Zenzo, 1983). Edges are defined as a discontinuity in some image attributes, usually the brightness for grey-scale images. For color images, the situation is different. Several definitions of color edges have been proposed (Pratt, 1991). A color edge can be said to exist if and only if the luminance field contains an edge. This definition ignores discontinuities in hue and saturation that occur in regions of constant luminance. Another way to define a color edge is to check if an edge exists in any of its constituent tristimulus components. A third definition is based on forming the sum of gradients of the tristimulus values or some linear or nonlinear color components. A color edge is said to exist if the gradient exceeds a threshold.

Most existing edge detection techniques for grey-scale images can be applied to color images with some merge operations on the detected results of the three spectral bands (Pratt, 1991). However, this needs a lot of computation time because they process pixels one by one and treat three spectral bands separately. Nevatia (1977) extended the Hueckel edge operator (Hueckel, 1971) to color images. He computed the edges in three chosen color components separately and merged them by a certain procedure. Di Zenzo (1986) extended the gradient-based edge detection technique to multispectral images and combined the three color component results by taking the RMS (root mean square), or the sum, or the maximum of their absolute values. Shiozaki (1986) developed an entropy operator using entropy of brightness in a local region of a picture, applied it to each color component, and merge the three values of the detected entropy for color edge detection. Machuca and Phillips (1983) treated color images as vector fields and transformed the color space from the RGB model into the YIQ model and detected edges in the hue plane. Here, additional computation time is required for color model transformation. Trahanias and Venetsanopoulos (1993) also treated color images as vector fields and a class of color edge detectors is defined using the magnitudes of linear combinations of the sorted vector samples. The operation of the essential vector order statistics on detecting local minima and maxima in the vector ordering is required. They combined them (the detected local minima and maxima) in a suitable way in order to produce a positive response for an edge pixel.

In this study, a method for reduction of color space dimensionality and its application to color edge detection are proposed. The proposed dimensionality reduction method is based on the moment-preserving technique originally proposed for block truncation coding (Mitchell et al., 1978). The method has two major steps as described in the following. First, an input image is partitioned into $n \times n$ non-overlapping square blocks. Then, a moment-preserving thresholding technique (Tsai, 1985) is applied to each color plane of each square block. The color values of each color plane are thresholded to two levels and a representative value is obtained for each level. Accordingly, two representative color vectors (R_1, G_1, B_1) and (R_2, G_2, B_2) are obtained for each block. The difference vector between these two color vectors $(R_1 - R_2, G_1 - G_2, B_1 - B_2)$ is computed to define an axis. All the data in the block are projected onto this axis. The three-spectral color image is so dimensionally reduced into a one-spectral image. Second, for each $n \times n$ block, the mass moment-preserving edge detection technique (Tabatabai and Mitchell, 1984) is used to locate an edge to subpixel accuracy in a given circular window within an $(n + 1) \times (n + 1)$ block, which in turn includes an $n \times n$ block. Alternatively, most existing edge detection techniques for grey-scale images can also be applied here if preferred. Besides, the information of the two representative color vectors (R_1, G_1, B_1) and

(R_2, G_2, B_2) and the edge location can be utilized further for color image compression.

In what follows, a brief review of moment-preserving thresholding (Tsai, 1985) and edge detection by mass moment preserving (Tabatabai and Mitchell, 1984) is first given in Section 2. In Section 3, the proposed approach to reduction of color space dimensionality and its application in color edge detection is described. Several experimental results to show the feasibility of the proposed method are presented in Section 4. Finally, some conclusions are made in Section 5.

2. Review of moment-preserving thresholding and subpixel edge detection

2.1. Moment-preserving thresholding

The thresholding operation is a necessary step in many image analysis applications. Surveys of literature on thresholding techniques can be found in (Rosenfeld and Kak, 1982; Weszka, 1978). The moment-preserving principle has been employed to perform many image processing works (Mitchell et al., 1978; Tsai, 1985; Tabatabai and Mitchell, 1984; Delp and Mitchell, 1991; Lyvers et al., 1989; Yang et al., 1994). A survey on moment-preserving techniques and applications can be found in (Chen and Tsai, 1989). Now, the moment-preserving thresholding algorithm developed by Tsai (1985) is reviewed below.

Given a grey-scale image f with N pixels whose grey-value at pixel (x, y) is $f(x, y)$, the ith grey-moment of f is

$$m_i = \frac{1}{N} \sum_x \sum_y f^i(x, y), \quad i = 0, 1, 2, \ldots. \tag{1}$$

Let O be an operator applied to the input image f, and F be the output image. If the ith moment of F is set equal to that of f, then O is said to preserve the ith moment of the input data in the output data. By definition, we see that if an operator preserves more moments, then more information of the input image will be retained in the output.

Suppose that it is desired to threshold f into two levels h_1 and h_2 with p_1 and p_2 being the fractions of pixels with h_1 and h_2, respectively. The algorithm by Tsai (1985) selects a threshold value in such a way that if all the pixels with below-threshold grey-value in f are replaced by grey-value h_1 and all the pixels with above-threshold grey-value in f are replaced by grey-value h_2, then the first three moments of image f are preserved in the resulting bi-level image F. The concept can be expressed as follows:

$$\begin{aligned} &p_1 h_1 + p_2 h_2 = m_1, \qquad && p_1 h_1^2 + p_2 h_2^2 = m_2, \\ &p_1 h_1^3 + p_2 h_2^3 = m_3, \qquad && p_1 + p_2 = 1, \end{aligned} \tag{2}$$

where the left term in each equality of (2) is a moment of F and the corresponding right term is a moment of f. To find the desired threshold value t, the equalities of (2) are first solved to obtained p_1, p_2, h_1, and h_2. Then, t is chosen as the grey-value closest to the p_1-tile of the histogram of f. When t, h_1, and h_2 have been computed, f can be bi-levelly thresholded into F as follows:

$$F(x, y) = \begin{cases} h_1 & \text{if } f(x, y) < t, \\ h_2 & \text{if } f(x, y) \geqslant t. \end{cases} \tag{3}$$

In more general cases, it is desired to threshold a given image f into more than two pixel classes. To threshold f into M pixel classes, the first $2M - 1$ moments are preserved to get the solution (Tsai, 1985). The threshold values can be computed deterministically and so the threshold selection is made automatically. This is very convenient for practical applications.

2.2. Subpixel edge detection

The subpixel edge detector proposed by Tabatabai and Mitchell (1984) is based on the mass moment-preserving principle and is reviewed here. Edge locations are described by line equations. The edge operator accepts as input a set of grid squares consisting of 69 pixels, arranged so as to best approximate the area of a unit circle (see Fig. 1(a)). The edge operator generates an ideal edge element defined over a unit circle with two grey-values, h_1 and h_2. The edge element includes a border line that separates the two intensity levels as shown in Fig. 1(b) and Fig. 2. Based on the moment-preserving principle, the moments of output image F are

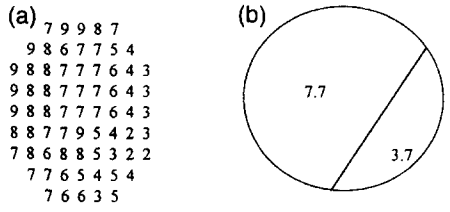

Fig. 1. Tabatabai–Mitchell edge detection on a unit circle. (a) Empirically obtained edge element. (b) Ideal edge element after edge detection.

then derived in terms of the desired parameters as well as some geometric relations and other information. By enforcing the moments in F to equal the original moments in f, a set of equalities are obtained. Finally, the equalities are solved to get the values of the desired parameters, from which the output image can be reproduced. Note that such an output image preserves the moment information of the input image.

A more exact definition of the output disk is obtained if the ideal edge element is described by the function $U(x, y, \alpha, \rho, h_1, h_2)$ as follows:

$$\begin{aligned} &U(x, y, \alpha, \rho, h_1, h_2) = h_1 \\ &\text{if } x\cos\alpha + y\sin\alpha \leqslant \rho, \\ &U(x, y, \alpha, \rho, h_1, h_2) = h_2 \\ &\text{if } x\cos\alpha + y\sin\alpha > \rho. \end{aligned} \tag{4}$$

To find the value of α, let $(\bar{x}, \bar{y})$ be the coordinates of the center of gravity of the grey-values inside the circle. It can be found that the direction of the edge

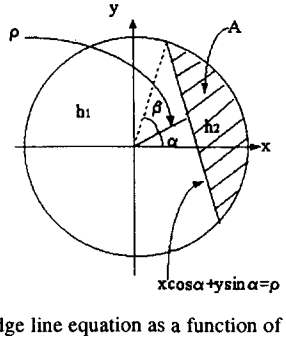

Fig. 2. Edge line equation as a function of α and ρ.

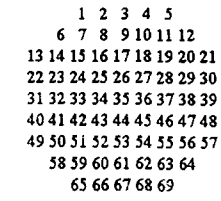

Fig. 3. Indexing associated with each square grid.

is perpendicular to the direction of the vector from the origin to $(\bar{x}, \bar{y})$. So, angle α can be calculated as

$$\sin\alpha = \frac{\bar{y}}{\sqrt{\bar{x}^2 + \bar{y}^2}}, \tag{5}$$

$$\cos\alpha = \frac{\bar{x}}{\sqrt{\bar{x}^2 + \bar{y}^2}}. \tag{6}$$

Let the grids of the unit circle be indexed as shown in Fig. 3. Then $(\bar{x}, \bar{y})$ used in Eqs. (5) and (6) above can be estimated as follows:

$$\bar{x} = \left(\sum_{j=1}^{69} x_j I_j w_j\right) \Big/ \left(\sum_{j=1}^{69} I_j w_j\right), \tag{7}$$

$$\bar{y} = \left(\sum_{j=1}^{69} y_j I_j w_j\right) \Big/ \left(\sum_{j=1}^{69} I_j w_j\right), \tag{8}$$

where

I_j = the intensity associated with the jth grid,

w_j = the weight associated with the jth grid,

(x_j, y_j) = the coordinates of the center of the jth grid,

$$w_2 = w_4 = w_{22} = w_{40} = w_{68} = w_{66} = w_{48} = w_{30} = 0.013782918,$$

$$w_3 = w_{31} = w_{39} = w_{67} = 0.015573185,$$

$$w_6 = w_{58} = w_{64} = w_{12} = 0.013068037,$$

and the remaining weighting coefficients are assigned a value of "0.015719006".

Angle α thus can be computed in terms of $(\bar{x}, \bar{y})$ above according to Eqs. (5) and (6).

To find ρ, h_1, and h_2, first define the first three sample moments of the empirically obtained data in the unit circle as

$$m_i = \sum_{j=1}^{69} w_j I_j^i, \quad i = 1, 2, 3. \tag{9}$$

By preserving the first three sample moments of the input in the output of the operator, the values of p_1, p_2, h_1, and h_2 can be obtained from Eqs. (2).

Given a circle of radius unity and an arbitrary angle $0 \leqslant \beta \leqslant \pi/2$, the area A shown in Fig. 2 is given by

$$A = \beta - \tfrac{1}{2}\sin 2\beta. \tag{10}$$

If we let $p = \min(p_1, p_2)$, which denotes exactly the fraction of area A with respect to the total circle area π, then we get

$$\beta - \tfrac{1}{2}\sin 2\beta = \pi p. \tag{11}$$

Eq. (11) is a transcendental equation, and one can use some numerical approximations to obtain β. Once β is obtained, then

$$\rho = \cos \beta. \tag{12}$$

With α and ρ being both computed, the edge line equation is determined, which can be used to locate the edge to subpixel accuracy. The criteria for acceptance of a pattern as an edge was based on

$$|h_1 - h_2| \geqslant 2\sigma \tag{13}$$

where σ^2 is the variance of the observed data. If condition (13) is satisfied, then an edge pattern is determined to be present in the circular window.

3. Proposed approach to reduction of color space dimensionality and its application in color edge detection

3.1. Proposed reduction of color space dimensionality

With regard to the color edge definitions which are described previously, in many image blocks the three color components are weakly correlated and the edges in different color bands may be different from one another. Taking the two-spectral image shown in Fig. 4 (with R and G planes only) as an illustrative example, we can see that the edge is not obvious on either of the two color planes but can be found more easily on a combination of them. There exist some ways to find out the color edge in such practical cases. The idea is to find first an axis onto which all color component values are projected to produce a new image with a single spectral band, and then find the edge on the new image. By this way of reducing the 3-dimensional color space to one dimension, edge detection methods for grey-scale images become applicable to color images. Three of the color space dimensionality reduction approaches investigated in this study are the Karhunen–Loève (KL) expansion (Fukunaga, 1990), the vector median (Astola et al., 1990), and the proposed method.

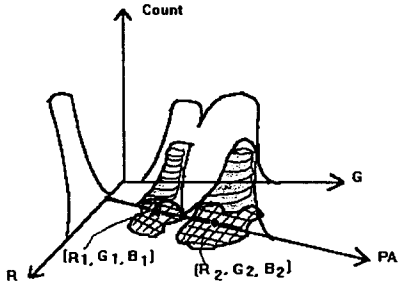

Fig. 4. Data distribution of a two-spectral image; the shaded regions are the original data, and PA represents an axis on which an edge can be found more easily.

From the theory of feature extraction, the KL expansion can be employed to perform feature space dimensionality reduction (Bigun, 1993), and the axis found is called the principal axis on which the features have the largest variance values. For each image block of the *RGB* color space, there exists a covariance matrix Σ, defined as follows:

$$\Sigma = \begin{bmatrix} E(R^2) - E^2(R) & E(RG) - E(R)E(G) & E(RB) - E(R)E(B) \\ E(RG) - E(R)E(G) & E(G^2) - E^2(G) & E(GB) - E(G)E(B) \\ E(RB) - E(R)E(B) & E(GB) - E(G)E(B) & E(B^2) - E^2(B) \end{bmatrix}. \tag{14}$$

Σ has three non-negative eigenvalues because it is a covariance matrix (Noble and Daniel, 1988). The greater the eigenvalue is, the more information is contained in the corresponding eigenvector. The eigenvector associated with the largest eigenvalue is the principal axis.

Projection on a vector median is another approach to reducing the color space dimensionality. The median operation in signal processing is a nonlinear filtering operation where a window moves over a signal, and at each point the median value of the data within the window is taken as the output. The vector median has properties similar to those of the standard median in the scalar case. According to Astola

et al. (1990), a vector median can be found as follows.

(1) For each vector x_i, compute the distance to all the other vectors using either the L_1 (absolute distance) or L_2 (Euclidean distance) norm and add them together, resulting in

$$S_i = \sum_{j=1}^{n} \| x_i - x_j \|, \quad i = 1, \ldots, n, \tag{15}$$

where n is the number of pixels in each block.

(2) Find the value $x_{\min}$ among all x_i such that S_i is the minimum of all S_i.

Once the vector median of a block is determined, all the data of a block are projected on the vector median of the block.

In the above two approaches, a lot of computation time is required when the eigenvalues and eigenvectors are computed or a vector median is selected. Due to this reason, another way based on the moment-preserving principle to find the desired axis is proposed in this study.

First, the bi-level thresholding method proposed by Tsai (1985), reviewed previously, is applied individually to each color plane of each color image block. Two sets of representative color vectors (R_1, G_1, B_1) and (R_2, G_2, B_2) are obtained. More specifically, R_1 and R_2, for example, correspond to h_1 and h_2 in (3), respectively. Because the two color vectors (R_1, G_1, B_1) and (R_2, G_2, B_2) can be regarded as the centroids of the two clusters in the *RGB* feature space on the two sides of the color edge, the difference vector $(R_1 - R_2, G_1 - G_2, B_1 - B_2)$ is meaningful. This direction is approximately perpendicular to the color edge, and so can be used as the desired axis for data projection mentioned previously. Since the computation involved in finding (R_1, G_1, B_1) and (R_2, G_2, B_2) uses analytic formulas, the computation time is reduced considerably, compared with that for computing the principal axis using the KL expansion or selecting a vector median. All the three methods are implemented and their results are compared with one another in Section 4.

3.2. Application in color edge detection

After finding the axis specified by the difference vector $(R_1 - R_2, G_1 - G_2, B_1 - B_2)$, we can project all the data in the image block onto the direction of this axis. The projected data behave just like a grey-scale image. Then, the Tabatabai–Mitchell (1984) edge detector is applied to the projected data to perform edge detection.

The criterion for acceptance of a pattern as a color edge is described as follows:

$$|R_1 - R_2| + |G_1 - G_2| + |B_1 - B_2| \geqslant \tau(\sigma_r + \sigma_g + \sigma_b), \tag{16}$$

where τ is a preset threshold value and σ_r, σ_g, and σ_b are the standard deviations of the three color planes of all image blocks, respectively. According to (Tabatabai and Mitchell, 1984), the lower bound value for τ is 2 and in this study 3.5 is selected experimentally.

3.3. Algorithm

An algorithm is given below to summarize the proposed method.

Algorithm (Reduction of color space dimensionality and its application in color edge detection)

Step 1. Read in a color image f.

Step 2. Partition f into $n \times n$ non-overlapping square blocks.

Step 3. For each block f_j, perform the following stages.

Stage 3.1. *Dimensionality reduction*

(a) Apply the bi-level thresholding operation of Tsai (1985) to each color plane of f_j individually to compute two representative color vectors (R_1, G_1, B_1) and (R_2, G_2, B_2).

(b) Compute the absolute difference vector of the two color vectors $\boldsymbol{d} = |\boldsymbol{x}_1 - \boldsymbol{x}_2|$, where $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are two representative color vectors (R_1, G_1, B_1) and (R_2, G_2, B_2), respectively.

(c) Normalize the difference vector, resulting in $\hat{\boldsymbol{d}} = \boldsymbol{d}/\|\boldsymbol{d}\|$ where $\|\boldsymbol{d}\|$ represents a $\sqrt{r^2 + g^2 + b^2}$ norm, making the found color coordinates consistent from block to block.

(d) Project all the pixels of f_j onto $\hat{\boldsymbol{d}}$ by computing $(r_i, g_i, b_i) \times \hat{\boldsymbol{d}}$ as the result.

(e) If $|R_1 - R_2| + |G_1 - G_2| + |B_1 - B_2| \geqslant \tau(\sigma_r + \sigma_g + \sigma_b)$, then continue to perform edge detection; otherwise skip the next stage and process the next block.

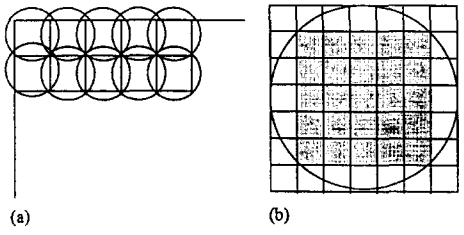

Fig. 5. Circular windows with $(n+1)\times(n+1)$ block size. (a) Illustration of circular window overlapping. (b) The shadow area represents the edge detection region.

Stage 3.2. *Edge detection*

(f) Sample the data of the projected result in an $(n+1)\times(n+1)$ circular window for edge detection which includes the $n\times n$ square block f_j in the center (see Fig. 5).

(g) Apply the mass-moment edge detector of (Tabatabai and Mitchell, 1984) to f_j to obtain an edge described by Eq. (4).

Step 4. Stop.

In Step 3(f), the benefit of sampling the projected result within an $(n+1)\times(n+1)$ circular window is that the number of window shifts for processing an image can be reduced, compared with (Tabatabai and Mitchell, 1984) (see Fig. 5) because the overlap area formed by the circles is smaller than that in (Tabatabai and Mitchell, 1984).

In Step 3(g), a more convenient way to compute $\bar{x}$ and $\bar{y}$ (mentioned in (7) and (8)) is to correlate the pixel grey-values in the window with a mask (Lyvers et al., 1989). Each mask represents the product of x_j and w_j. This is a good technique to save the computation time when $\bar{x}$ and $\bar{y}$ are computed. The corresponding set of masks is shown in Fig. 6.

-.00000	-.01497	-.01902	.00000	.01902	.01497	.00000
-.02235	-.04664	-.02332	.00000	.02332	.04664	.02235
-.05734	-.04664	-.02332	.00000	.02332	.04664	.05734
-.06900	-.04664	-.02332	.00000	.02332	.04664	.06900
-.05734	-.04664	-.02332	.00000	.02332	.04664	.05734
-.02335	-.04664	-.02332	.00000	.02332	.04664	.05734
-.00000	-.01497	-.01902	.00000	.01902	.01497	.00000

(a) Mask for M_X

.00000	.02235	.05734	.06900	.05734	.02235	.00000
.01497	.04664	.04664	.04664	.04664	.04664	.01497
.01902	.02332	.02332	.02332	.02332	.02332	.01902
.00000	.00000	.00000	.00000	.00000	.00000	.00000
-.01902	-.02332	-.02332	-.02332	-.02332	-.02332	-.01902
-.01497	-.04664	-.04664	-.04664	-.04664	-.04664	-.01497
-.00000	-.02235	-.05734	-.06900	-.05734	-.02235	-.00000

(b) Mask for M_Y

.00000	.02860	.06861	.08066	.06861	.02860	.00000
.02860	.08163	.08163	.08613	.08613	.08613	.02860
.06861	.08163	.08163	.08613	.08613	.08613	.06861
.08066	.08163	.08163	.08613	.08613	.08613	.08066
.06861	.08163	.08163	.08613	.08613	.08613	.06861
.02860	.08163	.08163	.08613	.08613	.08613	.02860
.00000	.02860	.06861	.08066	.06861	.02860	.00000

(c) Mask for M_R

Fig. 6. The set of three masks for the moment computation for window size 7×7.

4. Experimental results

The proposed algorithm has been tested on an IRIS Indigo workstation. Experiments using many different kinds of images have been conducted. Two of the tested images "house" and "pepper" are shown in Figs. 7(a) and 7(b), respectively. Each

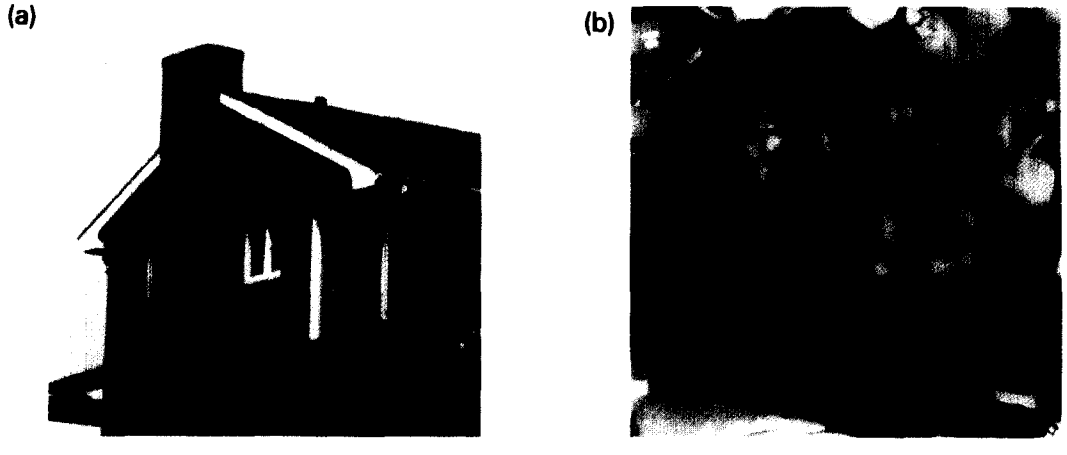

Fig. 7. The original images. (a) House. (b) Pepper.

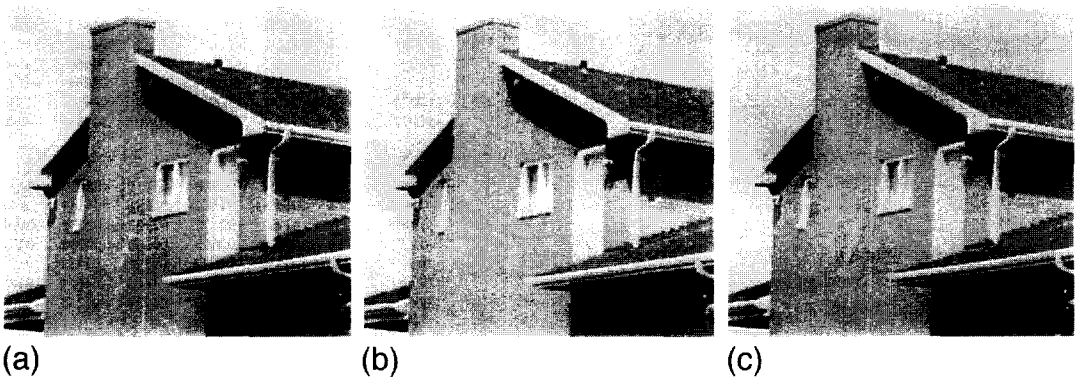

Fig. 8. The projection results of "house". (a) Result using the KL expansion. (b) Result using the vector median. (c) Result using the proposed method.

color image has 24 bits per pixel and is 512×512 in size. The circular window sizes tested are 5×5, 6×6, and 7×7.

Table 1 shows the comparative CPU times of using the KL expansion, the vector median approach, and the proposed method to find the desired axis for data projection. It can be seen that the larger the circular window is chosen, the more computation time is saved. Figs. 8 and 9 show the projection results of using the KL expansion, the vector median approach, and the proposed methods in (a), (b), and (c), respectively. From Figs. 8 and 9, we can see that the single-spectral images resulting from data projection using the three methods are almost the same, and the image contents are preserved in the projection results. It is difficult to find obvious differences among the three results by visual inspection. However, a little difference can be distinguished by computing error images. Two error images computed from the differences between the projection results of the images " house" and "pepper" using the KL expansion with respect to the proposed method are shown in Fig. 10. It can be seen that most of the image "house" consists of uniform regions, and the error image of the projection result is thus almost black. This means that there is almost no difference between the two projection results. The proposed method can also yield good results even when the image has regions with gradual changes in color. The projection results will only have some different values in brightness in regions with gradual color changes. The edge detection results for images

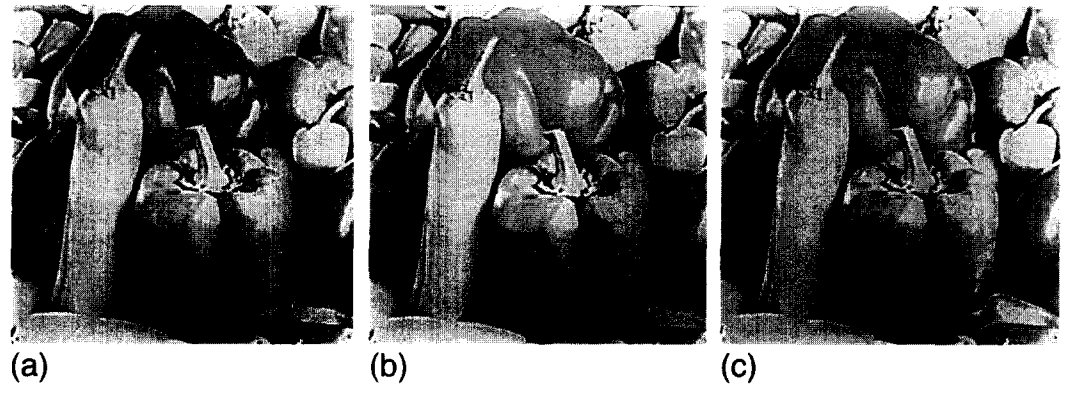

Fig. 9. The projection results of "pepper". (a) Result using the KL expansion. (b) Result using the vector median. (c) Result using the proposed method.

Table 1
The comparative CPU times of the color space dimensionality reduction using the KL expansion, vector median, and the proposed method for processing the "pepper" image

Circular window size	The KL expansion	The vector median	The proposed method
5×5	5.8 μ	4.8 μ	4.3 μ
6×6	5.3 μ	4.2 μ	3.8 μ
7×7	4.6 μ	3.9 μ	3.4 μ

"house" and "pepper" are shown in Figs. 11(a) and 11(b), respectively. A visual evaluation gives the impression that the proposed approach can be applied in color edge detection. It can also be applied to images with any color model other than the (R, G, B) model.

5. Conclusions and discussions

A new approach to reduction of color space dimensionality and its application in edge detection for color images have been proposed. The proposed method uses moment-preserving thresholding to obtain two representative colors for each image block and computes the difference of these two color vectors to find an axis for data projection in order to reduce the color space to one dimension. A single-spectral image is thus obtained. Experimental results show that this axis is close in effect to that computed by the KL expansion or the vector median approach. However, the computation time required by the proposed method is the least in the three methods. The moments of the projection results are finally com-

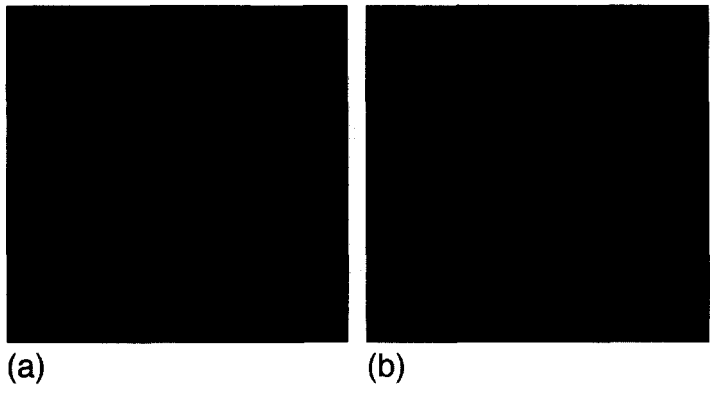

Fig. 10. The error images of the projection results. (a) Error image of "house" of the proposed method with respect to the KL expansion. (b) Error image of "pepper" of the proposed method with respect to the KL expansion.

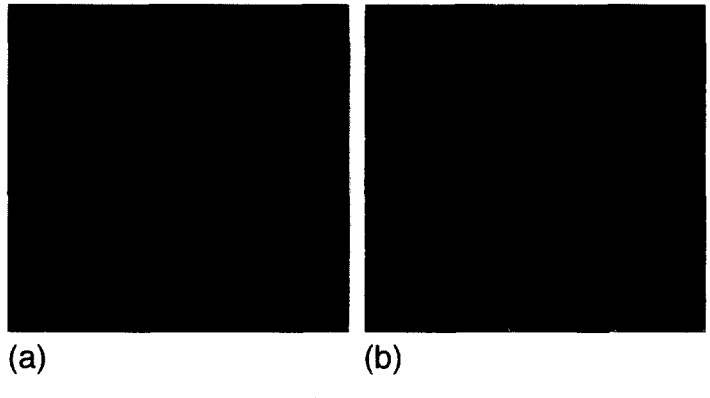

Fig. 11. The edge detection results. (a) Edges in "house". (b) Edges in "pepper".

puted for edge detection. The proposed approach can be applied to any color model. It has low complexity because the proposed approach employs a new way of circular window arrangement to reduce the overlapping areas of the windows used in edge detection. The information of the two color vectors and the edge location can be further employed for color image compression. This information can be obtained only from the proposed method and cannot be provided by the KL expansion and the vector median methods.

The idea of assuming only one edge feature in a block is not always true in the real world. However, the smaller the selected block size, the fewer the features existing in the block. So, the proposed approach is suitable for smaller blocks and not applicable for bigger blocks (8×8 blocks, for example). Applying the edge detection to bigger blocks will cause distortion in the edge detection result because more complicated features could be existing in the blocks.

References

Astola, J., P. Haavisto and Y. Neuvo (1990). Vector median filters. *Proc. IEEE* 78, 678–689.

Bigun, J. (1993). Unsupervised feature reduction in image segmentation by local transforms. *Pattern Recognition Lett.* 14, 573–583.

Chen, L.H. and W.H. Tsai (1989). Moment-preserving techniques in image processing–A survey and new approaches. *Proc. Natl. Sci. Counc. ROC* 13 (5), 280–301.

Davis, L.S. (1975). A survey of edge detection techniques. *Computer Vision, Graphics, and Image Processing* 4 (3), 248–270.

Delp, E.J. and O.R. Mitchell (1991). Moment-preserving quantization. *IEEE Trans. Commun.* 39, 1549–1558.

Di Zenzo, S. (1983). Advances in image segmentation. *Image Vision Comput.* 1 (4), 196–210.

Di Zenzo, S. (1986). A note on the gradient of a multi-image. *Computer Vision, Graphics, and Image Processing* 33, 116–128.

Fu, K.S. and J.K. Mui (1981). A survey on image segmentation. *Pattern Recognition* 13, 3–16.

Fukunaga, K. (1990). *Introduction to Statistical Pattern Recognition.* Academic Press, New York, 400–438.

Gonzalez, R.C. and P. Wintz (1987). *Digital Image Processing.* Addison-Wesley, Reading, MA, 334–340.

Hueckel, M. (1971). An operator which locates edges in digitized pictures. J. ACM 18 (1), 113–125.

Lyvers, E.P., O.R. Mitchell, M.L. Akey and A.P. Reeves (1989). Subpixel measurements using a moment-base edge operator. *IEEE Trans. Pattern Anal. Mach. Intell.* 11 (12), 1293–1309.

Machuca, R. and K. Philips (1983). Applications of vector fields to image processing. *IEEE Trans. Pattern Anal. Mach. Intell.* 5 (3), 316–329.

Mitchell, O.R., E.J. Delp and S.G. Carlton (1978). A new approach to image compression. *Proc. IEEE Internat. Conf. on Commun.*, 12B.1.1–4.

Nevatia, R. (1977). A color edge detection and its use in scene segmentation. *IEEE Trans. Syst. Man Cybernet.* 7 (11), 820–826.

Noble, B. and J.W. Daniel (1988). *Applied Linear Algebra.* Prentice-Hall, New York.

Peli, T. and D. Malah (1982). A study of edge detection algorithms. *Computer Vision, Graphics, and Image Processing* 20, 1–21.

Pratt, W.K. (1971). Spatial transform coding of color images. *IEEE Trans. Commun.* 18 (12), 980–992.

Pratt, W.K. (1991). *Digital Image Processing.* Wiley Interscience, New York.

Rosenfeld, A. and A.C. Kak (1982). *Digital Picture Processing.* Academic Press, New York, 223–237.

Shiozaki, A. (1986). Edge extraction using entropy operator. *Computer Vision, Graphics, and Image Processing* 36, 1–9.

Tabatabai, A.J. and O.R. Mitchell (1984). Edge location to subpixel values in digital imagery. *IEEE Trans. Pattern Anal. Mach. Intell.* 6 (2), 188–201.

Trahanias, P.E. and A.N. Venetsanopoulos (1993). Color edge detection using vector statistics. *IEEE Trans. Image Process.* 2 (2), 259–264.

Tsai, W.H. (1985). Moment-preserving thresholding: a new approach. *Computer Vision, Graphics, and Image Processing* 29, 377–393.

Weszka J.S. (1978). A survey of threshold selection techniques. *Computer Vision, Graphics, and Image Processing* 7, 259–265.

Yang, C.K., J.C. Lin and W.H. Tsai (1994). Color image compression using moment-preserving and BTC techniques. *Proc. IEEE ICIP-94* 3, 972–976.