Digital halftoning on the IBM 4250 Printer

by Gerald Goertzel
Gerhard R. Thompson

A method of reproducing high-quality continuous-tone images via the IBM 4250 Printer is presented. The approach is modeled on the halftone process used in conventional lithography but is adapted to discrete bilevel printing and digital processing. We use a combination of standard and novel techniques. These include generation and calibration of a set of halftone dot patterns, randomized error propagation in pattern selection, and resolution enhancement in areas of high intensity gradients. The resultant images have good gray-scale rendition and sharp edges without the problems of contouring and worminess often associated with digital image reproduction. We have named this approach PREPRESS: Picture Rendition using Error Propagation and Resolution Enhancement in Simulated Screening.

Introduction

This paper describes an algorithm for the reproduction of high-quality continuous-tone images via the IBM 4250 Printer [1]. The IBM 4250 Printer (see also the Appendix) is a high-quality text and line-art printer which prints on special metal-coated paper. It was designed to produce camera-ready copies for the generation of offset plates. At the Print '85 Graphic Arts Trade Show in Chicago, IBM demonstrated a new metal-coated plastic medium as a future technology that can be used with the 4250 Printer. With this new medium the 4250 can directly produce a negative for making offset plates, or directly produce the plate itself for short-run lithographic printing. The new medium eliminates the photographic steps normally associated with platemaking.

A high-quality digital halftoning procedure was considered essential for the 4250 and other bilevel printers. In particular, it would demonstrate the potential of the 4250 in composing pages containing text, graphics, and continuous-tone images. The steps involved in developing such a procedure are described here.

The first part of the paper is concerned with the technique for digital halftoning, the second with image preprocessing. The third part gives the details of various portions of the process.

Digital halftoning

In conventional lithography, continuous-tone images are printed by a halftoning process. The conventional process is described in numerous publications, e.g., [2, 3]. Briefly, an original is exposed through a screen which converts the image into dots of various sizes. Larger dots are associated with darker areas and smaller dots with lighter areas. Halftone screens are characterized by the number of screen lines per inch. Screens typically run from 65 to 300 lines per inch. For black-white printing the screens are turned so that the lines make a 45-degree angle with the edges of the page.

For a printer such as the 4250, in which the basic dot size cannot be modified, a digital process is used. As a first step, the original is scanned, and digitized gray values are stored in a computer. We are concerned with the remaining steps, i.e., creating high-quality printed images from such stored documents.

Quality problems associated with digital halftoning are discussed in [4, 5], where a brief survey of existing approaches is also given. Our approach is modeled on the halftone process used in conventional lithography. Thus we associate areas of the 4250 output, in which halftone dots are generated, with pixels of the scanned document [6]. As in the conventional case, our digital halftone blocks are arranged on a 45-degree grid. This implies that each such block should contain $2n^2$ pels, where n is an integer. Increasing n lowers the resolution of the output image, whereas decreasing n reduces the choice of output patterns and corresponding density levels.

We have found $n = 3$ to be a satisfactory compromise, corresponding to 18 pels on the output image for each digital halftone block. (A block of this size yields a halftone screen of 141 lines per inch, which matches fairly well the 150 lines per inch used in many magazines.) These blocks may be of any shape that contains the 18 pels and tiles the plane with the desired 45-degree symmetry. A rectangular 3 by 6 block is convenient for digital processing. The blocks are shown superimposed on the output pels in **Figure 1.**

To maintain the 45-degree symmetry for the input document, we associate two adjacent input pixels with each halftone block, as is shown in **Figure 2.** With this association of input pixels with blocks, an image scanned at 200 pixels per inch prints full size on the 4250, which has 600 pels per inch.

We have chosen a set of halftone patterns containing 0 to 18 black pels; each pattern is obtained from the previous pattern by adding one pel. The set of patterns (omitting the pattern for an all-white block) is shown in **Figure 3.**

Once the patterns are defined, they may be used to represent continuous-tone images, provided that a gray value can be associated with each pattern. A simplistic assumption is that the blackness of a pattern is proportional to the number of black pels in the pattern. We give the results of printing three images with this assumption in **Figure 7,** shown later. In the printed images, the pattern used for each block is that with a gray value closest to the average of the gray values of the two input pixels associated with that block. The first input image is a gray wedge, uniformly and continuously decreasing in reflectivity from left to right. The output clearly is neither uniform nor continuous. The gray values are also greatly distorted in the woman and in the building.

One reason that the apparent gray values are not proportional to the number of black pels is the overlap of the pels when printed. For the 4250 Printer, each printed pel has a diameter approximately twice the pel spacing. One possible method of calibrating the patterns would be by calculating the area occupied by black pels, taking overlap into account, and making the assumption that the blackness is proportional to this area. Such calculations have been described in [7]. Applying the results of this model yields some improvement but is incorrect because the shape of the printed pels is irregular and is dependent on adjacent printed pels.

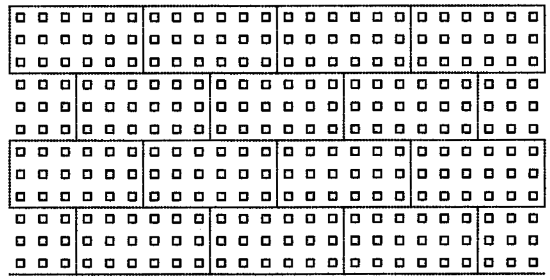

Figure 1

Output pel locations in relation to halftone blocks.

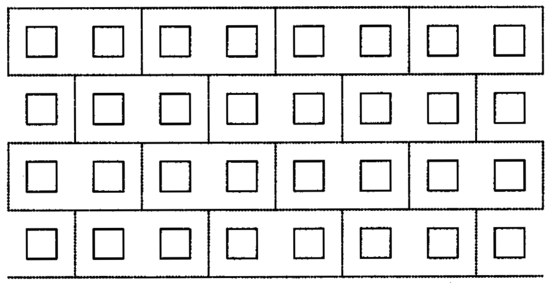

Figure 2

Input pixel locations in relation to halftone blocks.

A better approach is to print a region filled with a given pattern and to measure its reflectivity [5]. This is repeated for each pattern in turn. A linear transformation of the reflectivities gives the gray values; the lightest pattern (no black pels) is given the value 0, and the darkest pattern (all black pels) is given the value 255.

The calibration should be done by printing on the final output medium, i.e., on paper using an offset press. If the 4250 Printer output on aluminized paper is used for proofing, these proofs should be printed with a calibration obtained by measuring the aluminized paper. The calibrations will differ, primarily because of the dot gain in the printing process.

The eighteen patterns (omitting the pattern with no black pels) are used in succession to print the chart shown in **Figure 4. Figure 5** contains a photomicrograph from each halftoned area of Figure 4. The gray values associated with each halftone pattern are given in **Table 1** both for the proof on aluminized paper and for the output of an offset press.

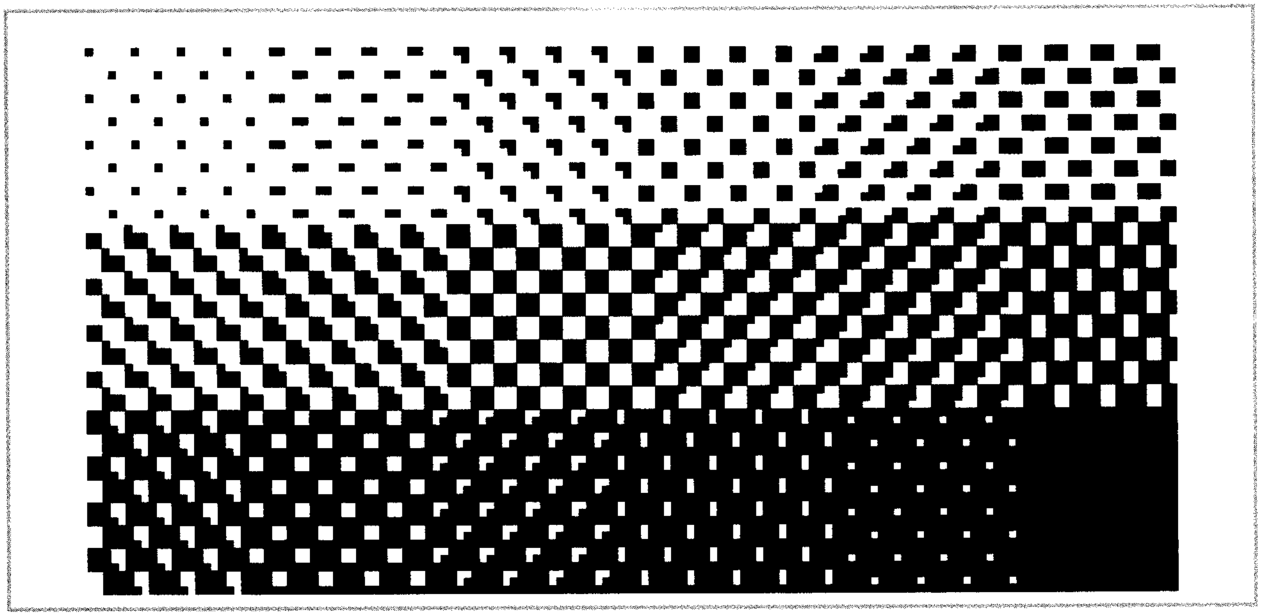

Figure 3

Set of 18 enlarged halftone patterns with no pel overlap.

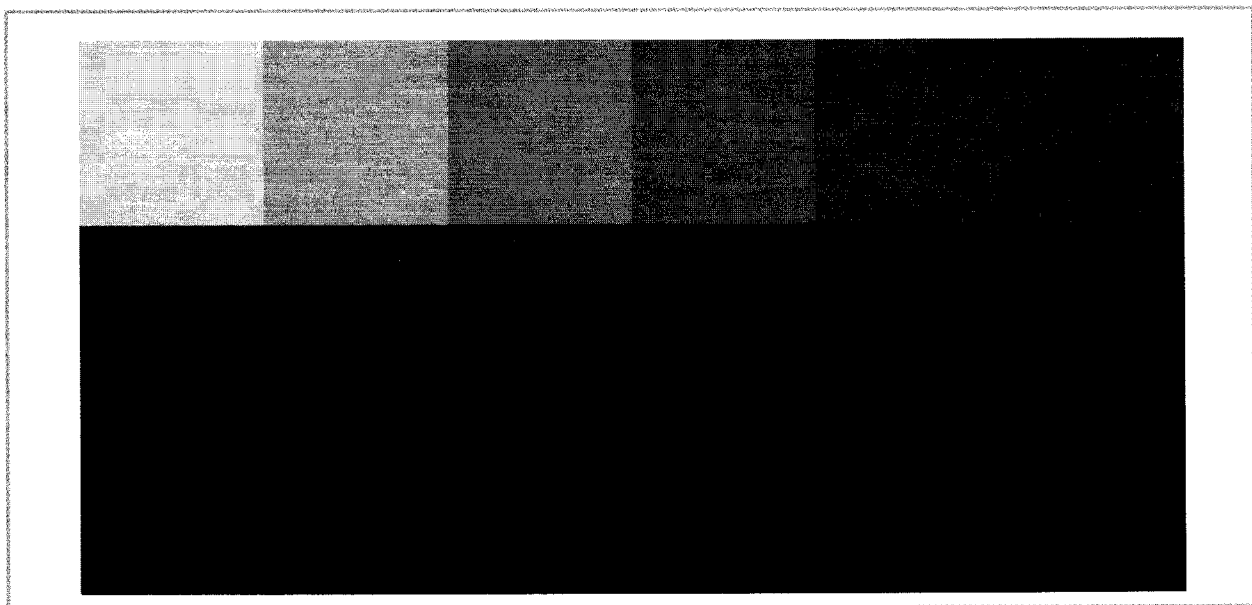

Figure 4

Chart showing 18 halftoned areas, each printed with one halftone pattern.

Using the calibrated gray values instead of gray values proportional to the number of pels printed produces the results presented in **Figure 8,** shown later. In the printed images, the pattern used for each block is that with a gray value closest to the average of the gray values of the two input pixels associated with that block. Note the limited

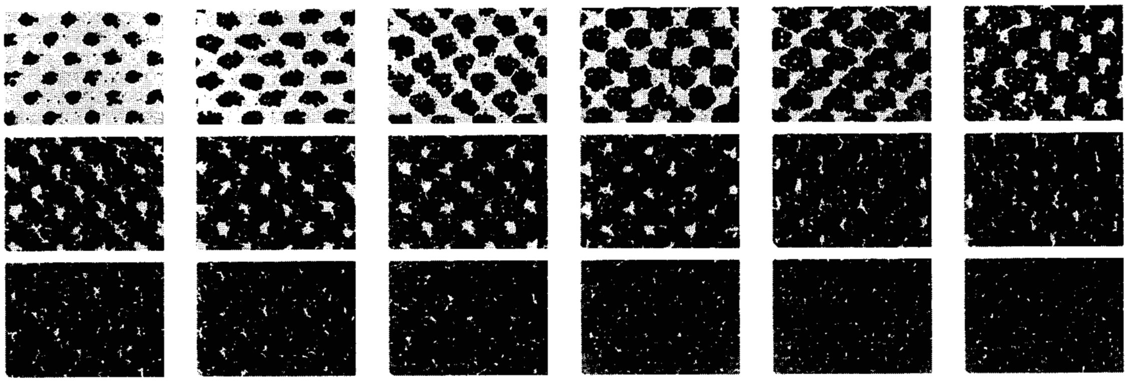

Figure 5

Photomicrographs of the 18 halftone patterns as printed by the IBM 4250 Printer.

number of gray levels and the contouring in these figures. This is a well-known problem associated with digital halftoning. The conventional solution has been either to increase the size of the halftone block, thus increasing the number of available gray levels, or to abandon halftone blocks and use the "error diffusion" halftoning technique originally described in [8].

We chose instead to eliminate contouring by using error diffusion (also known as error propagation) between adjacent halftone blocks. This is similar to a method described in [9]. If an area is printed using two neighboring halftone patterns intermixed, the eye will perceive the mean gray value. Thus, when the proportions of the two patterns are varied, any gray value from that of the lighter pattern to that of the darker pattern may be achieved. One criterion for pattern selection is that such intermixing of patterns should yield pleasing image areas, without generating distracting artifacts or artificial patterns in the image.

To achieve the desired average gray values using error propagation, we associate an error value with each output block. These values are initially all zero. The output patterns are selected by a scanning process; the output is constructed for the first line from left to right, then for the second line from left to right, and then for succeeding lines in order until the entire output is defined.

Usually the desired gray value of a block lies between the gray values of two patterns in the table. The pattern with the lower gray value is selected for output and the difference between the gray value of the pattern and the desired gray value is the error. This error is propagated to the adjacent block and to two blocks in the next line, as shown in **Figure 6.** x is the fraction of the error propagated to the right. The remainder of the error is propagated to the two blocks below.

Table 1 Gray values associated with the eighteen halftone patterns of Figure 4.

Pattern number	*Proof*		*Press*	
	Percent reflectivity	*Gray value*	*Percent reflectivity*	*Gray value*
0	72.8	0	87.9	0
1	59.4	56	52.9	110
2	51.7	88	35.6	165
3	44.0	120	24.5	200
4	39.1	141	19.3	216
5	34.1	162	14.7	231
6	28.8	184	12.6	238
7	24.9	200	10.8	243
8	22.5	210	10.1	246
9	22.1	212	9.9	246
10	19.9	221	8.6	250
11	17.4	232	7.6	253
12	16.5	235	7.4	254
13	14.6	243	7.3	254
14	14.3	245	7.2	255
15	13.1	250	7.1	255
16	12.2	253	7.1	255
17	12.0	254	7.1	255
18	11.8	255	7.1	255

When blocks are processed, the desired gray value is that obtained by adding the input to the propagated error.

As a first illustration of error propagation between blocks, observe the uniform gray wedge and the two standard examples printed when one fourth of the error is propagated to the right and the remaining error is divided equally and propagated to the two blocks on the next line ($x = 0.25$). These images are given in **Figure 9.** A close inspection of this figure shows a problem with error propagation, especially in

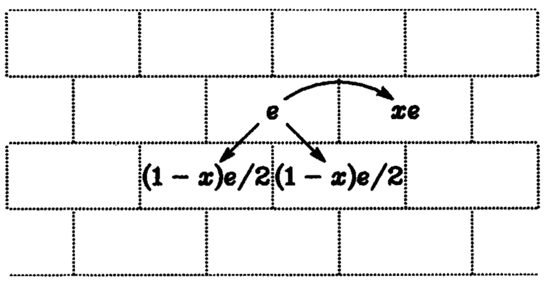

Figure 6

Error propagation between halftone blocks.

Figure 7

Standard image set assuming nonoverlapping pels.

the light areas. Some unwanted patterns appear. A clearer illustration of the problem is seen in **Figure 10** in an area of uniform gray with a gray value of 20. Note the long horizontal alignment of dots near the top of the box and vertical and other horizontal alignments elsewhere.

This problem of unwanted patterns may be reduced by distributing some of the error randomly; x is set equal to a pseudorandom number, equidistributed on the interval 0 to 0.5. The result of this change is shown in **Figures 11** and **12.** The unwanted patterns are reduced.

The use of random numbers to eliminate unwanted patterns has been suggested by Dietz and Juels [5]. They use the random number to determine in which direction to propagate all of the error. The use of random numbers to determine the fraction of the error propagated in a given direction has been analyzed by Ulichney [10], who discusses many distribution paths and weighting schemes for distributing errors to neighboring pel positions. Our use of random numbers is similar but is applied to blocks of pels forming a halftone screen rather than to individual pels.

• *Resolution enhancement*

The resolution of the output images shown thus far is limited by the use of the 3 by 6 halftone blocks and by the input scanning resolution of 200 pixels per inch. In photographic halftoning using contact screens, resolution is not limited to the halftone dot spacing, since both the dot shape and dot center location can vary. This effect, known as partial dots [11], is seen at edges where the gray value changes by a large amount. Dots are altered to permit a more precise and clearer rendition of detail. In digital halftoning, higher resolution may be obtained by controlling the printing of pels individually rather than in blocks. Where a one-to-one correspondence exists between input pixels and output pels, this is readily accomplished. High-resolution input data are available to control the printing of each corresponding output pel. However, in our case we are scanning at 200 pixels per inch and printing at 600 pels per inch; thus there are nine output pels for each input pixel. High-resolution input data are not available at edges but must be derived; hence we call the technique for determining which individual pels to print in an edge area *resolution enhancement.* (This technique is intended to improve the

Figure 8

Standard image set using calibrated halftone patterns.

Figure 9

Standard image set using calibrated gray values and error propagation with $x = 0.25$.

resolution at edges, thus decreasing staircasing. It is not intended to emphasize the edges. Edge emphasis is discussed later in this paper and in [5].)

At edges where the gray value changes by a large amount, resolution is more important than gray-tone accuracy. A pixel in this region is treated not as part of a halftone block but as an edge pixel. First the number of black pels in the half-block is determined from its gray value so as to obtain approximately the right amount of gray for the half-block. The 3 by 3 block of pixels with the edge pixel at the center is thresholded to give this desired number of black pels. The resultant 3 by 3 binary array is used to control the printing

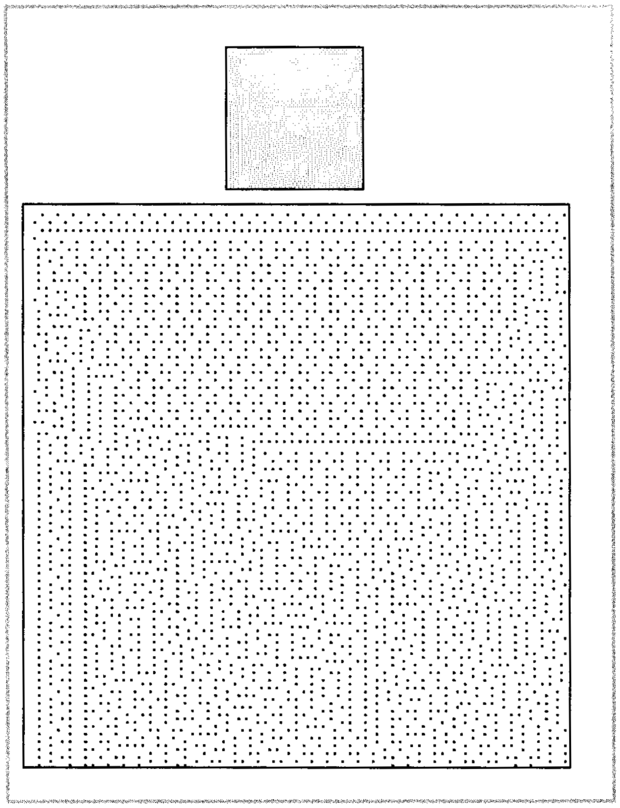

Figure 10

Area of uniform gray and a 4× enlargement illustrating horizontal pattern alignment caused by error propagation with $x = 0.25$.

of the 3-by-3-pel region in the center pixel. The result of this resolution enhancement is shown in **Figure 13.**

Preprocessing

The halftoning process we have described assumes that the scanned images are represented as gray levels from 0 to 255, with 0 representing white and 255 representing black. If these assumptions are not applicable, a first step of preprocessing is to modify the input gray levels to make the assumptions true. This is accomplished by a linear mapping of the input gray levels onto the range 0–255.

A second step consists in modifying the contrast and brightness of the image, if so desired. As an example, **Figure 14** contains an image with two different contrasts and two different brightnesses. The mapping functions used to modify these images are shown in **Figure 15.** They are explained in more detail later.

In addition to the range, contrast, and brightness corrections, a Laplacian filter which enhances high-spatial-frequency components in the image often improves the output [4]. Applying Laplacian edge enhancement to the standard images prior to halftoning with resolution enhancement results in **Figure 16.** Almost everyone prefers the enhanced image of the building; many prefer the unenhanced image of the woman.

Detailed description of algorithms

A program has been constructed to carry out the halftoning procedure described above. The halftoning procedure will

Figure 11

Standard image set using calibrated gray values and pseudorandom error propagation ($0 \leq x \leq 0.5$).

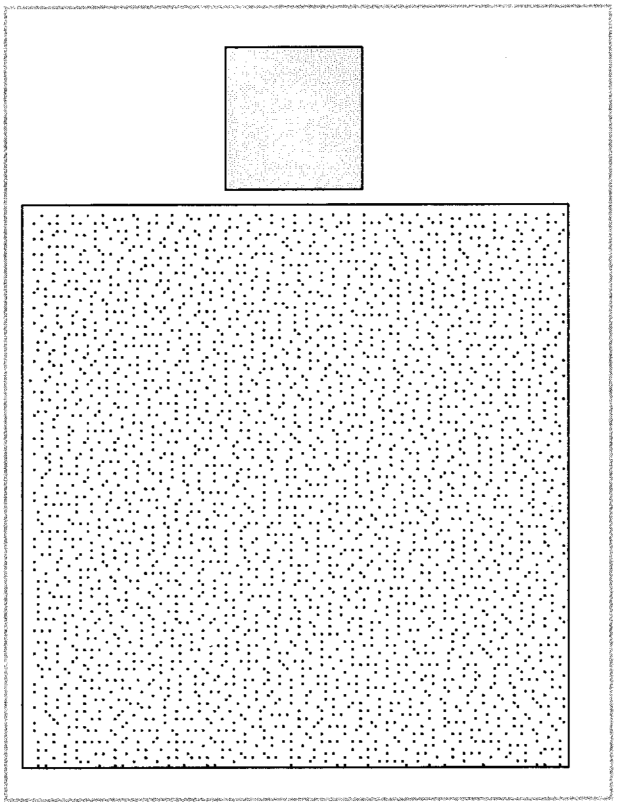

Figure 12

Area of uniform gray and a 4× enlargement with pseudorandom error propagation ($0 \leq x \leq 0.5$).

convert each pair of scanned input pixels to a 3 by 6 set of output pels. This produces images of the same size for the case in which the resolution of the output printer is three times the resolution of the input scanner. For the 4250 Printer, this requires that the input be scanned at 200 pixels per inch for reproductions of the same size.

If the generated output image is not the desired size, the input gray-level image should be scaled. Scaling a halftoned image as a bitmap, without distorting the patterns and the gray values, is a difficult problem with no known solution.

In this part of the paper we describe sequentially, and in some detail, the various steps in the halftoning process. The input is a gray-level image, the output a bitmap ready for printing. The steps of the procedure are as follows:

1. Range adjustment of input gray levels.
2. Contrast and brightness adjustment.
3. Laplacian edge enhancement (optional).
4. Halftoning.
   a. Characterization of a pixel pair.
   b. Processing pixels not on edges.
   c. Error propagation for pixels not on edges.
   d. Processing edge pixels.
   e. Error propagation for edge pixels.

• *Range adjustment of input gray levels*

Scanners may produce data sets in which the range of gray values is less than 0 to 255. Also the low values may correspond to either dark or light areas. In this step, the input gray values are mapped linearly on the range 0 to 255, with 0 corresponding to white and 255 to black. This mapping is characterized by two numbers: the value to be mapped onto 0 and the value to be mapped onto 255. Input

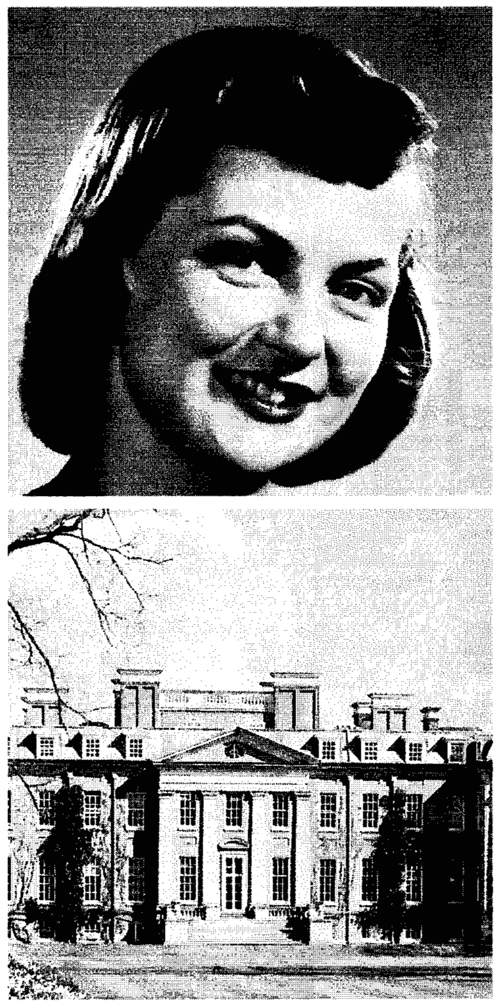

Figure 13

Standard image set with resolution enhancement.

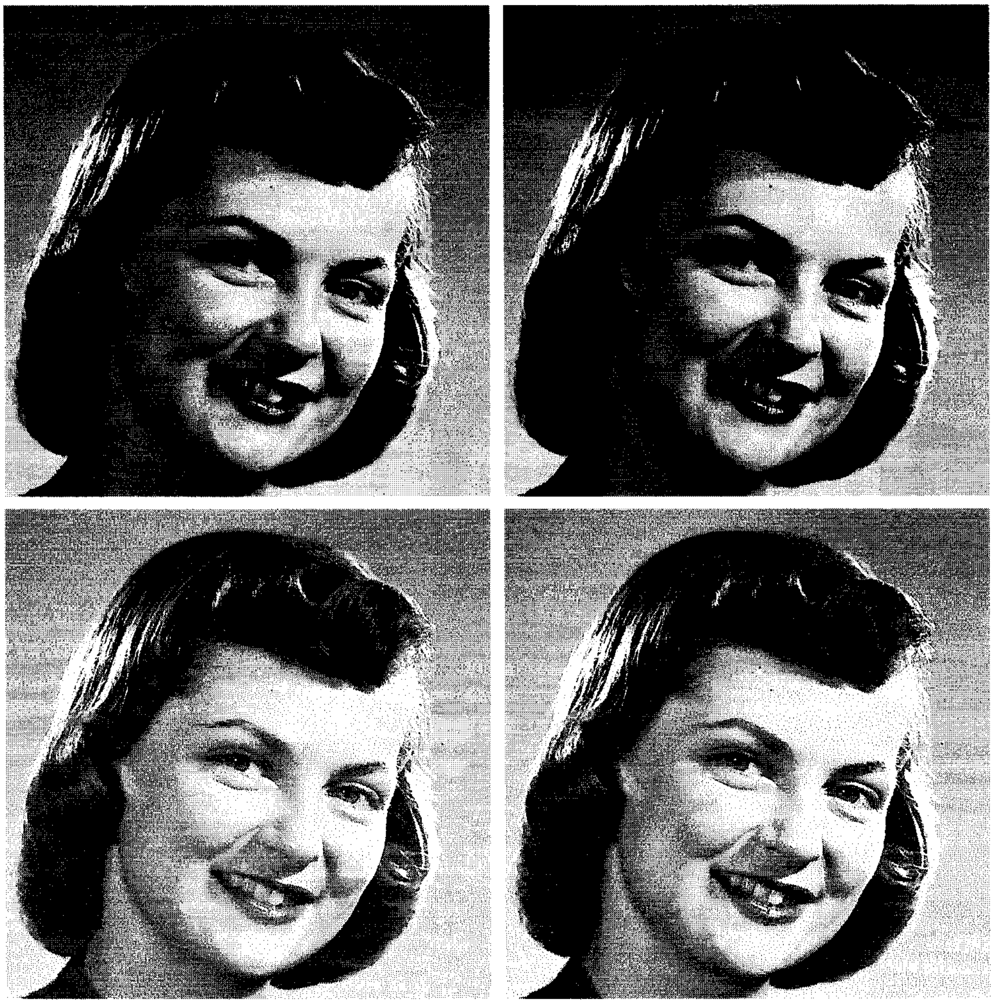

Figure 14

Example of contrast and brightness change. Contrast increases to right, brightness going down.

values mapped below 0 are set to 0. Input values mapped above 255 are set to 255. An example of one such mapping is illustrated in **Figure 17.**

• *Adjustment of contrast and brightness*
The contrast and brightness of the gray-level input image are adjusted by a second mapping. This mapping is described in terms of a normalized input data range (after the above range adjustment) between 0 and 1, where 0 represents 0 percent print-area coverage and 1 represents 100 percent print-area coverage. The output data range is also normalized between 0 and 1. The mapping is specified by a curve with three regions. The central region is a straight line which is specified by a slope and a point on the line. The slope determines the change in contrast produced by the mapping. A slope greater than 1 increases the contrast; a slope less than 1 decreases the contrast. Similarly the point determines the change in brightness. The point is described by specifying the normalized input gray value which is to yield a normalized output gray value of 0.5.

Outside the linear region, the transformation is a parabolic curve from the linear portion to the (0, 0) or (1, 1) corner of the unit square [12]. The slope of the parabolic curve at the interface is equal to the slope of the linear portion. The linear portion of the mapping curve is terminated by reaching a border region, which is the region between the unit square and a smaller concentric square. We usually take this smaller square to have a side of 0.8.

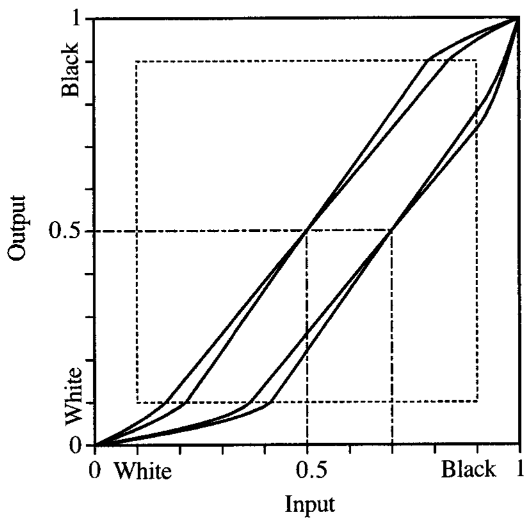

Figure 15

Transformations used for images in Figure 14.

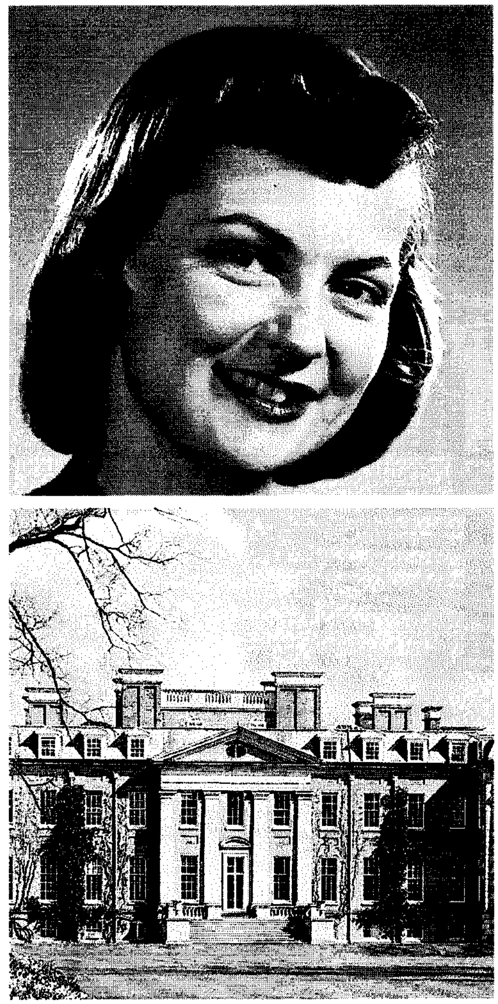

Figure 16

Standard image set with added Laplacian edge enhancement.

A sample mapping is shown in **Figure 18.** The four different mappings in Figure 15 yielded the four images in Figure 14.

• *Laplacian edge enhancement*
Laplacian edge enhancement consists in replacing the image intensity with the difference of the image intensity and some positive multiple of the Laplacian. Intensity peaks have a negative Laplacian and intensity valleys have a positive Laplacian. Subtracting a multiple of the Laplacian from the intensity emphasizes variations in intensity.

Consider the 3-by-3-pixel area shown in **Figure 19.** We compute an estimate of the Laplacian at the center pixel e. Within a multiplicative factor, the Laplacian is given by

$$L = \left(-e + \frac{b + d + f + h}{4}\right).$$

Subtracting a multiple of L from e yields, for the center pixel,

$$e - CL = (1 + C)e - C\,\frac{b + d + f + h}{4}.$$

Thus, if Laplacian edge enhancement is desired, $e - CL$ replaces the gray value e for each pixel in the image. We have used $C = 2$ in our programs. The effect of this enhancement was shown in Figure 16.

• *Halftoning*
The halftoning process is applied line by line to the input image, scanning from left to right. The input pixels are paired as shown in Figure 2, with each pair producing 18 output pels, as shown in Figure 1. Each input line produces three output lines.

Characterization of a pixel pair Each pixel can be processed in two possible ways: as half of a halftone block or as an edge pixel (a pixel on an edge). For each pair of pixels, four mutually exclusive cases are distinguished:

- Neither pixel is an edge pixel.
- The left pixel is an edge pixel.
- The right pixel is an edge pixel.
- Both pixels are edge pixels.

To determine whether a pixel is an edge pixel, we calculate an estimate of the magnitude of the intensity

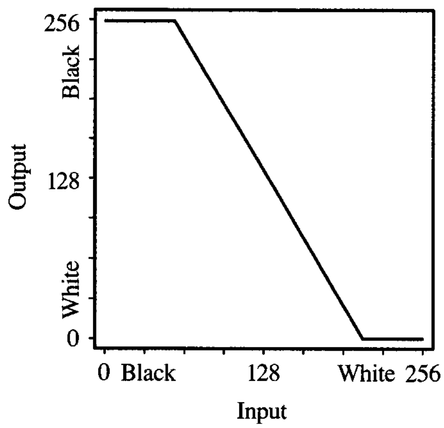

Figure 17

Example of range adjustment.

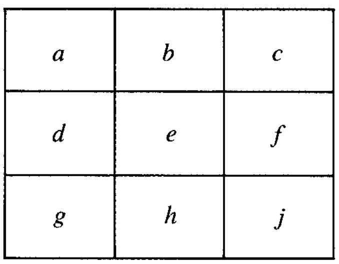

Figure 19

3 × 3 area of input image where $a, b, c, \cdots j$ are the gray values of the corresponding pixels.

gradient [13] at the pixel. Referring to Figure 19,

$$G^2 = (a - j)^2 + (c - g)^2,$$

where G is proportional to the intensity gradient. We have obtained good results using the following empirical criterion: If $G^2 \geq 100^2$ the pixel is an edge pixel; otherwise it is not.

The treatment of edge pixels is described below in the subsection entitled *Processing edge pixels.* If only one pixel in a pair is an edge pixel, the other is treated as half a pair of identical pixels and handled as described below for pixels not on edges. The appropriate half of the block appears on the output image.

Processing pixels not on edges These pixel pairs are used to select the appropriate halftone pattern from the set shown

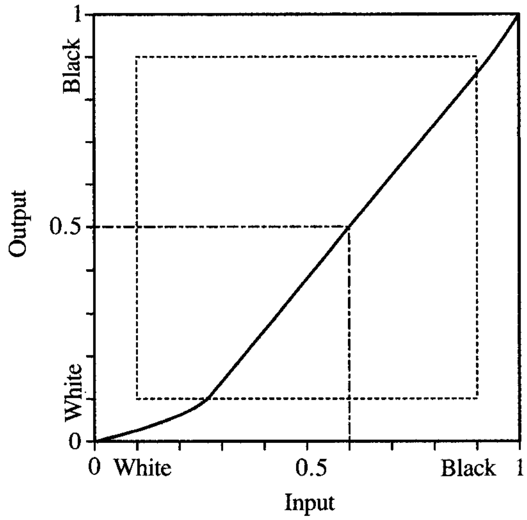

Figure 18

Adjustment of contrast and brightness.

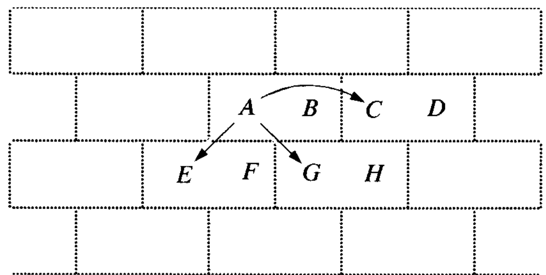

Figure 20

Pixel pairs and corresponding gray values.

in Figure 3. Gray values of the patterns are required. The measured values used in our program are given in Table 1.

The pixel pair (A, B) has the gray values A and B; see **Figure 20.** The gray value associated with a pattern number n is designated $g(n)$. The error associated with a pixel pair (X, Y) is designated $e(X)$. Errors on a given line were propagated from the present and the previous line when they were processed. The errors are taken as zero for the first line.

The steps of the algorithm for processing the pixel pair (A, B) are as follows:

1. Calculate the desired gray value d:

$$d = \frac{A + B}{2}.$$

Table 2 Gray value vs. number of selected pels for printing edge pixels.

Gray value	*Number of pels*
0–27	0
28–83	1
84–139	2
140–167	3
168–195	4
196–223	5
224–251	7
252–255	9

2. Find the trial pattern number n such that d lies at or above the gray value for n and below the gray value for $n + 1$:

$$g(n) \le d < g(n + 1).$$

3. If the error added to the desired gray value is greater than $g(n + 1)$, increase n by 1. That is, if $d + e(A) \ge g(n + 1)$, $n = n + 1$.
4. If the pattern number n is greater than 18, set it to 18.

The pattern just determined is placed in the output image. Note that the propagated error is always positive and causes an increase in the trial pattern number by at most one. This prevents unwanted artifacts in the image caused by the intermixing of nonsequential patterns in uniform gray areas.

Error propagation for pixels not on edges Let p designate the error to be propagated from pixel pair (A, B). Using the definitions for e, d, and g given above,

$$p = e(A) + d - g(n).$$

Distribute the error in accordance with the arrows in Figure 20:

$$e(C) \leftarrow e(C) + xp,$$

$$e(E) \leftarrow e(E) + \frac{(1 - x)p}{2},$$

$$e(G) \leftarrow \frac{(1 - x)p}{2},$$

where x is a pseudorandom number equidistributed on the range 0 to 0.5.

Processing edge pixels The appropriate number of pels, m, selected for printing an edge pixel (i.e., a half-block) is based on **Table 2.**

The selection process is explained below.

We consider the 3 by 3 square of pixels with the edge pixel at the center. Our edge-enhancement algorithm requires that the m pixels with the largest gray values be selected. This is a thresholding operation producing a 3 by 3 binary array containing m selected positions corresponding to the m largest gray values. The binary array is then used to determine which pels in the 3 by 3 half-block of the edge pixel are made black. This is illustrated in **Figure 21,** where the center pixel covers a light/dark boundary and has a gray value of 200. The intensity gradient indicates that this is an edge pixel, so from table lookup $m = 5$. Therefore, the five darkest pixels are found, and the corresponding pels in the edge pixel are selected for printing. However, because of the dot gain in the printing process, only alternately selected pels are made black. If all of the selected pels were made black, edges would stand out unnaturally. This half-block is placed in the output image.

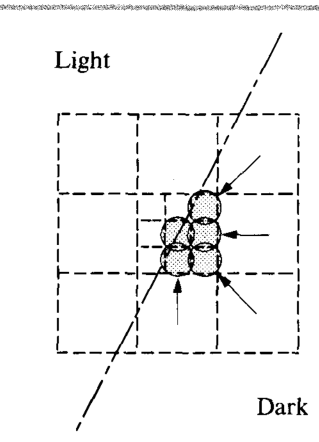

Figure 21

Placement of black pels in an edge pixel.

Error propagation for edge pixels For a pixel pair containing one or two edge pixels, the error down to the right is reset to $0.5g(1)$.

Conclusions

We have presented a digital halftoning algorithm which was specifically designed for the IBM 4250 Printer but which has application to other printers as well. We call this algorithm the PREPRESS algorithm. Key features of the algorithm are

- Simulation of the halftone screening process used in offset printing.
- Calibration of the printing process by measuring reflectance of a printed test chart.
- Randomized error propagation to smooth transitions between halftone patterns and eliminate contouring without producing unwanted patterns.
- Resolution enhancement for areas of high density gradient.

Appendix: The IBM 4250 Printer

Electroerosion printing creates text and images on a special medium by selectively vaporizing a shiny metallic film to

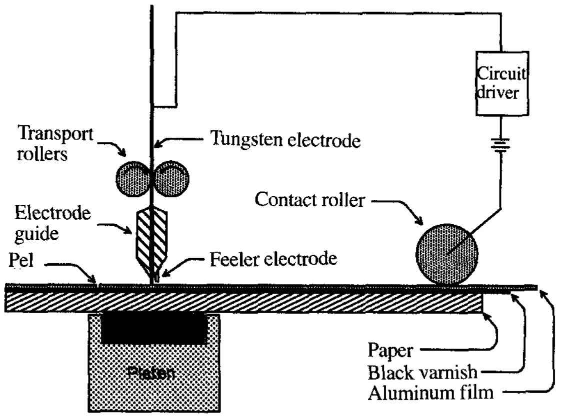

Figure A1

Method of printing.

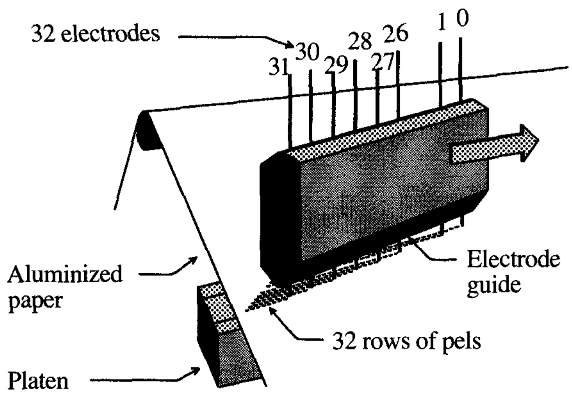

Figure A2

Electrode configuration.

reveal a dark contrast layer or clear undercoat, depending on the medium being used. The paper medium may be used for proof or camera-ready copy. The clear plastic medium may be used for negatives or short-run plates.

As depicted in **Figure A1,** the IBM 4250 prints on a medium with tungsten electrodes that are thinner than human hair. A guide holds the electrodes in contact with the aluminum film. The electrodes and the aluminum form part of an electrical circuit which, when activated, results in a brief but intense current flow that vaporizes the aluminum directly beneath the electrode to reveal the varnish underneath. This small, round black spot is called a picture element, or "pel," and is the basic unit with which the images of characters and graphics are formed.

The IBM 4250 Printer uses 32 electrodes held in a guide which moves horizontally across the medium, writing 32 rows of pels with each sweep. The guide is carried at a slight

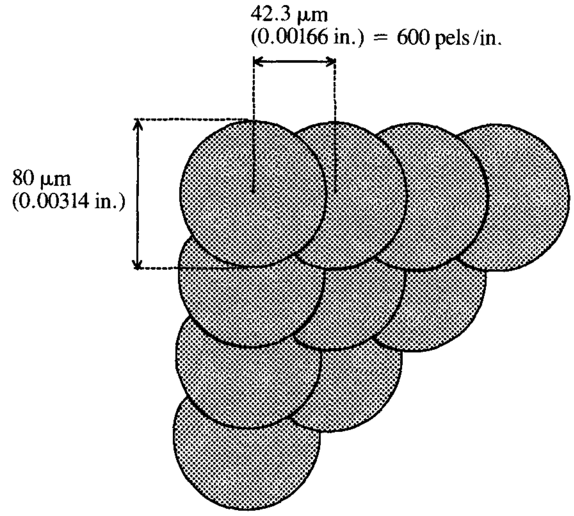

Figure A3

Pel characteristics.

angle to its direction of travel so that the rows of pels are spaced 1/600 inch apart. See **Figure A2.**

As the guide moves across the medium, the electrodes are fired so that the pels in each row are also spaced 1/600 inch apart. This gives the printer a resolution of 600 pels per inch, both vertically and horizontally (see **Figure A3**).

The pels created by the IBM 4250 Printer are round and more than 1/600 inch in diameter. Consecutive pels overlap considerably, resulting in very smooth edges on printed characters, rules, and graphics.

The aluminum surface of each medium is rather abrasive; because the tungsten electrodes are actually held in contact with this surface as the guide moves across the medium, they are gradually worn away. The guide carries a special "feeler" electrode which measures electrode wear. As the writing electrodes are worn away, they are automatically advanced to maintain optimum length.

Acknowledgments

We wish to thank Keith Pennington for motivating the development of this digital halftoning algorithm and to thank various other members of our department who have shown great patience when frequently asked to judge our halftone images. Their continuous enthusiasm spurred our efforts. Among this group are Bill Pennebaker, Joan Mitchell, Jack Lee, and Fred Mintzer. We are also indebted to various members of the electroerosion group, including Mitch Cohen, Ali Afzali, Mukesh Desai, and Eva Simonyi. Ray Agolli of Photographic Services was helpful, willing, and enthusiastic when printing our many trial halftones. The Appendix was adapted from material supplied by Udo Maier and Russ Rice of IBM Boulder. The images in the Appendix

were abstracted from that material. Carol Thompson rendered invaluable assistance in text processing.

The house used in our examples is Hursley House, which is the mansion around which the IBM development labs at Hursley, England, were built. It was supplied as a scanned image by IBM Hursley. The image of the woman used in our examples was scanned from the *IEEE Std 167A-1980 Facsimile Test Chart.*

References and notes

1. *IBM 4250 Printer Component Description and Programming Information*, Order No. GA33-1554, July 1983; available through IBM branch offices.
2. *Pocket Pal—A Graphic Arts Production Handbook*, International Paper Company, New York, 1983.
3. *The Lithographers Manual*, Raymond N. Blair, Ed., The Graphic Arts Technical Foundation, Inc., Pittsburgh, PA, 1983.
4. Dimitris Anastassiou and Keith S. Pennington, "Digital Halftoning of Images," *IBM J. Res. Develop.* **26**, No. 6, 687–697 (November 1982).
5. Henry Gordon Dietz and Ronald J. Juels, "Digital Halftone Techniques for Microcomputers," *Microcomputer Applications in Medicine and Bioengineering* (ISMM—International Society for Mini and Microcomputers), Acta Press, Anaheim, CA, 1984, pp. 89–93.
6. In this paper we use the term *pixel* for the digital representation of a scanned picture element with a possible gray value from 0 to 255. We use the term *pel* for a bilevel picture element with a value of 0 or 1.
7. P. Stucki, "MECCA—A Multiple-Error Correction Computation Algorithm for Bilevel Image Hardcopy Reproduction," *Research Report RZ-1060*, IBM Research Laboratory, Zurich, Switzerland, 1981.
8. Robert Floyd and Louis Steinberg, "An Adaptive Algorithm for Spatial Gray Scale," *1975 SID International Symposium, Digest of Technical Papers*, pp. 36–37.
9. Yi-Hsin Chen, Frederick C. Mintzer, and Keith S. Pennington, "PANDA: Processing Algorithm for Noncoded Document Acquisition," *IBM J. Res. Develop.* **31**, No. 1, 32–43 (1987, this issue).
10. Robert Ulichney, "Digital Halftoning," Ph.D. Thesis, Massachusetts Institute of Technology, Cambridge, 1986.
11. Paul G. Roetling, "Analysis of Detail and Spurious Signals in Halftone Images," *J. Appl. Photog. Eng.* **3**, No. 1, 12–17 (Winter 1977).
12. We do not know whether a straight line in place of the parabola would produce adequate results. We preferred a transformation without corners.
13. The use of the gradient to detect edges is found, for example, in Reference [14], p. 85.
14. Azriel Rosenfeld and Avinash C. Kak, *Digital Picture Processing*, Academic Press, Inc., New York, 1982.

Received December 9, 1985; accepted for publication July 21, 1986

Gerald Goertzel *IBM Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598.* Mr. Goertzel joined the IBM Advanced Systems Development Division in 1964. He transferred to the Research Division in 1973. In the Advanced Systems Development Division he managed a project to develop a Clinical Decision Support System and also contributed to the design of a small computer which was extensively used in channels and large devices. At Research he led an effort to create a high-level computer design system and worked in the fields of data compression and image processing. He devised and helped evolve a program-development system of particular use in developing efficient portable code for image processing. He has also participated in the development of a halftoning process to create plates for the printing of continuous-tone black and white images on an offset press from scanned and digitized data. His present activities are concerned with the printing of color images. Prior to his joining IBM, Mr. Goertzel worked at Republic Aviation (1941), taught aeronautical engineering at New York University (1942–1943), and worked on the Manhattan Project (1944) and at Oak Ridge National Laboratory (1947–1948). He taught physics at New York University from 1948 to 1953. During this period he helped found, consulted for, and was a director of the Nuclear Development Corporation of America. In 1953 he joined that company as a full-time employee, remaining there until 1960. In 1960 he and a partner formed Sage Instruments, Inc., a company which manufactured special medical and biological instruments. Mr. Goertzel received the degrees of Mechanical Engineer and Master of Science in physics from Stevens Institute of Technology, Hoboken, New Jersey, in 1940 and a Ph.D. in physics from New York University, New York, in 1947. Mr. Goertzel is a member of Sigma Xi and a Fellow of the American Physical Society, the New York Academy of Sciences, and the American Association for the Advancement of Science.

Gerhard R. Thompson *IBM Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598.* Mr. Thompson joined the IBM Federal Systems Division in 1968. After one year he transferred to the Advanced Systems Development Division, where he worked on technology support and hardware modeling for large-scale integrated chips used in a small computer. He continued working on advanced versions of this computer when he transferred with the project to East Fishkill, New York, in 1973. In 1978 he joined the advanced engine design group in Kingston, New York. There he was responsible for the logic design of a custom very large-scale integrated circuit chip which is one of the four chips in the central processing unit of the IBM 8150 Computer. His career at Research started in 1982, when he joined the Communication Products Division's image group at Yorktown. This group was later merged into the Research Division's Image Technologies Department, of which he is currently a member. Here he has worked with Gerald Goertzel in developing a digital halftoning technique for continuous-tone images and image-processing algorithms for binary and gray-scale rotations. His present activities are concerned with the printing of color images. Mr. Thompson graduated from Drexel University, Philadelphia, Pennsylvania, in 1967 with a B.S. in electrical engineering. He received an M.S. in electrical engineering from Princeton University, New Jersey, in 1968. Through the IBM Graduate Work Study Program, he earned an M.S. in computer science from Syracuse University, New York, in 1983.