Adaptive Methods for Dithering Color Images

Lale Akarun, *Member, IEEE,* Yasemin Yardımcı, *Member, IEEE,* and A. Enis Çetin, *Senior Member, IEEE*

***Abstract*—Most color image printing and display devices do not have the capability of reproducing true color images. A common remedy is the use of dithering techniques that take advantage of the lower sensitivity of the eye to spatial resolution and exchange higher color resolution with lower spatial resolution. In this paper, an adaptive error diffusion method for color images is presented. The error diffusion filter coefficients are updated by a normalized least mean square-type (LMS-type) algorithm to prevent textural contours, color impulses, and color shifts, which are among the most common side effects of the standard dithering algorithms. Another novelty of the new method is its vector character: Previous applications of error diffusion have treated the individual color components of an image separately. Here, we develop a general vector approach and demonstrate through simulation studies that superior results are achieved.**

I. Introduction

THE EYE can distinguish between thousands of distinct colors, whereas only a few dozen gray levels are enough to represent all gray scales. The high sensitivity of the eye to color has increased the demand for sophisticated color dithering algorithms that create the illusion of a "true color" image with a limited palette.

Color digital images that represent each color component with one byte are called *true color images*. Such a color image can contain 2^{24} colors. Most color image printing and displaying devices do not have the capability of reproducing true color images [1], [2]. Consequently, the number of colors has to be reduced drastically to produce a color image with a limited palette. This results in highly visible degradation in image quality. A common remedy is the use of dithering techniques that exploit the lower sensitivity of the eye to spatial resolution and exchange higher color resolution with lower spatial resolution. The eye averages the colors in a neighborhood of the point of interest and creates the illusion of more colors. Error diffusion achieves this effect by distributing the error encountered in quantizing a pixel to neighboring pixels, ensuring in effect, that the neighboring pixels are biased in the reverse direction. Some well-known error diffusion filters are Floyd–Steinberg, Stucki, Jarvice, Judice, and Ninke [3]–[6]. These error diffusion filters use a fixed kernel: The coefficients of the error diffusion filter are preset. In contrast, some more recent algorithms aim at finding the optimum coefficents for the given image [7]–[9] or to change the coefficients adaptively [9]–[11]. Most of the work on error diffusion deals with the dithering of gray-scale images. In order to extend these results to color images, the most straightforward approach is to consider each color component as an individual gray scale image [6], [7], [9], [12]. This scalar approach ignores the correlation between the color components.

Common problems that are addressed by dithering are color shift and the appearance of false textural contours. When an image is quantized into a limited number of colors that form a palette, the color of a region is classified into one of the colors in the palette, resulting in a shift that is visible to the eye. A worse-looking defect, however, is the formation of false edges in areas of the image where there is originally a smooth transition between two colors. Error diffusion solves these two problems in most of the cases: By balancing the quantization errors, it produces an average color shade that is closer to the original color of the region and breaks up the false edges. However, if the given color is close to the boundary of a color cluster and the filter coefficients are not adequately selected, the quantization errors accumulate and eventually, a palette color from a different cluster is produced. This manifests itself as a color impulse that is very disturbing. Another adverse effect of dithering manifests itself around edges: Colors from two sides of the edge are smeared to each other and sharp edges are converted to jagged edges.

In this paper, a new adaptive vector error diffusion technique is proposed to correct these adverse effects. The term *vector error diffusion* has been used to distinguish the present approach from previous approaches that have applied scalar error diffusion to the individual color components. The error diffusion filter coefficients are obtained by minimizing the mean square error between the average color of the original image and the dithered image. Considering the fact that there may be significant differences in the statistics of different regions in a typical image, a least mean square-type (LMS-type) adaptive algorithm is used to solve the minimization problem. This adaptive technique not only produces dithered images in which the average color shade is closest to the original and the occurance of color impulses is greatly reduced, but false textural contours do not appear.

Section II introduces the adaptive vector error diffusion algorithm. We introduce the general vector approach and formulate our adaptive error diffusion algorithm in this vector form. Section III deals with the scalar case: We reduce the adaptive error diffusion equations to a simpler form for this special case and discuss how to apply scalar adaptive error

Manuscript received March 18, 1996; revised January 28, 1997. This work was supported by TUBITAK under Grant EEEAG-115 and by the Boğaziçi University Research Fund under Grant 96HA0139. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. H. Joel Trussell.

L. Akarun is with the Department of Computer Engineering, Boğaziçi University, 80815 Bebek, İstanbul, Turkey (e-mail: akarun@boun.edu.tr).

Y. Yardımcı is with the Department of Electrical Engineering, Boğaziçi University, İstanbul, Turkey.

A. E. Çetin is with Bilkent University, Ankara, Turkey.

Publisher Item Identifier S 1057-7149(97)04733-7.

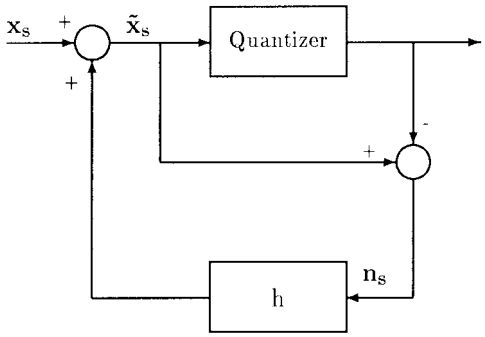

Fig. 1. Block diagram of the error diffusion method.

diffusion to color images. In Section IV, we present our simulation examples and discuss the effects of scaling the error diffusion filter coefficients. Finally, in Section V, we outline our future research plans and conclude.

II. Adaptive Vector Error Diffusion

A. Vector Error Diffusion

Fig. 1 shows the block diagram of the error diffusion technique. A color image is represented by 3×1 vectors $\mathbf{x}_s$ for $s = 0, 1, 2, \cdots, (M_1 \times M_2) - 1$. To simplify the equations, double indices denoting the location of the pixels have been replaced by the single index s. For an image of size $M_1 \times M_2$ the (m_1, m_2)nd pixel corresponds to the index $s = m_1 + m_2 M_1$. Given the image $\mathbf{x}_s$ and a quantizer Q, error diffusion works as follows: The quantization error $\mathbf{n}_s$ is diffused to neighboring pixels to create a dithered image $\tilde{\mathbf{x}}_s$. Then, the dithered image is quantized instead of the original image, as follows:

$$\tilde{\mathbf{x}}_s = \mathbf{x}_s + \sum_{k<s} \mathbf{H}_{s-k} \mathbf{n}_k \tag{1}$$

$$\mathbf{n}_s = \tilde{\mathbf{x}}_s - Q(\tilde{\mathbf{x}}_s) \tag{2}$$

where $k < s$ corresponds to a causal error diffusion mask and the filter coefficients $\mathbf{H}_i, \quad i = 1, \cdots N$ are matrices. If error diffusion is applied separately to the color components, these matrices are chosen to be diagonal (as in [12]); the error in a given color component is diffused to only that color component. In the present formulation, we do not require the error diffusion matrices to be diagonal in order to take full advantage of the correlation between the color components. N is the size of the causal filter mask. Fig. 2 shows the mask of the Floyd–Steinberg filter, where $N = 4$. Fig. 2(a) shows the Floyd–Steinberg filter for the scalar representation, where the coefficients are scalars, and Fig. 2(b) shows the same filter in the vector representation, where the coefficients are diagonal matrices.

B. Adaptive Error Diffusion Algorithm

In our algorithm, the error diffusion filter matrix $\mathbf{H} = [\mathbf{H_1}|\mathbf{H_2}|\cdots|\mathbf{H_N}]$ is obtained by minimizing the mean square error (MSE) between the average color value of the original image and the dithered image. In order to obtain the optimum

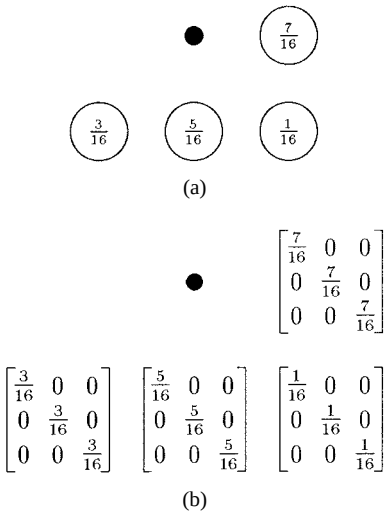

Fig. 2. Filter mask of the error diffusion method and the Floyd–Steinberg filter coefficients $(N = 4)$. (a) Scalar representation. (b) Vector representation.

error diffusion filter coefficients the quantity

$$E\{\|\mathbf{x_s} - Q(\tilde{\mathbf{x}}_\mathbf{s})\|^2\} \tag{3}$$

is to be minimized with respect to $\mathbf{H}_i$. This is equivalent to minimizing

$$E\left\{\left\|\mathbf{n_s} - \sum_{k<s} \mathbf{H}_{s-k}\mathbf{n_k}\right\|^2\right\} \tag{4}$$

with respect to the filter coefficient matrices. Differentiating (4) with respect to $\mathbf{H}_i$ and setting the results to zero, the following set of linear equations are obtained:

$$E\{\mathbf{n}_s \mathbf{n}_{s-i}^{\mathrm{T}}\} = \sum_{k<s} \mathbf{H}_{s-k} E\{\mathbf{n}_k \mathbf{n}_{s-i}^{\mathrm{T}}\} \tag{5}$$

where $i = 1, 2, \cdots, N$, and $\mathbf{n}_s$ is the quantization error at location s. By solving the system of equations in (5), the error diffusion filter coefficient matrix $\mathbf{H} = [\mathbf{H_1}|\mathbf{H_2}|\ldots|\mathbf{H_N}]$ can be obtained. The covariance matrix $E\{\mathbf{n}_k \mathbf{n}_{s-i}^{\mathrm{T}}\}$ is not available in practice but can be estimated from the quantization error statistics of the original image. However, this approach does not yield good results because a typical image cannot be assumed to be a stationary random field.

Considering the fact that there may be significant differences in the statistics of different regions of a typical image, we use an LMS-type adaptive algorithm to determine the error diffusion filter coefficients. The key step is to replace the ensemble average with the index average and to compute the stochastic gradient vector. We minimize

$$e^2(s) = \left(\mathbf{n}_s - \sum_{k<s} \mathbf{H}_{s-k}\mathbf{n_k}\right)^{\mathrm{T}} \left(\mathbf{n}_s - \sum_{k<s} \mathbf{H}_{s-k}\mathbf{n_k}\right). \tag{6}$$

At the index s, the filter coefficient vector is updated using the following equation:

$$\mathbf{H}(s+1) \;=\; \mathbf{H}(s) - \mu \nabla e^2(s) \tag{7}$$

where μ is the step size. The gradient is given by

$$\nabla e^2(s) = -2\left(\mathbf{n}_s - \sum_{k<s} \mathbf{H}_{s-k}\mathbf{n}_k\right)\mathbf{n}_{s-i}^{\mathrm{T}} \tag{8}$$

$$= -2\mathbf{e}(s)\mathbf{n}_{s-i}^{\mathrm{T}}. \tag{9}$$

In our simulation studies we employed the so-called normalized LMS update equation [13]

$$\mathbf{H}(s+1) \;=\; \mathbf{H}(s) + \mu\mathbf{e}(s)\left[\mathbf{n}_{s-1}^{\mathrm{T}} | \ldots | \mathbf{n}_{s-N}^{\mathrm{T}}\right] \tag{10}$$

where the step size parameter μ is chosen in the interval $(0,2)$ as in the case of the normalized LMS algorithm, and $\mathbf{e}(s)$ can be calculated as

$$\mathbf{e}(s) = (\tilde{\mathbf{x}}_s - Q(\tilde{\mathbf{x}}_s)) - \sum_{k<s} \mathbf{H}_{s-k}\mathbf{n}_{\mathbf{k}}. \tag{11}$$

Since the convergence is quite fast, we have used the coefficients found in the first iteration of the algorithm. The rows of $\mathbf{H}$ are normalized following the update: This corresponds to the weights summing to one.

III. Adaptive Scalar Error Diffusion

When the error diffusion filter coefficient matrices are chosen as diagonal, the error in a given color component is diffused to only that color component. The resulting filters reduce to the formulation given in [9], where the error diffusion filter coefficients h_i are scalars. In this formulation, the error diffusion equations now become

$$\tilde{\mathbf{x}}_{\mathbf{s}} = \mathbf{x}_{\mathbf{s}} + \sum_{k<s} h_{s-k}\mathbf{n}_{\mathbf{k}} \tag{12}$$

$$\mathbf{n}_{\mathbf{s}} = \tilde{\mathbf{x}}_{\mathbf{s}} - Q(\tilde{\mathbf{x}}_{\mathbf{s}}). \tag{13}$$

This error diffusion procedure is actually separable as mentioned above. The vector $\mathbf{h} = (h_1 \cdots h_N)^T$ represents the filter coefficients in the support region. $\mathbf{h}$ is obtained by minimizing the sum of MSE's in each color component. Analogous to (3), the following quantity is minimized to obtain the optimum error diffusion filter coefficients:

$$E\{\|\mathbf{x}_{\mathbf{s}} - Q(\tilde{\mathbf{x}}_{\mathbf{s}})\|^2\} = \sum_{\ell=1}^{3} \left(x_s^\ell - Q(\tilde{x}_s^\ell)\right)^2 \tag{14}$$

where ℓ denotes a specific color component such as red, green, or blue. Proceeding as in (4)–(11), the update equations for h^ℓ, the filter coefficient vector for color component ℓ, are obtained as

$$\mathbf{h}^\ell(s) \;=\; \mathbf{h}^\ell(s) + \mu\frac{e_\ell(s)\mathbf{z}_s^\ell}{\|\mathbf{z}_s^\ell\|^2} \tag{15}$$

where the vector $\mathbf{z}_s^\ell$ contains the past quantization errors

$$\mathbf{z}_s^\ell = \left[n_{s-1}^\ell \ldots n_{s-N}^\ell\right]^{\mathrm{T}} \tag{16}$$

and the step size parameter μ is chosen in the interval $(0,2)$ as in the case of the normalized LMS algorithm, and $e_\ell(s)$ can be calculated as follows:

$$e_\ell(s) = (\tilde{\mathbf{x}}_s - Q(\tilde{\mathbf{x}}_s))^\ell - \sum_{k<s} h_{s-k}^\ell n_k^\ell. \tag{17}$$

This scalar formulation is simpler in terms of computational complexity. However, the correlation between the color components is not taken into account. In red–green–blue (RGB) color space, the correlation between the red, green, and blue components is rather high. One alternative is to work in a color space in which this correlation is reduced. The Karhunen–Loeve color coordinate system, a pseudo-Karhunen–Loeve transformation (KLT) color coordinate system, and the YIQ color coordinate system are possible candidates:

A. KL Color Coordinate System

In a common image, the color components in the RGB coordinate system are highly correlated with each other. If the second-order statistics of the image are known, it is possible to derive an orthogonal color coordinate system by a KLT [14]. The KL color transform is defined as

$$\begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} = \begin{bmatrix} m_{11} & m_{12} & m_{13} \\ m_{21} & m_{22} & m_{23} \\ m_{31} & m_{32} & m_{33} \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

where the transformation matrix $\mathbf{M}$ composed of terms m_{ij} consists of the eigenvectors of the covariance matrix of the RGB vectors. Let $\mathbf{U}$ represent the covariance matrix with general terms u_{ij}. The transformation matrix satisfies the relationship

$$\mathbf{MUM}^{\mathrm{T}} = \Lambda$$

where Λ is a diagonal matrix with the eigenvalues of $\mathbf{U}$ along its diagonal.

B. Pseudo-KL Coordinate System

Considering the difficulty of estimating the color covariance for each individual image, a compromise is to estimate the covariance of a general image and to find the KL transformation for that image. Such a transformation is given in [15], as follows:

$$\begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} = \begin{bmatrix} 0.33 & 0.33 & 0.33 \\ 0.5 & 0 & -0.5 \\ -0.25 & 0.5 & -0.25 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}.$$

C. YIQ Color Coordinate System

YIQ is the color coordinate system of the National Television Systems Committee (NTSC) television standard defined [14] as

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

IV. Simulation Results

We have implemented both the vector adaptive error diffusion algorithm and the scalar adaptive error diffusion algorithm. For scalar adaptive error diffusion, we use different color coordinate systems as described in Section III. We compare the results of the adaptive algorithms with that of the Floyd–Steinberg filter. For the adaptive filters, we use the same

support size, that is, $N = 4$, as in Fig. 2. The color images corresponding to the results are given in Figs. 3 and 4. The original images of Figs. 3(a) and 4(a) have been quantized to 16 colors with the median-cut algorithm [16] in Figs. 3(b) and 4(b). It is observed that quantization to 16 colors results in the appearance of false edges. Figs. 3(c) and 4(c) are the results of error diffusion with the Floyd–Steinberg algorithm. It is observed that the application of the Floyd–Steinberg algorithm causes the creation of color impulses and false textures. Those color impulses are most disturbing as isolated yellow pixels on a dark green background and dark green pixels on the tip of the leaf in Fig. 3(c), and as red pixels on the yellow pepper in the lower left side in Fig. 4(c). Disturbing textures appear on the tip of the leaf in Fig. 3(c) and on the green pepper in the middle in Fig. 4(c). It is also noted that some false contours remain in these images. Fig. 3(d)–(g) and 4(d)–(g) are the results of scalar error diffusion using the RGB, YIQ, pseudo-KL, and KLT color coordinate systems, respectively. Finally, Figs. 3(h) and 4(h) show the results of the vector adaptive error diffusion algorithm. It is observed that all the adaptive algorithms show enhanced performance when compared to the Floyd–Steinberg algorithm. The vector adaptive error diffusion shows the best results, as expected: The appearance of disturbing colored impulses and false textures is minimized, while the false contours are completely eliminated. When the scalar adaptive error diffusion algorithm is used, utilization of the RGB color coordinate system yields the worst performance, while use of the YIQ, pseudo-KL, and KLT color coordinate systems bring comparable improvements. It should be noted, however, that the KLT coordinate system is image-dependent, and brings an additional computational burden. The pseudo-KL coordinate system gives the overall best result among the three when the presence of false textures is also taken into account. Table I summarizes these subjective comparisons.

In terms of computational complexity, both the adaptive algorithms and doing vector error diffusion bring additional burden. In vector error diffusion, the error from three color components is diffused; therefore, the number of operations is increased threefold. Computing the filter coefficients adaptively as opposed to using fixed coefficients also means additional computation, as seen in (10). However, it should be noted that quantization and dithering are performed together, with quantization taking several orders of magnitude longer than dithering. Therefore, the additional increase in computational complexity due to the adaptive dithering algorithms would not be significant.

The areas that are most difficult to handle by color dithering are areas of smooth change, like those found in Fig. 3: While error diffusion reduces the false contours that show in such areas, it also leads to the appearance of color impulses, isolated pixels of opposite color on a smooth background. In many standard algorithms, the error diffusion filter coefficients are defined to sum to one. That corresponds to diffusing all of the error encountered in quantizing a pixel to neighboring pixels. If the coefficients are scaled by a figure that is smaller than one, the effect is a cross between error diffusion dithering and doing no dithering at all: The result is a compromise between the reduction of false contours and the appearance of color impulses. If the scaling coefficient is around 90%, the appearance in terms of color impulses is improved significantly with no visible increase in false contours.

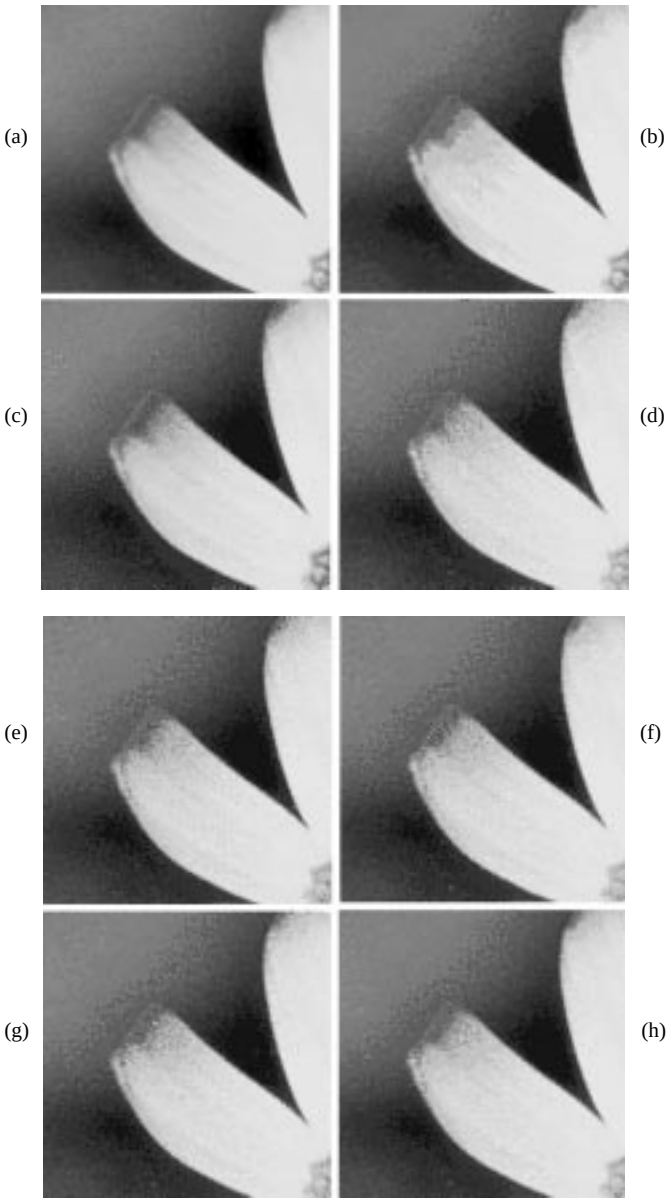

Fig. 3. (a) Original image. (b) Quantized to 16 colors with the median-cut algorithm. (c) Floyd–Steinberg dithering. (d)–(g) are results of the scalar adaptive error diffusion algorithm in different color coordinate systems: (d) RGB; (e) YIQ; (f) Pseudo-KL; (g) KLT; (h) Result of dithering using the vector adaptive error diffusion algorithm.

V. Conclusion

In this paper, we have developed a novel vector-based approach for color error diffusion. We have derived an adaptive algorithm for obtaining the optimum error diffusion filter matrices. The adaptation criterion is based on the quantization error statistics obtained from the dithered image. We have simplified the vector error diffusion equations for the scalar case and suggested the use of color coordinate systems that have less correlation between the color components. In standard error diffusion methods, the filter has a unity gain. We

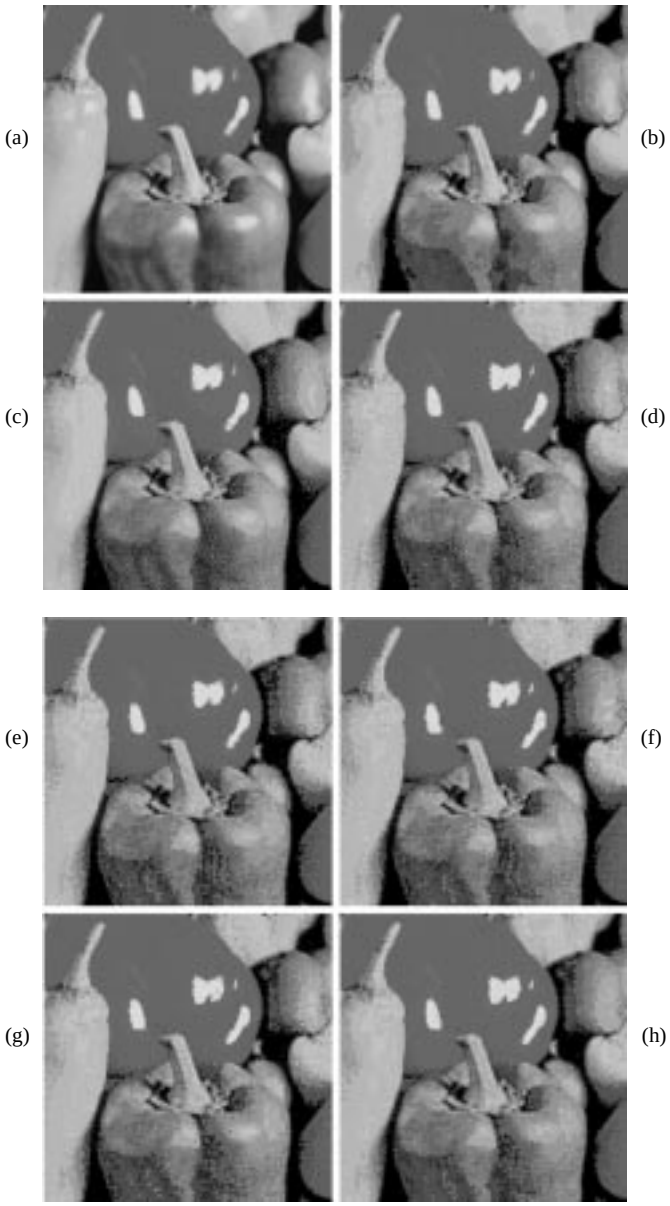

Fig. 4. (a) Original image. (b) Quantized to 16 colors with the median-cut algorithm. (c) Floyd–Steinberg dithering. (d)–(g) Results of the scalar adaptive error diffusion algorithm in different color coordinate systems: (d) RGB; (e) YIQ; (f) Pseudo-KL; (g) KLT; (h) Result of dithering using the vector adaptive error diffusion algorithm.

observed that this restriction can be relaxed and the diffusion filter may have a gain slightly less than one.

Our simulations have shown the superiority of the vector adaptive error diffusion algorithm: The appearance of color impulses is greatly eliminated and smoother color transitions are achieved. The use of the pseudo-KL coordinate system with the scalar adaptive error diffusion filter yields the second best performance. Both of the adaptive error diffusion algorithms show a distinct improvement in performance when compared with the Floyd–Steinberg filter.

The scaling of the error diffusion filter coefficents is an important step in the performance of the algorithms. This scaling coefficient controls the balance of false edges and color impulses; it can, therefore be varied in different regions of an image to achieve different goals. Our further work aims to include the control of this coefficient in the adaptation algorithm.

TABLE I
SUBJECTIVE COMPARISON OF DITHERING ALGORITHMS

Flower (Figure 3)			
Algorithm	Texture	Impulses	False Edges
(b) No dithering	none	none	heavy
(c) F-S [4]	medium	heavy	medium
(d) Scalar AED: RGB	some	medium	none
(e) Scalar AED:YIQ	low	low	none
(f) Scalar AED:p-KL	low	low	none
(g) Scalar AED:KLT	low	low	none
(h) Vector AED	low	none	none
Peppers (Figure 4)			
Algorithm	Texture	Impulses	False Edges
(b) No dithering	none	none	heavy
(c) F-S [4]	medium	heavy	medium
(d) Scalar AED: RGB	medium	low	none
(e) Scalar AED:YIQ	medium	low	none
(f) Scalar AED:p-KL	low	low	none
(g) Scalar AED:KLT	medium	low	none
(h) Vector AED	low	none	none

REFERENCES

[1] J. Allebach, "Digital color imaging: Bringing color to the desktop," *IEEE SP-Society Distinguished Lecturer's Series: 3. Sinyal İsleme Kurultayı*, Goreme, Turkey, May 1995.
[2] T. N. Pappas and D. L. Neuhoff, "Printer models and error diffusion," *IEEE Trans. Image Processing*, vol. 4, pp. 66–79, Jan. 1995.
[3] R. A. Ulichney, "Dithering with blue noise," in *Proc. IEEE*, vol. 76, pp. 56–79, Jan. 1988.
[4] R. W. Floyd and L. Steinberg, "An adaptive algorithm for spatial grayscale," in *Proc. SID*, 1976, vol. 17, no. 2, pp. 75–77.
[5] J. F. Jarvis, C. N. Judice, and W. H. Ninke, "A survey of techniques for the display of continuous tone pictures on bilevel displays," *Comput. Graph. Image Processing*, vol. 5, pp. 13–40, 1976.
[6] R. Gentile, E. Walowit, and J. Allebach, "Quantization and multilevel halftoning of color images for near-original quality," *J. Opt. Soc. Amer. A.*, vol. 7, pp. 1019–1026, 1990.
[7] A. Zakhor, S. Lin, and F. Eskafi, "A new class of B/W and color halftoning algorithms," in *Proc. ICASSP-91*, Toronto, Ont., Canada, May 1991, vol. 4, pp. 2801–2804.
[8] B. L. Shoop and E. K. Ressler, "Optimal error diffusion for digital halftoning using an optical neural network," in *Proc. ICIP*, Austin, TX, Oct. 1994, pp. 1036–1040.
[9] L. Akarun, Y. Yardimci, and A. E. Çetin, "New methods for dithering color images," in *Proc. IEEE Workshop Nonlinear Signal and Image Processing*, Halkidiki, Greece, June 1995, vol. 2, pp. 523–526.
[10] R. P. Mironov and R. K. Kunchev, "Adaptive error duffision for image quantization," *Electron. Lett.*, vol. 29, pp. 2021–2022, Nov. 1993.
[11] P. W. Wong, "Error diffusion with dynamically adjusted kernel," in *Proc. ICASSP-94*, Adelaide, Australia, Apr. 1994, vol. 5, pp. 113–116.
[12] M. T. Orchard and C. A. Bouman, "Color quantization of images," *IEEE Trans. Signal Processing*, vol. 39, pp. 2677–2690, Dec. 1991.
[13] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1985.
[14] W. K. Pratt, *Digital Image Processing*, 2nd ed. New York: Wiley, 1991.

[15] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision.* Reading, MA: Addison-Wesley, 1992.
[16] P. Heckbert, "Color image quantization for frame buffer display," *Comput. Graph.*, vol. 16, pp. 297–307, July 1982.

Lale Akarun (S'87–M'92) received the B.S. and M.S. degrees in electrical engineering from Boğaziçi University, İstanbul, Turkey, in 1984 and 1986, and the Ph.D. degree from Polytechnic University, Brooklyn, NY, in 1992.

From 1993 to 1995, she was Assistant Professor of Electrical Engineering at Boğaziçi University, where she is now Associate Professor of Computer Engineering. Her current research interests are in image processing, computer vision and computer graphics.

Yasemin Yardımcı (S'92–M'95) received the B.S. and M.S. degrees in electrical engineering from Boğaziçi University, İstanbul, Turkey in 1983 and 1988, and the Ph.D. degree from Vanderbilt University, Nashville, TN, in 1994.

She worked as an adjunct professor at İstanbul University and Bilkent University, Ankara, Turkey, and supervised projects on speech processing. She is spending the 1996–1997 academic year in the Department of Electrical Engineering at the University of Minnesota, Minneapolis. Her current research interests are in speech processing and array processing.

A. Enis Çetin (S'85–M'87–SM'95) received the B.Sc. degree in electrical engineering from Middle East Technical University, Ankara, Turkey, and the M.S.E. and Ph.D. degrees from University of Pennsylvania, Philadelphia, in 1987.

He was Assistant Professor of electrical engineering at the University of Toronto, Ont., Canada, from 1987 to 1989. He is currently Professor of Electrical Engineering at Bilkent University, Ankara, Turkey, and is on sabatical leave at the University of Minnesota, Minneapolis. During the summers of 1988, 1991, and 1992, he was a consultant and resident visitor at Bell Communications Research (Bellcore), Morristown, NJ. In 1996, he was a consultant for Optron, Inc., and KAREL Elektronik, Ankara, Turkey.

He is currently a member of the DSP Technical Committee of the IEEE Circuits and Systems Society.