A Progressive Scheme for Digital Image Halftoning, Coding of Halftones, and Reconstruction

Stefanos Kollias, *Member, IEEE*, and Dimitris Anastassiou, *Member, IEEE*

***Abstract*—A novel digital halftoning technique is presented for the efficient transformation of gray scale images into bilevel ones, based on a progressive generation of the bilevel image pixels in a parallel way. An image distortion criterion, in which the gray-tone image is approximated by a filtered version of the halftoned image, is used for this purpose. It is also shown that this criterion can be used for the reconstruction of gray-tone images from halftoned versions of them. The quality of the reconstructed image is examined through simulation studies and it is shown that it can further be improved by postprocessing the image using a nonlinear edge-preserving noise-smoothing filter. Following the above, a combined scheme is derived in which continuous-tone images are progressively coded and transmitted in bilevel form and can be reconstructed in gray-scale form.**

I. Introduction

Digital halftoning of images is a nonlinear signal processing technique for the rendition of photographs in bilevel displays or printers [1]–[4]. Image halftoning methods create a bilevel image whose average density, i.e., the proportion of black dots in a small neighborhood around each pixel, approximates the darkness associated with that pixel. These methods have been developed in an empirical way so that, when the halftoned image is viewed from some distance, it is perceived as a pseudo-gray-scale image.

In this paper, we introduce a suitable-to-progressive transmission halftoning algorithm which is simple, effective, and amenable to parallel implementation. The algorithm divides the image pixels into classes, which are processed one after the other in a progressive manner. The computation of the pixel values in each class is the result of a simple thresholding decision and can be conceived of as being performed by an artificial neural network. An appropriate image distortion error criterion is used for this purpose. The technique may be easily incorporated in progressive transmission methods [5] by progressively coding and transmitting the halftoned bilevel image. As a consequence, an approximate bilevel image is built up quickly in the former steps, while a more detailed image is progressively transmitted in the latter steps of the algorithm. The process of transmission can be interrupted whenever the generated halftoned image is considered to be of sufficiently good quality.

The inverse problem of reconstructing a gray-tone image from a halftoned version of it is also of great importance. The same image distortion measure, used in image halftoning, can be included in the reconstruction process. As a result, a combined progressive halftoning and reconstruction scheme is described which can be used for gray-tone image coding and transmission in bilevel form.

The paper is organized as follows. Section II presents the image distortion measure which is used in the proposed scheme. The progressive halftoning technique is described in Section III. The reconstruction procedure, which uses linear and nonlinear filtering, is presented in Section IV. Examples are given in Section V, illustrating the performance of the proposed techniques.

II. The Error Criterion

Let $u(m, n)$ be a given continuous-tone image with pixel values normalized in the interval [0, 1], which is approximated by a halftoned bilevel image $x(m, n)$ with pixel values being either 0 or 1, 0 standing for a black dot and 1 for a white dot. It is desired that the distortion introduced by this approximation be as small as possible. The bilevel image should, therefore, be generated so that an appropriately chosen distortion measure be minimized. Different choices of distortion measures will generally lead to different solutions of this problem. It has been shown [6], [7] that a frequency-weighted mean-squared error criterion is appropriate, according to the properties of the human visual system. A specific form of this criterion is given by:

$$E = \sum_{m=1}^{N} \sum_{n=1}^{N} \left\{ u(m, n) - \sum_{i=-k}^{k} \sum_{j=-k}^{k} w_{ij} x(m - i, n - j) \right\}^2 \tag{1}$$

where w_{ij} are the model coefficients normalized so that their sum equals unity, k is an integer which defines the model support region as shown in Fig. 1, and the image size is $N \times N$ pixels. This criterion tends to enhance the high-frequency components of the output image, but this can be desirable because the edges will also be sharpened (note that if u is bilevel, then x is not necessarily found identical to u). According to the above criterion, the orig-

Manuscript received September 19, 1988; revised October 10, 1989.
S. Kollias is with the Computer Science Division, Department of Electrical Engineering, National Technical University of Athens, Athens, Greece.
D. Anastassiou is with the Department of Electrical Engineering, Columbia University, New York, NY.
IEEE Log Number 9107585.

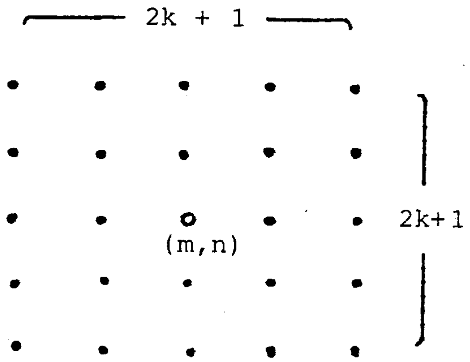


Fig. 1. The error criterion model support region.

inal image should be approximated by a lowpass-filtered version of the output bilevel image. Since, however, we are using a one-pass technique, we do not attempt to find the absolute minimum of (1); we attempt to approximate, for each pixel (m, n), the value of $u(m, n)$ by:

$$\sum_{i=-k}^{k} \sum_{j=-k}^{k} w_{ij} x(m - i, n - j). \tag{2}$$

This approximation implies that

$$x(m, n) \simeq c_{mn} \tag{3}$$

where

$$c_{mn} = \left(\frac{1}{w_{00}}\right) u(m, n) - \sum_{i=-k}^{k} \sum_{j=-k}^{k} \left(\frac{w_{ij}}{w_{00}}\right) \cdot x(m - i, n - j) \tag{4}$$

with $(i, j) \neq (0, 0)$. Assuming that the bilevel pixel values on the right-hand side of (4) are known, the bilevel value of $x(m, n)$ can be computed by the following thresholding rule:

$$x(m, n) = \begin{cases} 1 & \text{if } c_{mn} \geq 0.5 \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

Equations (3)–(5) describe the operation of a simple processing element, usually called neuron in artificial neural network literature, which computes its output by thresholding a weighted average of its $(k^2 - 1)$ neighboring neuron outputs. Thus, the halftoning problem described by (1) can be solved using a near-neighbor-connected artificial neural network, each neuron of which corresponds to a pixel of the image [6]–[8]. The performance of the technique described by (3)–(5) depends on the selection of the coefficients w used in (1) and (4). These coefficients define a filter, which can be thought of as the filter used by the human visual system to decide on whether a halftoned image is a good reproduction of a gray-tone image. A form of this filter can be derived using the results of psychophysical experiments [6], [9]. Another form of this filter can be derived, as is described next. Let us assume that a training set of both pairs of gray-tone images and good halftoned reproductions of them is available. Such a set can, for example, be obtained by selecting appropriate results of standard halftoning techniques, such as error diffusion [1] or dithering [4]. The filter coefficients w can then be obtained by minimizing the image distortion measure in (1), using the available set of data. Since this measure is a linear function of the coefficients w, its minimization can be performed by fast least-squares estimation algorithms.

III. The Progressive Halftoning Technique

A short description of the progressive halftoning scheme can be found in [10]. The pixels of the desired bilevel image are separated in different classes, which are processed in consecutive steps, one class at each step. This class separation has been proposed in [9] and used in various schemes [5]. The values of the pixels in each class are calculated by (4) and (5), using the pixel values of previously computed classes and ignoring the rest of the x terms, i.e., the pixel values which have not yet been computed. In doing this, only the subset of the model support region which contains the pixels already processed is considered, and the corresponding coefficients in (4) are renormalized so that they sum to one.

The classes are generated in the following way. An initial coarsely sampled rectangular lattice is chosen, composed of the pixels $x(ml, nl)$ where l is a fixed distance equal to a power of two, i.e., 2^r and $m, n = 1, \cdots, N$. In general, l should be equal to or slightly larger than $2k$ so that it defines a rectangular area which is similar to the model support region. Let us assume that this original lattice has already been halftoned using, for example, a constant threshold. A set of classes of sublattices is then generated. The first class is a rectangular lattice containing the pixels $x(ml + l/2, nl + l/2)$. The second class is a diagonally arranged lattice composed of the pixels $x(ml, nl + l/2)$ and $x(ml + l/2, nl)$. The pair of the next two classes are defined exactly as the first pair, if the value of l is divided by two and is then used to select the pixel locations. The definition of the following classes is similarly performed. It can easily be verified that the number of classes or, equivalently, the number of steps of the progressive halftoning algorithm is equal to $2r$. In the case of $l = 4$, the bilevel image consists of replicas of the 4×4 window shown in Fig. 2(a). This figure illustrates the five pixel classes defined in this case. However, several variations of the above class definition are possible by separating the classes shown in Fig. 2(a) into subclasses. At the extreme, a scheme results which contains l^2 classes. The resulting classes in this case, for $l = 4$, are shown in Fig. 2(b).

It should be mentioned that the calculated pixel values at each step of the progressive halftoning technique are used in the next steps to compute the values of other pixels in their neighborhood, so that the computed average value is approximately equal to the corresponding continuous-tone image pixel value. Each stage of progressive halftoning corresponds to a rendition of the image at a

1	5	3	5
5	4	5	4
3	5	2	5
5	4	5	4

(a)

1	13	3	15
9	5	11	7
4	16	2	14
12	8	10	6

(b)

Fig. 2. Definition of classes ($l = 4$) using a: (a) minimum; (b) maximum number of classes.

TABLE I
THE RELATION BETWEEN PARALLELISM AND PICTURE QUALITY OF THE HALFTONING TECHNIQUE

Parallelism	Number of Classes	Picture Quality	Complexity of Calculations
parallel	large	better	small
parallel	small	worse	small
recursive	large	better	small
recursive	small	better	large

lower spatial resolution, in which each "augmented" pixel can be thought of as a collection of pixels in one of the forms shown in Figs. 6(a)–(g) later in the paper. Therefore, if aliasing is to be reduced, (4) must be modified to include a weighted average of gray-scale image pixel values over a region, say R, around the currently processed pixel instead of the value of this pixel. The value $u(m, n)$ in (4) should, therefore, be replaced by

$$\sum_{(n,j)\in R} w'_{ij} u(m - i, n - j) \tag{6}$$

where, for simplicity, the weights w'_{ij} are chosen equal to w_{ij} and the region R has the same form as in Fig. 1. The complexity of the above scheme is equal to the number of computations required to calculate the right-hand side of (4) and the cost of the thresholding operation (5). The number of computations required to evaluate (4) depends on the model support region parameter k and on the class definition adopted. Our experiments, concerning the error criterion (1), have shown [7] that a value of $k = 3$ is usually required for good results. As a consequence, a distance of $l = 8$ is more appropriate in the progressive halftoning scheme. Moreover, we found that the extra complexity, caused by the inclusion of the term (6) in the right-hand side of (4), does not lead to significantly better results. For this reason, we have not used this term in the examples presented in the rest of the paper.

The progressive generation of the halftoned image is the main advantage of the proposed technique, compared with other recursive schemes like error diffusion. It should be stressed that the computation of each pixel value, in all steps of the algorithm, can be performed independently of all other pixels of the same class. Following the above, an efficient parallel implementation can be used for the computation of pixels which belong to the same class. The progressive halftoning scheme can, therefore, be viewed as a feedforward multilayer neural network, each layer of which corresponds to a class of the bilevel image pixels. The number of classes, or layers, should be selected considering the relation between the parallelism of the whole approach and the quality of the generated halftoned image. This relation is explained next. The use of the error criterion (1) in the form of (3)–(5) is more effective in the latter steps of the progressive scheme, where many previous pixel classes have already been computed and can be used for a meaningful application of (2). On the contrary, in the former steps of the method, the effects of the simple thresholding decision are more dominant. As a consequence, a larger number of classes, as in Fig. 2(b), is more appropriate for achieving better picture quality, using (3)–(5), when a parallel computation of the pixels in each class is performed. A smaller number of classes, as in Fig. 2(a), can be adequate for good picture quality when the pixels of each class are computed recursively. This is true because in the latter steps of the method, where the distance between pixels of the same class is smaller than the model support region width, the computed value of each pixel is used next to calculate the values of the neighboring pixels. The above discussion is summarized in Table I. It should be added that recursive nonparallel computation is slower, but requires simpler hardware.

Using the above-described technique, a continuous-tone image can progressively be coded and transmitted as a bilevel image. As a result, uncompressed 1 bit/pixel transmission is straightforward. If some extra compression is desired, the encoding of the different pixel classes in the bilevel image can be performed simultaneously with each thresholding decision (5) by using simple predictive arithmetic coding [5], [12], [13] in which the patterns of black and white dots in a template of previously computed neighboring pixels is analyzed by statistical count for determining the probabilities of a black or white dot occurring at the current pixel location. The specific forms of the templates used in our approach are described in Section V of the paper. In general, however, due to the nature of halftoned images, compression of the bilevel pixel values is relatively small, typically 0.7 to 0.8 bits per pixel. Nevertheless, the compression ratio can be improved without significantly deteriorating the image quality if the thresholding decisions are biased in the following manner. Whenever the condition of (5) is "fuzzy," i.e., whenever $c_{mn} \in (0.5 - \epsilon, 0.5 + \epsilon)$, then the decision is not based on (5); instead, in that case, the current pixel is thresholded to the value that will make the collected statistics more skewed, i.e., leading to lower entropy, enhancing therefore the compressibility of the output image. The compression will become larger as the bias ϵ is increased, but too large an increase of ϵ will cause the reproduced image quality to deteriorate.

IV. Reconstruction of Gray-Tone Images

The problem of reconstructing a gray-scale image from a halftoned version of it is of great importance in many applications. In image transmission, a combined halftoning and reconstruction scheme can be used for gray-scale image coding and transmission in bilevel form. This scheme is not an optimum technique of gray-scale image transmission, but is meaningful whenever the receiver has access to a bilevel or even gray-scale display. In the following, the distortion measure described by (1) is also used for reconstructing a gray-scale image from the halftoned one. The reconstructed image $u'(m, n)$ is generated as follows:

$$u'(m, n) = \sum_{i=-k}^{k} \sum_{j=-k}^{k} w_{ij} x(m - i, n - j) \tag{7}$$

using the linear filter with coefficients w. Equation (7) can be applied in each step of the progressive scheme at the time that the pixels of each class are received, resulting in a combined progressive coding, transmission, and reconstruction scheme. Results which show the quality of these reconstructed images are given in the next section. Due to the nature of halftoned images, a slight roughness can be observed in the textured areas of the reconstructed image, which simple linear filtering has not been able to remove. This roughness can be eliminated by postprocessing the reconstructed image using a nonlinear edge-preserving and noise-smoothing filter. We have used the following two-dimensional recursive filter from [14]:

$$\begin{aligned} y(m, n) = \tfrac{1}{3} \{ & g[\Delta_1(m, n)]\, \Delta_1(m, n) \\ & + g[\Delta_2(m, n)]\, \Delta_2(m, n) \\ & + g[\Delta_3(m, n)]\, \Delta_3(m, n) + u'(m, n) \} \end{aligned} \tag{8}$$

where $m \geq 0$, $n \geq 0$, and

$$\begin{aligned} \Delta_1 &= y(m - 1, n) - u'(m, n) \\ \Delta_2 &= y(m, n - 1) - u'(m, n) \\ \Delta_3 &= y(m - 1, n - 1) - u'(m, n). \end{aligned} \tag{9}$$

Function g should take negative values in areas where an edge-enhancing effect is desired and positive values in areas where noise smoothing is preferred. An appropriate choice of g is [14] the second derivative of a Gaussian function:

$$g(\Delta_i) = \exp\left(\frac{-q\Delta_i^2}{2\sigma^2}\right)\left(1 - \frac{\Delta_i^2}{\sigma^2}\right) \tag{10}$$

where parameter q takes values in the interval (0, 1) determining how far the g curve stretches into the negative region, and the value of σ controls the mainlobe width of the g curve. It should be added that a symmetrical edge-sharpening effect is obtained by incorporating a multidirectional technique in which the outputs of the filter applied in different directions are averaged.

Equations (7) and (8) can be used to reconstruct a gray-tone image from a binary halftoned version of it. It should be mentioned that in the case of image transmission, the application of (7), (8) does not augment the achieved compression ratio because these equations are based only on the received halftoned image, provided that a standard set of model weights is used. An image of even better quality can be produced at the receiver at the expense of a somewhat lower compression ratio, if the reconstruction error

$$\epsilon(m, n) = u(m, n) - y(m, n) \tag{11}$$

is appropriately coded, transmitted, and used for the reconstruction of the image. Standard adaptive DCT can be used for coding this error. However, due to its rather random nature, the reconstruction error can also be coded using specific techniques. In this paper, an appropriate thresholding of the error image is used to zero-out rather small error values without affecting the quality of the reconstructed image, and is then followed by quantization of the remaining nonzero values. A deadband quantizer, which is similar to a technique proposed for encoding the highband channel of subband coding of images [15], is used for this purpose. Run-length coding, followed by variable-length coding, can be designed for the coding of the positions and magnitudes of the nonzero values of the error image.

V. Examples

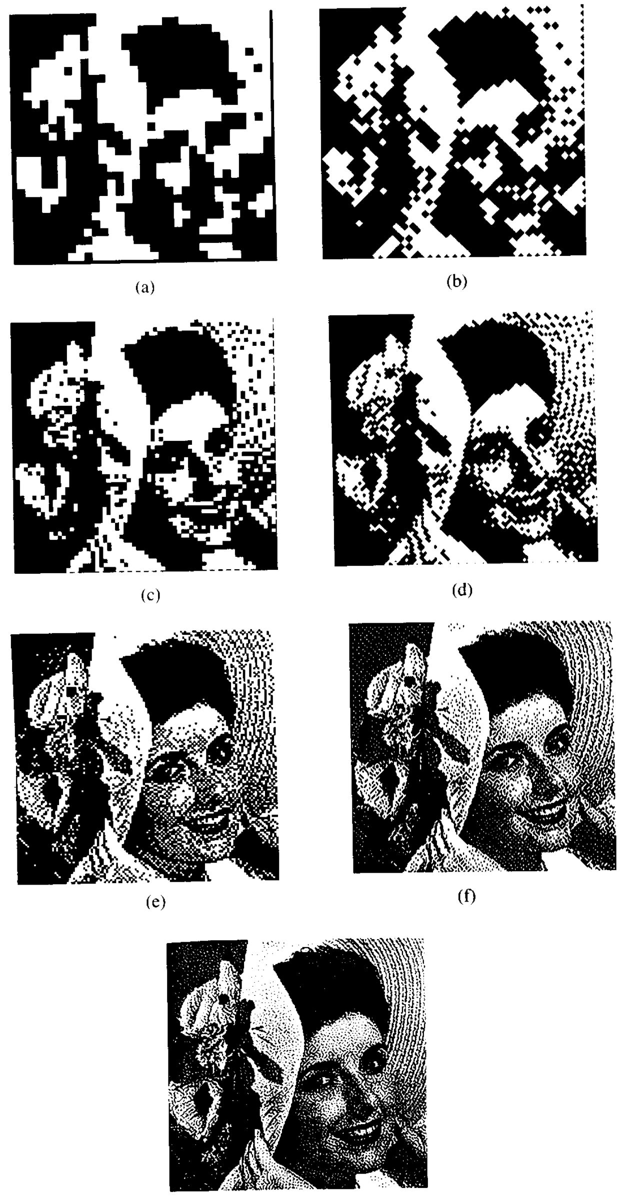
Examples are given next, which illustrate the performance of the method. A standard 256 × 256 pixel 8-bit gray-tone image, shown in Fig. 3, has been used in the first experiment. The progressive scheme was used to generate a halftoned version of this image, based on the class definition of Fig. 2(a) with $l = 8$, $k = 4$. A simple easily-implementable choice of the model coefficients w_{ij}, which has the general form of coefficients obtained from psychophysical experiments, was to select them as exponentially decreasing powers of two with respect to the corresponding pixel distances from the current (m, n) point. These values are shown in Fig. 4(a). Another choice, obtained using a training set of gray-scale and corresponding halftoned images [7] as was described in Section III, is given for comparison purposes in Fig. 4(b). The initial image, halftoned using a constant threshold, as well as the progressively generated images in each step of the algorithm, using the weights of Fig. 4(a), are shown in Fig. 5(a)–(g). A similar result was obtained using the class definition of Fig. 2(b) with $l = 8$ and $k = 4$, or $k = 3$. It should be mentioned that only a part of the halftoned image pixel values have been computed in the end of each step of the algorithm. Fig. 5(a)–(g) has been created by letting all pixels in a window (template) around the currently processed time (m, n), have the same intensity $x(m, n)$ with that pixel. This has been accomplished, without undesired overlapping, by using a window of l^2 pixels in the initial image and by reducing its size by a factor of two in each step of the algorithm. This window was rectangular or diagonally arranged, depending on the corresponding class of pixels which was generated, as shown

Fig. 3. An original gray-scale image.

1	1	2	4	2	1	1
1	2	4	8	4	2	1
2	4	8	16	8	4	2
4	8	16	32	16	8	4
2	4	8	16	8	4	2
1	2	4	8	4	2	1
1	1	2	4	2	1	1

-0.2	-0.5	-1.0	-1.2	-1.7	-1.5	-1.0
-1.0	-0.3	1.2	4.0	2.5	0.3	-1.2
-1.2	2.5	5.7	10.1	7.6	4.7	0.2
-1.5	5.7	11.6	16.0	11.6	5.7	-1.5
0.2	4.7	7.6	10.1	5.7	2.5	-1.2
-1.2	0.3	2.5	4.0	1.2	-0.3	-1.0
-1.0	-1.5	-1.7	-1.2	-1.0	-0.5	-0.2

Fig. 4. Two sets of model coefficients.



(a) (b) (c) (d) (e) (f) (g)

Fig. 5. The initial image and the six steps of a progressively halftoned image.

in Fig. 6(a)–(g). Consequently, the black dots in the images shown in Fig. 5 have the shape of the corresponding windows in Fig. 6. Furthermore, the edges in the image shown in Fig. 5(g) are enhanced due to the use of the error criterion (1). The produced image is somewhat smoother if the set of coefficients shown in Fig. 4(b) is used. However, the edges are still more enhanced than in images produced by standard error diffusion, such as the halftoned image shown for comparison purposes in Fig. 7. The quality of the produced bilevel image can be improved to an arbitrarily good level by increasing the resolution and further applying the progressive scheme in an iterative way, provided that sufficient sampling density is available.

The encoding of the generated pixel classes, using predictive arithmetic coding, was also investigated in this example. The compression ratios achieved in each step of the method, after the initial thresholding, are shown in Table II. These ratios were computed by evaluating the statistics of each configuration of a reference template around each pixel compared to the pixel value. Four pixels were used in the reference template, defining sixteen configurations. The statistics of occurrences of 0's and 1's for each of these configurations were collected as counts $N_0(i)$ and $N_1(i)$, for $i = 1, 16$, respectively. From these counts, the probabilities $P_0(i)$ and $P_1(i)$ of the current pixel being a 0 or 1 were estimated by using the proportioning equations

$$P_0(i) = \frac{N_0(i)}{N_0(i) + N_1(i)}$$

$$P_1(i) = 1 - P_0(i). \tag{12}$$

These probabilities were used in the arithmetic coding compression of the bilevel image.

In each step of the method, the four reference pixels were chosen forming a rectangular or diagonal arrangement toward the current pixel (Fig. 8), depending on

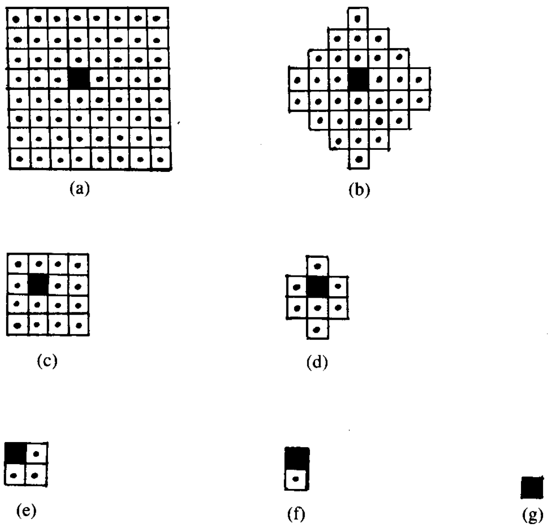


Fig. 6. The neighborhood of each pixel (m, n), which is given the same intensity $x(m, n)$, initially and in the six steps of the algorithm.

Fig. 7. A halftoned image using the standard error diffusion method.

TABLE II
THE COMPRESSION RATIOS IN EACH STEP OF THE METHOD

Steps	Compression
1	0.54
2	0.53
3	0.50
4	0.81
5	0.73
6	0.82
Total	0.75

whether a rectangular or diagonal lattice was used in that step of the halftoning process. The distance of the reference pixels from the current one, in both forms shown in

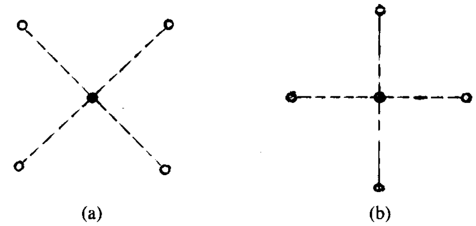


Fig. 8. The two forms of the reference templates used in the encoding technique.

TABLE III
THE COMPRESSION RATIOS IN EACH STEP OF THE BIASED IMPLEMENTATION OF THE METHOD

Steps	Compression
1	0.33
2	0.28
3	0.33
4	0.71
5	0.56
6	0.69
Total	0.62

Fig. 8, was chosen so as to include the values of the previously computed pixels. In this example, a distance of 4, 2, 1 pixels was chosen in the three corresponding steps, where each form was used. A larger compression was possible in the former steps of the method, which were dominated by the thresholding operation. However, in the latter steps, where the halftoning criterion effects were more visible, the compression ratios were rather small. For this reason, the overall rate achieved was 0.75 bits/pixel. We have also examined the biased implementation of the decision criterion, which was mentioned in Section III of the paper, using a value of $\epsilon = 0.2$ which did not seriously affect the quality of the halftoned image. Table III summarizes the compression ratios achieved in each step of the progressive method, the overall rate being 0.62 bits/pixel.

The performance of the reconstruction technique was then examined using the generated bilevel image and the set of coefficients shown in Fig. 4(a). The halftoned image shown in Fig. 5(g) was used as input to the reconstruction filter, the output of which is shown in Fig. 9(a). By comparing this image with the original image shown in Fig. 3, it can be seen that a good reconstruction of the image edges has been accomplished. However, as was earlier mentioned, a roughness is observed in the textured areas of the image, which is due to the nature of the halftoned image and the error of the approximation in (7). The mean squared error, defined as the normalized sum of the squares of differences between the original and generated image pixel values, was found to be -16.4 dB. By postprocessing the generated image by the nonlinear filter defined in (8), this roughness has been eliminated without affecting the edges severely, as shown in Figure 9(b). The

(a)

(b)

Fig. 9. The reconstructed image after the application of the: (a) linear; and (b) nonlinear filter.

mean square error was improved and was found to be -17.9 dB.

The performance of the algorithm was also examined, using the image shown in Fig. 10(a). Fig. 10(b)–(d) shows the halftoned and reconstructed images by linear and subsequent nonlinear filtering, respectively, at a rate of 0.7 bits/pixel. The mean squared errors between the reconstructed and original image were -19.4 and -21.5 dB after applying the linear and nonlinear filter, respectively.

In order to examine the performance of a combined halftoning/reconstruction procedure, the error between the original and reconstructed images, shown in Figs. 3 and 9(b), respectively, was computed and its statistics were used for coding purposes. As has already been mentioned, a deadband quantizer was designed based on a tradeoff between picture quality and compression ratio. The deadband of this quantizer ranged from -16 to 16. By zeroing

(a)

(b)

(c)

(d)

Fig. 10. Original image (a) and finally generated halftoned (b) linearly (c) and nonlinearly (d) reconstructed images.

out the values of the error image pixels which belonged in this range, we formed an error image in which only 18% of the pixels had nonzero values. A nonuniform symmetric quantization in the range $[-80, -16]$, $[16, 80]$, corresponding to a codeword of length of 4 bits, resulted in a reconstructed image which could hardly be distinguished from the original one, corresponding to a mean squared error of -24.9 dB. Using run-length followed by variable-length coding, we encoded the above positions and nonzero magnitudes of the error image at 0.54 and 0.43 bits/pixel, respectively. For this purpose, we used Huffman coding which was, for simplicity, designed based on the statistics of the specific image. A standard code could be designed, based on the statistics of several characteristic cases, if this technique was to be generally used.

V. Conclusions

A progressive halftoning technique was described, using an appropriate image distortion criterion. The technique separates the desired bilevel image pixels into different classes and computes the pixel values in each class in a simple, parallel form. A linear filter is included in this technique to permit the reconstruction of a gray-tone image from a halftoned version of it. It has been shown that postprocessing by a nonlinear filter can further improve the quality of the reconstructed image, without affecting the compression ratio achieved by the halftoning approach. The above procedure can be used in a scheme where a continuous-tone image is progressively halftoned, coded, transmitted, and possibly reconstructed. In this case, the reconstruction can be improved at any desired level by appropriately coding the reconstruction error.

REFERENCES

[1] J. Jarvis, C. Judice, and W. Ninke, "A survey of techniques for the display of continuous tone pictures on bilevel displays," *Comput. Graph. Image Process.*, vol. 5, pp. 13–40, 1976.

[2] J. Stoffel and J. Moreland, "A survey of electronic techniques for pictorial image reproduction," *IEEE Trans. Commun.*, vol. 29, pp. 1898–1924, 1981.

[3] D. Anastassiou and K. Pennington, "Digital halftoning of images," *IBM J. Res. Devel.*, vol. 26, pp. 687–697, 1982.

[4] R. Ulichney, "Dithering with blue noise," *Proc. IEEE*, vol. 76, pp. 56–80, 1988.

[5] "Progressive coding scheme for binary images," *Japan.* Prop. ISO/UTC1/SC2/W, Jan. 1988.

[6] D. Anastassiou, "Neural network based digital halftoning of images," presented at Proc. IEEE Symp. Circ. Syst., Helsinki, Finland, June 1988.

[7] S. Kollias and D. Anastassiou, "An adaptive least squares algorithm for the efficient training of artificial neural networks," *IEEE Trans. Circ. Syst.*, vol. 36, pp. 1092–1001, 1989.

[8] T. Bernard, P. Garda, A. Reichart, B. Zavidovique, and F. Devos, "Design of a halftoning integrated circuit based on analog quadratic minimization by non linear multistage switched capacitor network," in Proc. IEEE Symp. CAS, Helsinki, Finland, June 1988.

[9] J. Mannos and D. Sakrison, "The effects of a visual fidelity criterion on the encoding of images," *IEEE Trans. Inform. Theory*, vol. 20, pp. 526–536, 1974.

[10] D. Anastassiou and S. Kollias, "Progressive halftoning of images," *Electron. Lett.*, vol. 24, pp. 489–490, 1988.

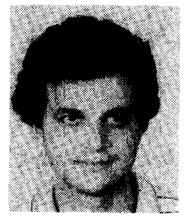
[11] D. Anastassiou, "Applications of information theory to image encoding," Ph.D. Thesis, Dept. *Elec. Eng. Comput. Sci.*, Univ. Cal., Berkeley, CA, pp. 106–108, 1979.

[12] J. Mitchell and W. Pennebaker, "Software implementation of the Q-coder," IBM Res. Rep. RC 12660, 1987.

[13] C. Langdon and J. Rissanen, "Compression of black-white images with artihmetic coding," *IEEE Trans. Commun.*, vol. 29, pp. 858–867, 1981.

[14] T.-C. Tsai and D. Anastassiou, "Stable non-linear filters for edge-enhancing noise-smoothing of images," in *Proc. ISCAS-88*, Helsinki, Finland, June 1988.

[15] H. Gharavi and A. Tabatabai, "Subband coding of monochrome and color images," *IEEE Trans. Circ. Syst.*, vol. 35, pp. 207–214, 1988.



Stefanos D. Kollias (S'81–M'85) was born in Athens, Greece, in 1956. He received the Diploma degree in electrical engineering from the National Technical University of Athens, Athens, Greece, in 1979, the M.Sc. degree in communication engineering from the University of Manchester, Manchester, England, in 1980, and the Ph.D. degree in electrical engineering from the Computer Science Division of NTUA in 1984.

Since 1986, he has been Assistant Professor in the Computer Science Division of the Department of Electrical Engineering, NTUA. From 1987 to 1988, he was a Visiting Research Scientist in the Department of Electrical Engineering and the Center for Telecommunications Research, Columbia University, New York, on leave from NTUA. His interests include digital signal processing, image and video coding and processing, HDTV coding, image analysis, parallel architectures, and artificial neural networks. He received, as a graduate student/teaching assistant in NTUA, a ComSoc scholarship from the IEEE Communications Society in 1982.

Dimitris Anastassiou (S'77–M'78) received the Diploma in electrical engineering from the National Technical University of Athens in 1974, and the M.S. and Ph.D. degrees in electrical engineering from the University of California, Berkeley, in 1975 and 1979, respectively.

From 1978 to 1983, he worked at the IBM Thomas J. Watson Research Center, Yorktown Heights, NY, as a Research Staff Member, doing research on various aspects of image processing and coding. Since 1983, he has been with the Department of Electrical Engineering of Columbia University, New York, NY, where he is currently an Associate Professor. His research interests include high-definition television, digital image processing and coding packet video communications, digital signal processing, and VLSI architectures for image processing.

Dr. Anastassiou has received an IBM Outstanding Innovation Award for his contributions to the development of the IBM videoconferencing system, and a 1985 NSF Presidential Young Investigator Award.