PANDA: Processing Algorithm for Noncoded Document Acquisition

by Yi-Hsin Chen
Frederick C. Mintzer
Keith S. Pennington

With a scanned-document-handling system, documents can be scanned, stored, transmitted to remote locations, viewed on displays and terminals, edited, and printed. These systems hold much promise for office automation, since they facilitate the communication and storage of information that is not easily recoded into the traditional formats for text and graphics. However, most of the current systems that perform these functions are intended for documents that at every point are either black or white, but not gray. These systems effectively exclude documents that contain regions with varying shades of gray (known as continuous-tone regions). PANDA, the Processing Algorithm for Noncoded Document Acquisition, is a technique that processes *mixed documents*, those that contain continuous-tone regions in addition to text, graphics, and line art. PANDA produces a high-quality binary representation of all regions of a mixed document. Furthermore, all regions of the binary representation, including the continuous-tone image regions, are significantly compressed by the run-length-based compression algorithms that underlie scanned-document-handling systems. This is a key feature of PANDA. Indeed, this compression makes practical the inclusion of mixed documents into many existing systems.

Introduction

IBM has a family of products that can process scanned documents consisting of text, graphics, and line art, but not continuous-tone images. These include the IBM 8815 Scanmaster I [1], which can scan and print such documents; DISOSS/PS [2], a software package that can distribute them; and the Image View Facility [3], a software package that can display them on a variety of IBM terminals with the assistance of the GDDM [4] software product. A key component of these products is the Modified Modified READ (MMR) compression technique, which is based on run-length coding and is briefly described in another paper in this issue [5]. Other manufacturers also offer systems that handle scanned documents; most of these systems also contain compression algorithms based on other techniques for coding run-lengths. Examples of these other techniques include the one-dimensional Modified Huffman (MH) code

and the two-dimensional Modified READ (MR) code, both of which are CCITT (International Telegraph and Telephone Consultative Committee) Group III standards [6]. Like the IBM product line, the other systems are usually limited to handling documents that consist of text and line art only. For mixed documents, most existing systems simply threshold all areas of a document, giving a poor representation of the continuous-tone regions. **Figure 1(a)** is an example of a thresholded continuous-tone image.

There are many well-known halftoning techniques that provide good binary representations of continuous-tone images. These include error diffusion, ordered dither, and super-circle. Informative reviews of these techniques are given in [7] and [8]. However, the techniques that best represent continuous-tone images often provide poor representations of text and graphics, with straight lines represented by jagged, discontinuous strokes. PANDA separates the document into two types of regions, which we call *text regions* and *image regions* (defined more fully in the section entitled *Image/text regions*). In *text* regions the underlying document is essentially black or white, while in *image* regions the underlying document is continuous-tone. By thresholding the document in *text* regions and halftoning the document in *image* regions, PANDA produces a representation that has text with sharp edges as well as an accurate representation of the grays in the continuous-tone regions.

A problem with the binary representations of continuous-tone regions is that when run-length-based compression is applied to these representations the data may compress poorly—or even expand, often by as much as a factor of three. For that reason, the use of these halftoning techniques in many current scanned-document systems is impractical. This phenomenon is illustrated in **Table 1**, which gives the compression of several different binary representations of the continuous-tone image of a girl's face taken from Facsimile Test Chart No. 1 of the Institute of Image Electronics Engineers of Japan (IIEEJ). The various representations of this image are given in **Figure 2**.

The lack of compression of images produced by the existing halftoning methods is due to the mismatch between the model of the data assumed by facsimile compression techniques and the actual data produced by the various halftoning methods. The one-dimensional compression techniques, such as Modified Huffman, expect the binary *pels* (picture elements) of the image to appear in long runs of same-colored pels. These techniques achieve compression by coding the lengths of runs of same-colored pels and by using shorter code words to represent the run-lengths that are statistically more frequent. In practice, longer runs are more frequent, so their run-lengths are coded with code words that are shorter than the lengths of the runs themselves; short runs are less frequent, so their run-lengths are coded with code words that are longer than the runs themselves. The

(a)

(b)

Figure 1

Continuous-tone image of panda, processed by (a) thresholding; (b) PANDA.

two-dimensional techniques, such as MR and MMR, also expect runs of black pels to be vertically aligned with corresponding runs of black pels in the line above, offset by at most a few pels. These techniques code the offsets of the beginnings and ends of runs of black pels when the runs are well aligned with runs of pels on the line above. When the runs of black pels are poorly aligned with corresponding

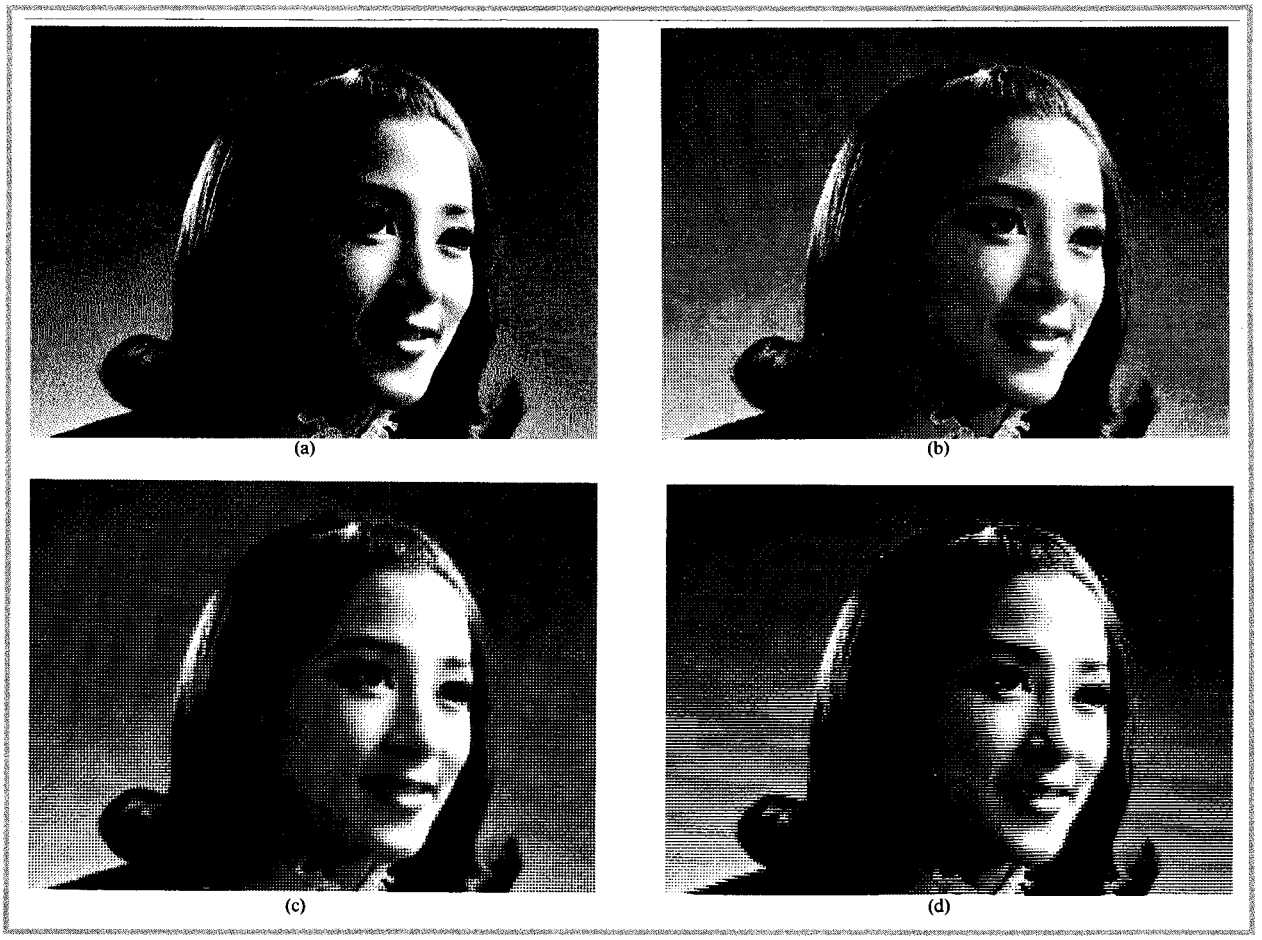

Figure 2

Continuous-tone image of girl's face from IIEEJ Facsimile Test Chart No. 1, halftoned with (a) error diffusion; (b) ordered dither; (c) super-circle; (d) PANDA.

Table 1 Compression of the image of Figure 2 (girl's face) halftoned by several techniques.

Halftone technique	*Bits after compression by MMR*	*Bits per pixel after compression*
Error diffusion	725 240	1.937
Ordered dither	562 216	1.502
Super-circle	395 520	1.056
PANDA	192 424	0.514

runs of pels on the line above, the two-dimensional techniques revert to one-dimensional coding, as described earlier, to code the data.

The well-known halftoning methods simulate shades of gray by filling areas with black and white pels in the proper proportions. The binary images produced generally consist of many short runs of black and white pels. Those images compress poorly when one-dimensional run-length coding is applied because the code words for the short run-lengths are longer than the runs themselves. Furthermore, the well-known halftoning methods generally produce runs of black pels that align poorly with corresponding runs on the preceding line; hence, the two-dimensional coding techniques revert to one-dimensional coding and perform no better.

PANDA was created to allow mixed documents to be compressed sufficiently for efficient handling by existing systems, while providing a high-quality representation of text, line art, and continuous-tone regions. **Figure 1(b)** is an example of a PANDA representation of a continuous-tone image; it was generated from the same multilevel image that was thresholded to produce Figure 1(a). PANDA, like other halftoning techniques, fills each area with the proper proportion of black pels to simulate the desired shade of gray. However, the pel patterns that are printed were chosen

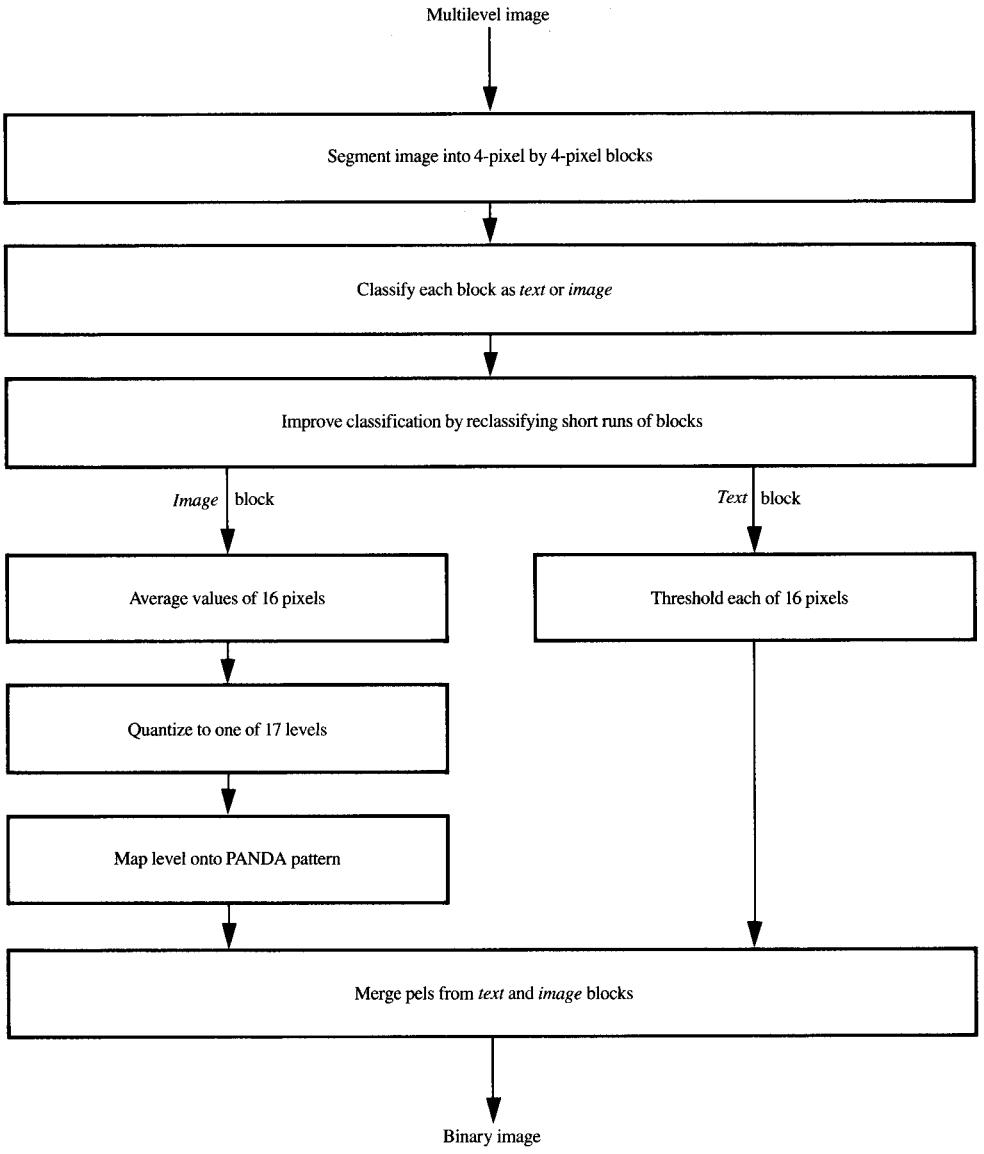

Figure 3

Flowchart of the basic PANDA processing.

to produce as many long runs of same-colored pels as possible when the gray tone of the continuous-tone image varies slowly. Since the gray tones of images generally do vary slowly, the PANDA halftones compress well. It is the choice of halftone patterns that leads to PANDA's superior compression in continuous-tone regions.

PANDA handles regions with text and line art by thresholding, the same method used in many existing products, including Scanmaster. Using PANDA on documents that contain no continuous-tone image produces the same binary representation as would be produced by thresholding, so there is no compression penalty involved.

Often there are interactions between halftone patterns and printers that lead to annoying artifacts. In addition, some people may have a personal preference for one halftoning technique over others. We have been able to process binary documents produced by PANDA, to identify the *image* regions, and to replace the PANDA halftone patterns with other halftone pattern sets. Documents can thus be stored and transmitted in a form that takes advantage of their

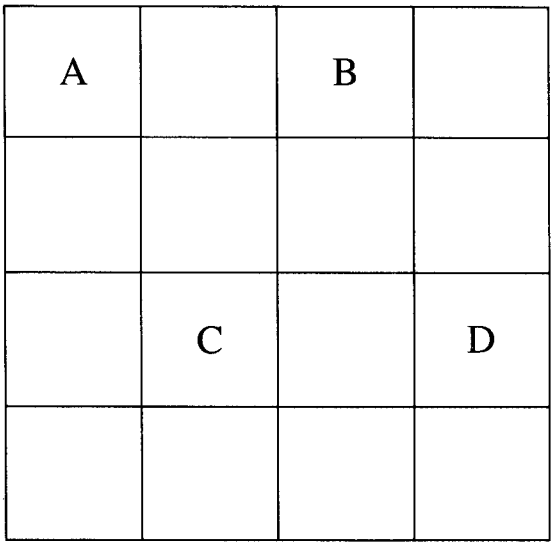

Figure 4

Diagram of a four-pixel by four-pixel block identifying the selected positions used in the *text/image* classification.

Figure 5

Image regions of IIEEJ Facsimile Test Chart No. 1.

PANDA representations to achieve greater compression, and can later be converted to other representations as desired.

In the following sections, we describe the PANDA processing and discuss several of the choices that led to it. Images are often used as examples. All images displayed in this paper were scanned and printed at 200 pels per inch. This is the resolution of the IBM Scanmaster I, a typical component of a scanned-document-handling system.

Outline of PANDA processing

Figure 3 is a flowchart of the basic processing of PANDA which shows that PANDA accepts a multilevel scan of a document as its input. In the following descriptions, we assume that a gray level of zero corresponds to a black *pixel* (picture element, usually multilevel) and that the higher the gray level, the whiter the pixel. The source images for the figures of this paper were quantized to eight bits per pixel when scanned; hence, the gray shade of each pixel was quantized to one of 256 levels. The document is processed first from left to right and then from top to bottom as follows.

1. The scanned document is segmented into nonoverlapping blocks that are four pixels by four pixels square.
2. Each block is classified as either *text* or *image*.

   A block that contains text information is more likely to contain pixels with gray levels corresponding to a white background. Therefore, *text/image* classification can be accomplished by simply examining selected pixels within the block. The more pixels examined, the better the detection of text, but the slower the processing speed. With properly chosen pixel positions in the block, our experiments have shown that four pixels is adequate. The pixel positions we have chosen to examine are labeled A, B, C, and D in **Figure 4**.

   If any of the selected pixels has a gray level above the white threshold, then the block is classified as *text*. Obviously, a block of all-white background is classified as *text*. With this choice, an all-white block will not require halftone processing, which is desirable because halftone processing usually is more time-consuming. For the same reason, we prefer to classify all-black blocks as *text* blocks. For speed, we examine only two pixels (C and D in Figure 4). If both of them have gray levels below the black threshold, then the block is classified as *text*.

   Blocks that are not classified as *text* are classified as *image*.
3. The classification is improved by reclassifying short runs of *image* blocks.

   At this point, we note that *image* regions and *text* regions tend to be sizable areas rather than small isolated islands. We use this fact to improve the classification.

   All runs of *image* blocks that are not within an *image region* and are shorter than 12 blocks are reclassified as *text* runs. *Image* regions for the example of IIEEJ Facsimile Test Chart No. 1 are illustrated in **Figure 5**. In producing the figure, all pixels in the *image* regions were halftoned; all pixels in *text* regions were converted to white pels.

4. Each pixel in a *text* block is thresholded. Pixels in blocks classified as *image* undergo several further stages of processing:
   - The average gray value of the 16 pixels in the block is computed.
   - This average gray value is quantized to one of 17 (0–16) levels by a process we call *quantization error diffusion*, which is described in a later section.
   - The quantized level is mapped onto its PANDA pattern. There are 17 PANDA patterns, one for each level. A detailed description of the patterns is given in the section entitled *PANDA patterns*.

   Our experiments have shown that it is unnecessary to eliminate the short runs of *text* blocks.
5. The final step of PANDA processing merges the bitmaps generated by the processing for the *text* blocks and those for the *image* blocks.

An example of a mixed document processed by PANDA is given in **Figure 6**.

This processing, as outlined above, works well on most documents with fixed thresholds. We have also produced an adaptive version of PANDA that is described in the section entitled *Adaptive PANDA*.

Various prior attempts have been made to distinguish between the continuous-tone images from the text and graphics regions in multilevel mixed documents [9–11]. The prior classification techniques were based on gradient, frequency, or statistical properties of the area surrounding the pixel (or block) examined. These schemes generally require more processing time and storage than the method described in this paper.

Image/text regions

In the discussion of *image/text* classification, we noted that the all-white blocks and some partially-black blocks are classified as *text* blocks. One may encounter a short image run (shorter than 12 blocks) imbedded in either a white area or a black area either of which in turn lies within an image area. We would like to keep those short image runs unchanged by the reclassification processing. Therefore, we need to know if an image run is within any image region in order to avoid excessive reclassification.

Since documents are processed row by row (i.e., in four-line blocks), discussion of image regions can be limited to rows. It follows that an image region can be represented by its starting and ending columns on a particular row, just as we represented image and text runs. Assuming that columns are numbered from left to right, the *image regions* are created and updated as follows:

- Initially there is no *image region*.
- An image run which is not in an *image region* and which is longer than 12 blocks creates a new *image region* with the same column range as the run.
- If an *image region* intersects no image runs, then this *region* is eliminated.
- Let run_i be the leftmost image run such that run_i intersects with *image region*$_k$.
  Then the start of *image region*$_k$ = start of run_i.
- Let run_j be the rightmost image run such that run_j intersects with *image region*$_k$.
  Then the end of *image region*$_k$ = end of run_j.

By definition, *text regions* are the parts of the document that are not *image regions*.

Quantization error diffusion

PANDA uses a technique that we call *quantization error diffusion* to quantize the gray level of a block to one of 17 values. As contrasted with simple quantization, this technique, we believe, adds much to the quality of the printed image.

With simple quantization, the 17 quantization levels are uniformly spaced between black and white, and the quantized value is the level of the 17 closest to, but not greater than, the input gray level. **Figure 7** is an example of an image quantized in this way. Note that in areas where the gray tones change very slowly, such as in the background to the right of the face, contours can be seen along the edges between adjacent levels. One such contour begins at the edge of the girl's hair just to the right of her left eye and proceeds downward and to the right at approximately a 45-degree angle. The image in Figure 2(d) was processed in the same manner, except that quantization error diffusion was used to perform the quantization. Note that in Figure 2(d) the contours are much less noticeable.

The major disadvantage of quantization error diffusion is that it tends to reduce the compression that can be obtained. However, it has been our experience that this effect is generally small. For the example of the girl's face given in Figures 2(d) and 7, the difference in compression is 0.008 bits per pel. We feel that the improvement in quality more than justifies this minor reduction in compression.

Quantization error diffusion is an adaptation of the *error diffusion* technique, which is generally credited to Floyd and Steinberg [12] and was popularized within IBM by Stucki [13] and by K. Y. Wong and T. D. Friedman in an unpublished manuscript entitled "A General Purpose Image Processing Algorithm for Text, Halftone and Continuous-tone Documents." With error diffusion, the input pixel is quantized to one of two levels, and the quantization error is passed to neighboring pixels in fixed proportions. Quantization error diffusion adapts the idea of *error diffusion* to the problem of quantizing the average value of a block of pixels to one of several levels.

To begin describing quantization error diffusion, let us denote $G_{m,n}$ as the average gray level of the block (m, n), the nth block on row m. Then the modified version $\tilde{G}_{m,n}$ of $G_{m,n}$

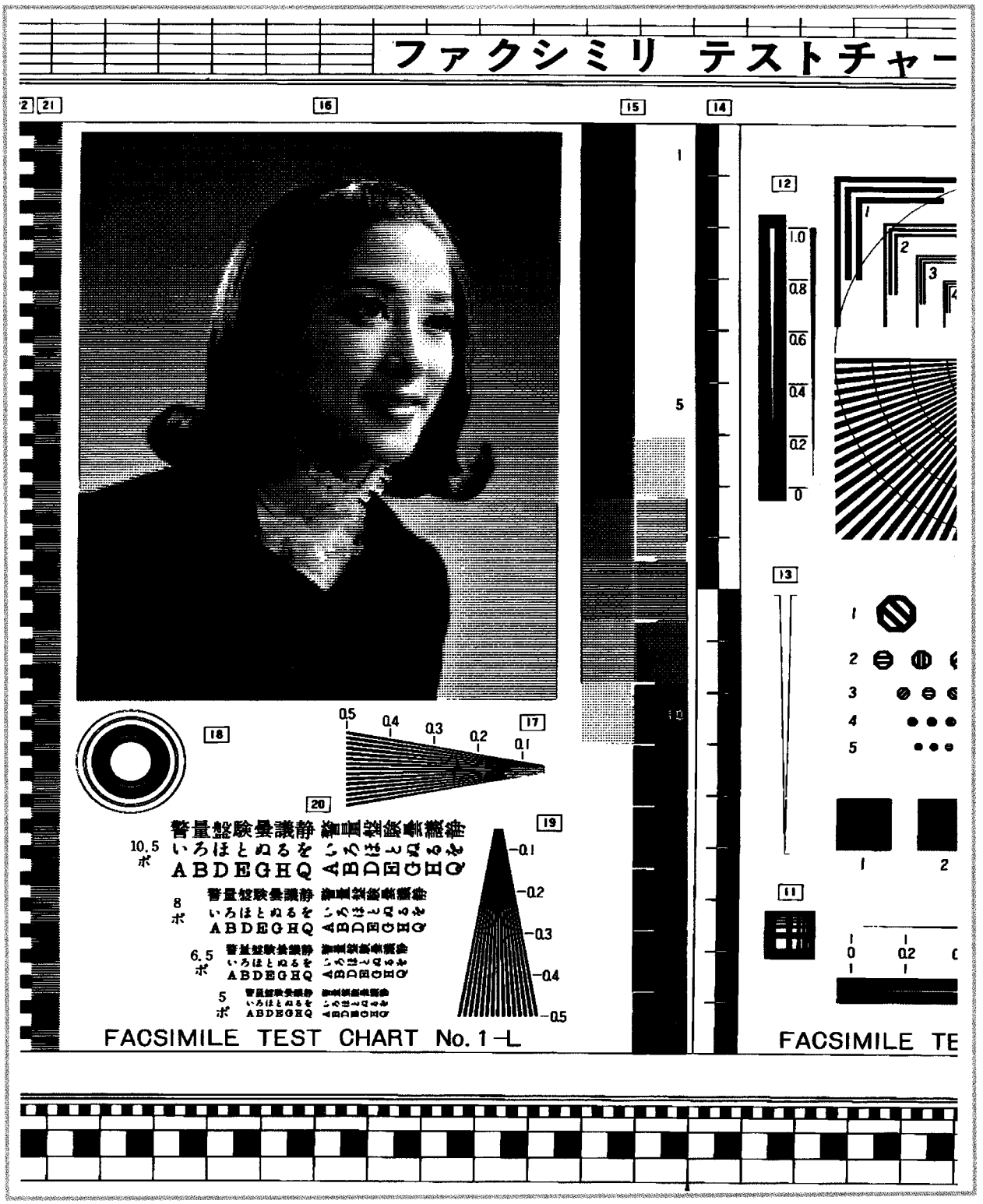

Figure 6

Portion of IIEEJ Facsimile Test Chart No. 1 scanned at 200 pixels per inch. MMR compression = 0.192 bits per pixel.

is computed as

$$\tilde{G}_{m,n} = G_{m,n} + \tfrac{1}{2} E_{m-1,n} + \tfrac{1}{2} E_{m,n-1},$$

and $\tilde{G}_{m,n}$ is quantized according to

$$\tilde{G}_{m,n} = Q_k + E_{m,n},$$

where Q_k is the nearest of the 17 quantization levels that is not greater than $\tilde{G}_{m,n}$, and $E_{m,n}$ is the error generated by block (m, n).

PANDA patterns

The 17 PANDA patterns are given in **Figure 8.** As stated previously, each pattern corresponds one-to-one with a quantization level and is labeled with that level.

As we mentioned earlier, the PANDA patterns were chosen to produce long runs of pels for a slowly changing continuous-tone image. Let us now examine Figure 8 to see what would happen if along a row of *image* blocks the quantization started at level 5 and gradually increased to level 8. This would generate four rows of output pels. The top row and the second row would consist only of long runs of black pels. These rows would be very well compressed by facsimile compression techniques. The bottom row would consist only of a long run of white pels. This row would also be very well compressed. Only the third row would have many runs and compress poorly. The net effect would be a well-compressed image.

Conversion to other binary representations

There are several situations in which it may be desirable to convert PANDA images to other binary representations. There may be an interaction between a specific printer and the PANDA patterns that produces annoying artifacts, or the user may prefer the visual effect of other halftone patterns.

In these cases, we recommend processing the image with PANDA, compressing the image with MR or MMR, transmitting or storing the image in this compressed form, and then converting the image to the desired representation at print time. Using this sequence of operations retains the storage and transmission-time advantages that PANDA was intended to capture, while freeing the user from the necessity of using PANDA halftone patterns to print continuous-tone regions of his documents.

We have developed a conversion procedure that operates on a binary image produced by PANDA, estimates the *text* and *image* regions generated during the PANDA processing, and replaces the PANDA halftone patterns with another set of halftone patterns.

We have tried this procedure on a number of examples with success. The image of Figure 6 was converted, in the *image* regions, to a set of super-circle patterns to create the image of **Figure 9.** The patterns we chose were four-pel by four-pel squares, the same size as the PANDA patterns. With super-circle patterns, the black bits fill the centers of the squares first. When the image of Figure 9 was reconverted to the PANDA patterns, there was no difference between the twice-converted document and the original, Figure 6. This indicates that our procedure consistently identifies *text* and *image* areas when it operates on a binary representation produced by PANDA (or our conversion procedure). However, the regions identified may not exactly match the regions identified during the PANDA processing.

Figure 7

Continuous-tone image of girl's face, processed with simple quantization and PANDA patterns.

Our conversion procedure requires exact knowledge of the pattern size and the pattern set in *image* regions. Although our procedure fills the immediate need to convert binary PANDA images to other halftone pattern sets, it does not work on rotated PANDA images, shifted PANDA subimages, or other halftoned images where the periodicity and pattern set are not known. Identification of *text* and *image* regions for this more general class of binary mixed documents is a difficult problem, and perhaps a fertile area for future research.

Adaptive PANDA

A number of thresholds are used in the PANDA processing. In the processing to classify blocks as either *text* or *image*, the input pixels are compared with a white threshold *WBAR* and a black threshold *BBAR*. In the processing of *text* blocks, each pixel is compared with the threshold *THR*. If the pixel is greater than *THR*, a white pel is produced; otherwise a black pel is produced. Proper selection of these thresholds is obviously a matter of importance.

For a broad class of documents, these thresholds may be simply chosen. We have typically chosen *BBAR* to be 10 percent of the maximum gray level and *WBAR* to be 90 percent of the maximum gray level, where the maximum

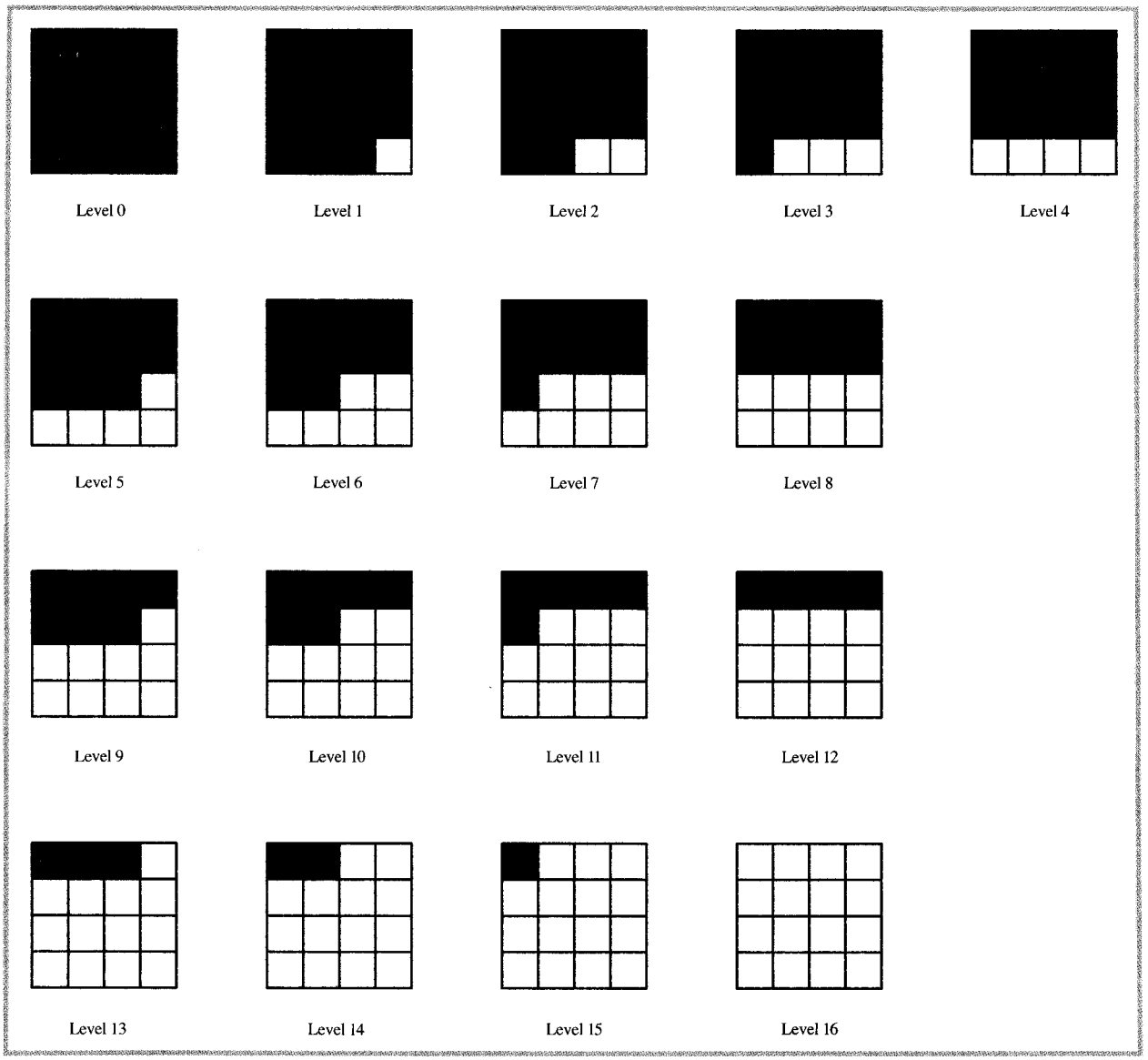

Figure 8

The set of PANDA patterns.

gray level corresponds to white and the gray level 0 corresponds to black. For *THR*, we have used the average of *WBAR* and *BBAR*. With these choices we have processed many documents with good results.

However, we have noticed some problems, particularly in *text* regions, when the quality of the source document is poor. Sometimes a document may have a dark background, or black print that is insufficiently dark, or both. The dark background is often caused by coloration of the paper, whereas insufficiently dark print is often caused by a worn-out typewriter (or printer) ribbon. To process poor-quality originals, we have developed an *adaptive PANDA processing* which adjusts the PANDA thresholds as the document is processed.

Adaptive PANDA maintains a histogram of the pixels previously examined during the processing of a document. As each new group of four lines of pixels becomes available, the histogram is updated.

The thresholds *WBAR* and *BBAR* are re-evaluated from the histogram each time 64 new lines of input have been processed. The histogram generally has a main peak among the high gray levels. This is due to the white background of most scanned documents. *WBAR* is chosen so that it is below the main peak, so that the value in the histogram at

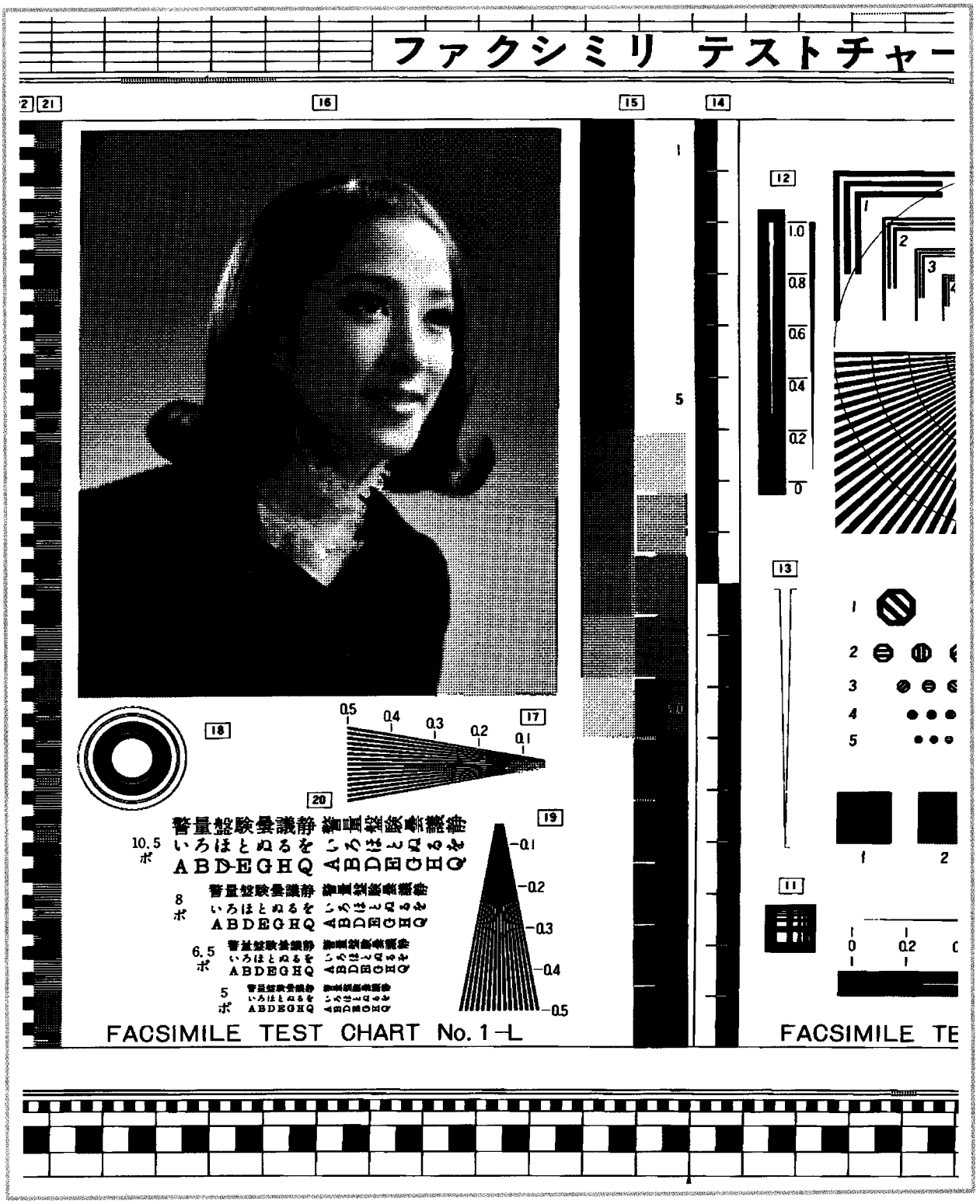

Figure 9

Super-circle representation of the IIEEJ test chart derived from Figure 6 by using the conversion procedure described in the paper.

Table 2 Compression of IIEEJ test chart, with image portions halftoned by several techniques, and text portions simply thresholded.

Halftone technique	*Kbytes storage after MMR compression*	*Transmission time* (min @ 9600 bits/s)
Error diffusion	177	2.4
Ordered dither	133	1.9
Super-circle	105	1.5
PANDA	65	0.9

WBAR is no greater than twice the average value of the histogram, and so that at least 6 percent of the examined pixels are whiter than *WBAR*. *BBAR* is chosen so that at least 6 percent of the examined pixels are blacker than *BBAR*. Both *WBAR* and *BBAR* are limited in range so that *WBAR* cannot go below a certain value and *BBAR* cannot go above a certain value. Both are constrained to be multiples of four and are limited to a maximum change of four at each re-evaluation. For thresholding *text* blocks, two thresholds are used. *TTHR* is used to threshold blocks classified as *text* in text regions, and *ITHR* is used to threshold blocks classified as *text* in image regions, where the text and image regions are as described in the section entitled *Image/text regions*. The use of two thresholds for processing *text* improves the quality when the *text/image* classification is less reliable. The threshold *ITHR* is computed as the average of *WBAR* and *BBAR*, and *TTHR* is computed as 75 percent of *WBAR* plus 25 percent of *BBAR*.

Although the adaptive PANDA processing is somewhat more complicated, neither the processing load nor the storage requirement is prohibitively increased. If the domain of interest is expected to include a sizable number of poor-quality source documents, adaptive PANDA is recommended.

Algorithmic choices

In devising the PANDA processing, a number of algorithmic choices were made. Perhaps the most fundamental of these was the choice of the block size. In order to adequately detect *text* blocks in the classification process, the block must be large enough to cover the sharp transition from very dark to very light in text or line drawings. However, the smaller the block, the more accurate the representation of the *text/image regions*, and the better the quality of the documents produced. Our experiments have shown that a block size of four pixels by four pixels is adequate for most documents scanned at 200 pixels per inch. If the scanning resolution were higher, a larger block size might be necessary for adequate detection of *text* blocks. There are other considerations for choosing block size, including speed and compression. The larger the block, the faster the processing speed, and the higher the compression when PANDA patterns are used. However, as mentioned before, too large a block size will degrade the quality of the documents produced.

If compression is not a concern, any appropriate halftoning technique can be used for processing the image region. We favor error diffusion, in one of its many forms, if quality is the sole consideration.

An obvious improvement in compression can be made by using the mirror image of PANDA patterns for processing alternate blocks. Although the compression is up to 10 percent better, depending on the document, the quality degrades dramatically. We do not recommend this modification unless the document will later be converted to another binary representation using our conversion procedure.

Another choice we made was the selection of which pixels to use in the *text/image* classification procedure. In the interest of processing speed, we limited ourselves to examining four selected pixels. We also examined the use of different pixel positions for even and odd blocks. The main criterion in choosing the pixel positions is that they should be spread as uniformly as possible, and in as many different rows and columns as possible. Examining only four pixels is adequate because the subsequent reclassification procedure works so well. This is also the reason why the selection of the pixels to be examined is not critical.

Conclusions

PANDA is a technique that processes multilevel scans of mixed documents and produces a high-quality binary representation that can be well compressed. This is important since it reduces both storage requirements and transmission times, thus making it practical for existing scanned-document-handling systems to accept mixed documents. **Table 2** illustrates this advantage for IIEEJ Facsimile Test Chart No. 1, for which the PANDA version is given in Figure 6.

In addition, a reduced-resolution version of the grayness of the original documents is preserved by the quantization error diffusion technique and can easily be recovered by translating PANDA patterns to the corresponding gray levels. Therefore, documents can be stored in MMR compressed PANDA form and converted to other representations, such as super-circle, at print time or display time.

The main features of PANDA are listed below.

Quality: PANDA produces text and graphics with sharp edges, yet produces an accurate representation of continuous-tone image regions.

Coding efficiency: Under any run-length-based compression scheme, continuous-tone image regions of PANDA documents compress much better than if these

regions were represented by other halftoning schemes. If the document contains no continuous-tone image regions, there is no compression penalty for using PANDA.

Compatibility: PANDA produces documents that are compatible with many existing systems, including Scanmaster and the Image View Facility (see [5]).

Convertibility: Continuous-tone image regions of PANDA documents can be converted to other halftone pattern sets, such as super-circle.

Simplicity: Since PANDA does not require either much processing time or storage, it is suitable for operation in a real-time environment. We have implemented the basic PANDA processing on a prototype signal processor connected to a modified Scanmaster. The signal processor performed PANDA processing fast enough to keep up with the scanner, which requires 30 seconds to scan a page.

References and note

1. *IBM Scanmaster I (Machine Type 8815) Description Manual*, Order No. GA18-2094, IBM Corporation; available through IBM branch offices.
2. *Distributed Office Support System/Professional Support (DISOSS/PS)*, Order No. SH20-2695, IBM Corporation; available through IBM branch offices. DISOSS/PS is a program (Program No. 5796-PRH) that provides 3270 display support for DISOSS/370.
3. *Image View Facility Program Description and Operations Manual*, Order No. SB19-5919 (Program Number 5785-ECX), IBM Corporation; available through IBM branch offices.
4. *Graphical Data Display Manager (GDDM) General Information Manual*, Order No. GC33-0100, IBM Corporation; available through IBM branch offices.
5. K. L. Anderson, F. C. Mintzer, G. Goertzel, J. L. Mitchell, K. S. Pennington, and W. B. Pennebaker, "Binary-Image-Manipulation Algorithms in the Image View Facility," *IBM J. Res. Develop.* **31,** No. 1, 16–31 (January 1987, this issue).
6. Roy Hunter and A. Harry Robinson, "International Digital Facsimile Coding Standards," *Proc. IEEE* **68,** No. 7, 854–867 (1980).
7. Peter Stucki, *Advances in Digital Image Processing: Theory, Application, Implementation*, Plenum Publishing Co., New York, 1979.
8. 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.
9. Hiroshi Ochi and Nobuji Tetsutani, "Method and Apparatus for Gray Level Signal Processing," U.S. Patent 4,545,811, 1985.
10. Sidney J. Fox, Filip J. Yeskel, and William J. Zimmermann, Jr., "Universal Thresholder/Discriminator," U.S. Patent 4,554,593, 1985.
11. Harry N. Kannapell and Paul G. Niefeld, "Text/Continuous Tone Image Decision Processor," U.S. Patent 4,577,235, 1986.
12. Robert Floyd and Louis Steinberg, "An Adaptive Algorithm for Spatial Gray Scale," *1975 SID International Symposium, Digest of Technical Papers*, pp. 36–37.
13. P. Stucki, "MECCA—A Multiple-Error Correction Computation Algorithm for Bilevel Image Hardcopy Reproduction," *Research Report RZ-1060*, IBM Research Laboratory, Zurich, Switzerland, 1981.

Received November 1, 1985; accepted for publication August 19, 1986

Yi-Hsin Chen *IBM Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598.* Ms. Chen is an advisory programmer at the IBM Thomas J. Watson Research Center. Since joining IBM in 1977, she has held various positions in design and development of software systems. In 1980, she worked on image-handling systems at the IBM Zurich laboratory. In 1981, she joined the CPD Advanced Technology Group, working in the area of Image Processing, and later became a member of the Image Technology Group of the Research Division, working on the development of techniques for image handling and halftoning. Ms. Chen received a B.S. in mathematics in 1972 from Tsinghua University, Hsinchu, Republic of China, and an M.S. in mathematics in 1973 and an M.S. in computer science in 1976, both from the University of Toronto, Ontario, Canada.

Frederick C. Mintzer *IBM Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598.* Dr. Mintzer attended Princeton University and received the Ph.D. degree in electrical engineering in 1978. Also in 1978, he joined the IBM Thomas J. Watson Research Center, engaging in research on distributed digital signal processing, signal processing architectures, and data communications. In 1980, he became the manager of the Signal Processing Applications project, and continued research in these areas. In 1983, he joined the Image Technologies Department as manager of the NCI Architectures project, engaging in research on image-processing algorithms. Dr. Mintzer received an Outstanding Innovation Award for his contributions to the Image View Facility in 1985. His current research is centered on image display and print algorithms.

Keith S. Pennington *IBM Thomas J. Watson Research Center, P.O. Box 218, Yorktown Heights, New York 10598.* Dr. Pennington is senior manager of the Image Technologies and Erosion Printing Studies departments at the Thomas J. Watson Research Center. He graduated with a B.Sc. in physics from Birmingham University, England, in 1957 and a Ph.D. in physics from McMaster University, Hamilton, Ontario, Canada, in 1961. He started his research career at Bell Telephone Laboratories, Murray Hill, New Jersey, where he developed the first multicolor holograms and also did research in holographic interferometry and optical information processing. He joined IBM Research in 1967 and subsequently made several contributions to the development of improved holographic materials and techniques for three-dimensional scene analysis. Dr. Pennington was appointed manager of the exploratory terminal technologies group in 1972, and in this position he initiated the work in the development of the resistive ribbon transfer printing technology and other printing technologies. He became manager of the Image Technologies Department in 1979 and has responsibility for several projects related to high-performance videoconferencing systems, document processing, and scanning systems, as well as novel high-resolution printing processes. Dr. Pennington has written three book chapters related to holography and optical information processing and during 1971–1972 served both as a participant and as a group leader for the National Academy of Sciences Undersea Warfare Committee. While at IBM, he has received two IBM Outstanding Contribution Awards, an Outstanding Innovation Award, and an Outstanding Technical Achievement Award. Dr. Pennington is a member of the Institute of Electrical and Electronics Engineers and the Optical Society of America.