Recent Developments in Document Image Watermarking and Data Hiding

Minya Chen, Edward K. Wong*, Nasir Memon** and *Scott Adams+*

*Department of Computer and Information Science
Polytechnic University
5 Metrotech Center,
Brooklyn, NY 11201

+Air Force Research Laboratory
32 Brooks Rd,
Rome, NY 13441

ABSTRACT

With the proliferation of digital media such as images, audio, and video, robust digital watermarking and data hiding techniques are needed for copyright protection, copy control, annotation, and authentication. While many techniques have been proposed for digital color and grayscale images, not all of them can be directly applied to binary document images. The difficulty lies in the fact that changing pixel values in a binary document could introduce irregularities that are very visually noticeable. Over the last few years, we have seen a growing but limited number of papers proposing new techniques and ideas for document image watermarking and data hiding. In this paper, we present an overview and summary of recent developments on this important topic, and discuss important issues such as robustness and data hiding capacity of the different techniques.

Keywords: data hiding; watermarking; binary images; document images; authentication; copyright control

1. INTRODUCTION

As digital devices such as scanners and digital cameras become more available, and mass storage media for digital data becomes more affordable, the use of digital images in practical applications is becoming more widespread. Practical imaging applications range from famous works of art, to bank checks, and medical images. Reliable methods for copyright protection, copy control, annotation, and authentication are therefore needed. A variety of digital watermarking and data hiding techniques have been proposed for such purposes. However, most of the methods developed today are for grayscale and color images [1], where the gray level or color value of a selected group of pixels is changed by a small amount without causing visually noticeable artifacts. These techniques cannot be directly applied to binary document images where the pixels have either a 0 or a 1 value. Arbitrarily changing pixels on a binary image causes very noticeable artifacts (see Figure 1 for an example). A different class of embedding techniques must therefore be developed. These would have important applications in a wide variety of document images that are represented as binary foreground and background; e.g. bank checks, financial instruments, legal documents, driver licenses, birth certificates, digital books, engineering maps, architectural drawings, road maps, etc. Until recently, there has been little work on watermarking and data hiding techniques for binary document images. Over the last few years, we have seen a growing but limited number of papers proposing new techniques and ideas for document image

watermarking and data hiding. In this paper, we present an overview and summary of recent developments on this important topic, and present discussions on important issues such as robustness and data hiding capacity.

Figure 1. Effect of Arbitrarily Changing Pixel Values on a Binary Image

Most document images are binary in nature and consist of a foreground and a background color. The foreground could be printed characters of different fonts and sizes in text documents, handwritten letters and numbers in a bank check, or lines and symbols in engineering and architectural drawings. Some documents have multiple gray levels or colors, but the number of gray levels and colors is usually few and each local region usually has a uniform gray level or color, as opposed to the different gray levels and colors you find at individual pixels of a continuous-tone image. Some binary documents also contain grayscale images represented as half-tone images, e.g. the photos in a newspaper. In such images, *n x n* binary patterns are used to approximate gray level values of a gray scale image, where *n* typically ranges from 2 to 4. The human visual system performs spatial integration of the fine binary patterns within local regions and perceives them as different intensities [2].

In Section 2 of this paper, we summarize recent developments in binary document image watermarking and data hiding techniques. In Section 3, we present a discussion on these techniques, and in Section 4, we give our concluding remarks.

2. RECENT DEVELOPMENTS

Watermarking and data hiding techniques for binary document images can be classified according to one of the following embedding methods: text line, word, or character shifting, boundary modifications, fixed partitioning of the image into blocks, modification of character features, modification of run-length patterns, or modifications of half-tone images. In the rest of this section we describe representative techniques for each of these methods.

2.1 Text Line, Word or Character Shifting

This class of methods shifts a text line, a group of words, or a group of characters by a small amount to embed data. They are applicable to documents with formatted text.

S. Low and co-authors have published a series of papers on document watermarking based on line and word shifting ([3] to [7]). These methods are applicable to documents that contain paragraphs of printed text. Data is embedded in text documents by shifting lines and words spacing by a small amount (1/150 inch.) For instance, a text line can be moved up to encode a '1' or down to encode a '0', a word can be moved left to encode a '1' or right to encode '0'. The techniques are robust to printing, photocopying, and scanning. In the decoding process,

distortions and noise introduced by printing, photocopying and scanning are corrected and removed as much as possible. Detection is by use of maximum-likelihood detectors. In the system they implemented, line shifts are detected by the change in the distance of the marked line and two control lines -- the lines immediately above and below the marked line. In computing the distance between two lines, the estimated centroids of the horizontal profiles (projections) of the two lines are used as reference points. Vertical profiles (projections) of words are used for detecting word shifts. The block of words to be marked (shifted) is situated between two control blocks of words. Shifting is detected by computing the correlation between the received profile and the uncorrupted marked profile. The line shifting approach has low embedding capacity but the embedded data is robust to severe distortions introduced by processes such as printing, photocopying, scanning, and facsimile transmission. The word shifting approach has better data embedding capacity but reduced robustness to printing, photocopying and scanning.

In [8], a combined approach that marks a text document by line or word shifting, and detects the watermark in the frequency domain by Cox et. al.'s algorithm [9] was proposed. It attempts to combine the unobtrusiveness of spatial domain techniques and the good detection performance of frequency domain techniques. Marking is performed according to the line and word shifting method described above. The frequency watermark X is then computed as the largest N values of the absolute differences in the transforms of the original document and the marked document. In the detection process, the transform of the corrupted document is first computed. The corrupted frequency watermark X^* is then computed as the largest N values of the absolute differences in the transform of the corrupted document and the original document. The detection of watermark is by computing a similarity between X and X^*. This method assumes that the transform of the original document, and the frequency watermark X computed from the original document and the marked document (before corruption) is available during the detection process.

In [10], it is shown that the height of a bounding box enclosing a group of words can be used to embed data. The height of the bounding box is increased by either shifting certain words or characters upward, or by adding pixels to end lines of characters with ascenders or descenders. The method was proposed to increase the data embedding capacity over the line and/or word shifting methods described above. Experimental results show that bounding box expansions as small as 1/300 inch can be reliably detected after several iterations of photocopying. For each mark, one or more adjacent words on an encodable text line are selected for displacement according to a selection criterion. The words immediately before and after the shifted word(s), and a block of words on the text line immediately above or below the shifted word(s), remain unchanged and are used as "reference heights" in the decoding process. The box height is measured by computing a local horizontal projection profile for the bounding box. This method is very sensitive to baseline skewing. A small rotation of the text page can cause distortions in bounding box height, even after de-skewing corrections. Proper methods to deal with skewing require further research.

In [11], character spacing is used as the basic mechanism to hide data. A line of text is first divided into blocks of characters. A data bit is then embedded by adjusting the widths of the spaces between the characters within a block, according to a predefined rule. This method has advantage over the word spacing method above in that it can be applied to written languages that do not have spaces with sufficiently large width for word boundaries; e.g. Chinese, Japanese, and Thai. The method has embedding capacity comparable to that of the word shifting method. Embedded data is detected by matching character spacing patterns corresponding to data bits '0' or '1'. Experiments show that the method can withstand document duplications. However, improvement is needed for the method to be robust against severe document degradations. This

could be done by increasing the block size for embedding data bits, but this also decreases the data embedding capacity.

2.2 Fixed Partitioning of Images

This class of methods partitions an image into fixed blocks of size *m* x *n,* and computes some pixel statistics or invariants from the blocks for embedding data. They can be applied to binary document images in general; e.g. documents with formatted text or engineering drawings.

In [12], the input binary image is divided into 3 x 3 (or larger) blocks. The flipping priorities of pixels in a 3 x 3 block are then computed and those with the lowest scores can be changed to embed data. The flipping priority of a pixel is indicative of the estimated visual distortion that would be caused by flipping the value of a pixel from 0 to 1 or from 1 to 0. It is computed by considering the change in smoothness and connectivity in a 3 x 3 window centered at the pixel. Smoothness is measured by the horizontal, vertical, and diagonal transitions, and connectivity is measured by the number of black and white clusters in the 3 x 3 window. Data is embedded in a block by modifying the total number of black pixels to be either odd or even, representing data bits 1 and 0, respectively. Shuffling is used to equalize the uneven embedding capacity over the image. It is done by random permutation of all pixels in the image after identifying the flippable pixels.

In [13], an input binary image is divided into blocks of 8 x 8 pixels. The numbers of black and white pixels in each block are then altered to embed data bits 1 and 0. A data bit 1 is embedded if the percentage of white pixels is greater than a given threshold, and a data bit 0 is embedded if the percentage of white pixels is less than another threshold. A group of contiguous or distributed blocks is modified by switching white pixels to black or vice versa until such thresholds are reached. For ordinary binary images, modifications are carried out at the boundary of black and white pixels, by reversing the bits that have the most neighbors with the opposite pixel value. For dithered images, modifications are distributed throughout the whole block by reversing bits that have the most neighbors with the same pixel value. This method has some robustness against noise if the difference between the thresholds for data bits 1 and 0 is sufficiently large, but this also decreases the quality of the marked document.

In [14], a data hiding scheme using a secret key matrix K and a weight matrix W is used to protect the hidden data in a host binary image. A host image F is first divided into blocks of size m x n. For each block F_i, data bits $b_1 b_2 ... b_r$ are embedded by ensuring the invariant

$$SUM((F_i \oplus K) \otimes W) \equiv b_1 b_2 ... b_r (\text{mod} \quad 2^r),$$

where $\oplus$ represents the bit-wise exclusive OR operation, $\otimes$ represents pair-wise multiplication, and SUM is the sum of all elements in a matrix. Embedded data can be easy extracted by computing

$$SUM((F_i \oplus K) \otimes W)(\text{mod} \quad 2^r)$$

The scheme can hide as many as $\lfloor \log_2(mn+1) \rfloor$ bits of data in each image block by changing at most 2 bits in the image block. It provides high security, as long as the block size (m x n) is reasonably large. In a 256 x 256 test image divided into blocks of size 4 x 4, 16,384 bits of information was embedded. This method does not provide any measure to ensure good visual quality in the marked document.

In [15], an enhancement was made to the method proposed in [14] by imposing the constraint that every bit that is to be modified in a block is adjacent to another bit that has the opposite value. This improves the visual quality of the marked image by making the inserted bits less visible, at the expense of sacrificing some data hiding capacity. The new scheme can hide up to $\lfloor \log_2(mn+1) \rfloor - 1$ bits of data in an *m* x *n* image by changing at most 2 bits in the image block.

2.3 Boundary Modifications

In [16], the data is embedded in the 8-connected boundary of a character. A fixed set of pairs of five-pixel long boundary patterns were used for embedding data. One of the patterns in a pair requires deletion of the center foreground pixel, whereas the other requires the addition of a foreground pixel. A unique property of the proposed method is that the two patterns in each pair are dual of each other -- changing the pixel value of one pattern at the center position would result in the other. This property allows easy detection of the embedded data without referring to the original document, and without using any special enforcing techniques for detecting embedded data. Experimental results showed that the method is capable of embedding about 5.69 bits of data per character (or connected component) in a full page of text digitized at 300 dpi. The method can be applied to general document images with connected components; e.g. text documents or engineering drawings.

2.4 Modifications of Character Features

This class of techniques extracts local features from text characters. Alterations are then made to the character features to embed data.

In [17], text areas in an image are identified first by connected component analysis, and are grouped according to spatial closeness. Each group has a bounding box that is divided into four partitions. The four partitions are divided into two sets. The average width of the horizontal strokes of characters is computed as feature. To compute average stroke width, vertical black runs with lengths less than a threshold are selected and averaged. Two operations – "make fat" and "make thin" – are defined by increasing and decreasing the lengths of the selected runs, respectively. To embed a "1" bit, the "make fat" operation is applied to partitions belonging to set 1, and the "make thin" operation is applied to partitions belongs to set 2. The opposite operations are used to embed "0" bit. In the detection process, detection of text line bounding boxes, partitioning, and grouping are performed. The stroke width features are extracted from the partitions, and added up for each set. If the difference of the sum totals is larger than a positive threshold, the detection process outputs 1. If the difference is less than a negative threshold, it outputs 0. This method could survive the distortions caused by print-and-scan (re-digitization) processes. The method's robustness to photocopying needs to be furthered investigated.

In [18], a scheme is presented to embed secret messages in the scanned grayscale image of a document. Small sub-character-sized regions that consist of pixels that meet criteria of text-character parts are identified first, and the lightness of these regions are modulated to embed data. The method employs two scans of the document – a low resolution scan and a high resolution scan. The low-resolution scan is used to identify the various components of the document and establish a coordinate system based on the paragraphs, lines and words found in the document. A list of sites for embedding data is selected from the low resolution scanned image. Two site selection methods were presented in the paper. In the first method, a text paragraph is partitioned into grids of 3 x 3 pixels. Grid cells that contain predominately text-type pixels are selected. In

the second method, characters with long strokes are identified. Sites are selected at locations along the stroke. The second scan is a full-resolution scan that is used to generate the document copy. The pixels from the site lists generated in the low-resolution scan are identified and modulated by the data bits to be embedded. Two or more candidate sites are required for embedding each bit. For example, if the difference between the average luminance of the pixels belonging to the current site and the next one is positive, the bit is a 1; else, the bit is a 0. For robustness, the data to be embedded is first coded using an error correcting code. The resulting bits are then scrambled and dispersed uniformly across the document page. For data retrieval, the average luminance for the pixels in each site is computed and the data is retrieved according to the embedding scheme and the input site list. This method was claimed to be robust against printing and scanning. However, this method requires that the scanned grayscale image of a document be available. The data hiding capacity of this method depends on the number of sites available on the image, and in some cases, there might not be enough sites available to embed large messages.

2.5 Modification of Run-Lengths

In [19], a method was proposed to embed data in the run-lengths of facsimile images. A facsimile document contains 1,728 pixels in each horizontal scan line. Each run length of black (or foreground) pixels is coded using modified Huffman coding scheme according to the statistical distribution of run-lengths. In the proposed method, each run length of black pixels is shortened or lengthened by one pixel according to a sequence of signature bits. The signature bits are embedded at the boundary of the run lengths according to some pre-defined rules.

2.6 Modifications of Half-Toning Images

Several watermarking techniques have been developed for half-tone images that can be found routinely in printed matters such as books, magazines, newspapers, printer outputs, etc. This class of methods can only be used for half-tone images, and are not suitable for other types of document images. The methods described in [20] and [21] embed data during the half-toning process. This requires the original grayscale image. The methods described in [13],[22],[23] and [24] embed data directly into the half-tone images after they have been generated. The original grayscale image is therefore not required.

In [20], a sequence of two different dither matrices (instead of one) was used in the half-toning process to encode the watermark information. The order in which the two matrices are applied is the binary representation of the watermark. In [25] and [26], two screens were used to form two halftone images and data was embedded through the correlations between the two screens.

In [22] and [23], three methods were proposed to embedded data at pseudo-random locations in half-tone images without knowledge of the original multi-tone image and the half-toning method. The three methods, named DHST, DHPT, and DHSPT, use one half-tone pixel to store one data bit. In DHST, N data bits are hidden at N pseudo-random locations by forced toggling. That is, when the original half-tone pixel at the pseudo-random locations differs from the desired value, it is forced to toggle. This method results in undesirable clusters of white or black pixels. In the detection process, the data is simply read from the N pseudo-random locations. In DHPT, a pair of white and black pixels (instead of one in DHST) is chosen to toggle at the pseudo-random locations. This improves over DHST by preserving local intensity and reducing the number of undesirable clusters of white or black pixels. DHSPT improves upon DHPT by choosing pairs of white and black pixels that are maximally connected with neighboring pixels before toggling.

The chosen maximally connected pixels will become least connected after toggling and the resulting clusters will be smaller, thus improving visual quality.

In [24], an algorithm called intensity selection (IS) is proposed to select the best location, out of a set of candidate locations, for the application of the DHST, DHPT and DHSPT algorithms. By doing so, significant improvement in visual quality can be obtained in the output images without sacrificing data hiding capacity. In general, the algorithm chooses pixel locations that are either very bright or very dark. It represents a data bit as the parity of the sum of the half-tone pixels at *M* pseudo-random locations and selects the best out of the *M* possible locations. This algorithm, however, requires the original grayscale image or computation of the inverse-half-toned image.

In [21], two data hiding techniques for digital half-tone images were described: *modified ordered dithering* and *modified multiscale error diffusion*. In the first method, one of the 16 neighboring pixels used in the dithering process is replaced in an ordered or pre-programmed manner. The method was claimed to be similar to replacing the insignificant one or two bits of a gray scale image, and is capable of embedding 4,096 bits in an image of size 256 x 256 pixels. The second method is a modification of the *multi-scale error diffusion* (*MSED*) algorithm for half-toning as proposed in [27], which alters the binarization sequence of the error diffusion process based on the global and local properties of intensity in the input image. The modified algorithm uses fewer floors (e.g. 3 or 4) in the image pyramid and displays the binarization sequence in a more uniform and progressive way. After 50% of binarization is completed, the other 50% is used for encoding the hidden data. It is feasible that edge information can be retained with this method.

3. DISCUSSION

Robustness to printing, scanning, photocopying, and facsimile transmission is an important consideration when hardcopy distributions of documents are involved. Of the methods described above, the line and word shifting approaches described in [3] to [8], and the method using intensity modulation of character parts [18] are reportedly robust to printing, scanning, and photocopying operations. These methods, however, have low data capacity. The method described in [17] reportedly can survive printing and scanning (re-digitization) if the strokes remain in the image. This method's robustness to photocopying still needs to be determined. The bounding box expansion method described in [10] is a robust technique, but further research is needed to develop appropriate document de-skewing technique for the method to be useful. The character spacing width sequence coding method described in [11] can withstand modest amount of document duplications.

The methods described in [12],[16],[14],[15],[19], and [21] to [24] are not robust to printing, scanning and copying operations but they offer high data embedding capacity. These methods are useful in applications when documents are distributed in electronic form, when no printing, photocopying, and scanning of hardcopies are involved. The method in [13] also has high embedding capacity. It offers some amount of robustness if the two thresholds are chosen sufficiently apart, but this also decreases image quality.

Methods based on character feature modifications require reliable extraction of the features. For example, the methods described in [17] and one of the two site-selection methods presented in [18] require reliable extraction of character strokes. The boundary modification method presented in [16] traces the boundary of a character (or connected-component), which can always

be reliably extracted in binary images. This method also provides direct and good image quality control. The method described in [19] was originally developed for facsimile images, but could be applied to regular binary document images. The resulting image quality, however, may be reduced.

A comparison of the above methods shows that there is a trade off between embedding capacity and robustness. Data embedding capacity tends to decrease with increased robustness. We also observed that for a method to be robust, data must be embedded based on computing some statistics over a reasonably large set of pixels, preferably spread out over a large region, instead of based on the exact locations of some specific pixels. For example, in the line shifting method, data is embedded by computing centroid position from a horizontal line of text pixels, whereas in the boundary modification method, data is embedded based on specific configurations of a few boundary pixel patterns.

In addition to robustness and capacity, another important characteristic of a data hiding technique is its "security" from a steganographic point of view. That is, whether documents that contain an embedded message can be distinguished from documents that do not contain any message. Unfortunately, this aspect has not been investigated in the literature. However, for any of the above techniques to be useful in a covert communication application, the ability of a technique to be indistinguishable is quite critical. For example, a marked document created using line and word shifting can easily be spotted as it has characteristics that are not expected to be found in "normal" documents. The block-based techniques and boundary-based technique presented in Sections 2.2 and 2.3 may produce marked documents that are distinguishable if they introduce too much irregularities or artifacts. This needs to be further investigated. A similar comment applies to the techniques presented in Section 2.4. In general, it appears that the development of "secure" steganography techniques for binary documents has not received enough attention in the research community and much work remains to be done in this area.

Table 1 summarizes the different methods in terms of embedding techniques, robustness, advantages/disadvantages, data embedding capacity, and limitations. Robustness here refers to robustness to printing, photocopying, scanning, and facsimile transmission.

4. CONCLUDING REMARKS

We have presented an overview and summary of recent developments in binary document image watermarking and data hiding research. Although there has been little work done on this topic until recent years, we are seeing a growing number of papers proposing a variety of new techniques and ideas. Research on binary document watermarking and data hiding is still not as mature as for color and grayscale images. More effort is needed to address this important topic. Future research should aim at finding methods that offer robustness to printing, scanning, and copying, yet providing good data embedding capacity. Quantitative methods should also be developed to evaluate the quality of marked images. The steganographic capability of different techniques needs to be investigated and techniques that can be used in covert communication applications need to be developed.

ACKNOWLEDGMENT

The authors would like to thank Mr. Richard Simard of Air Force Research Laboratory for reading the initial draft of the paper and providing us with helpful and valuable comments.

REFERENCES

1. M. Swanson, M. Kobayashi, and A. Tewfik, "Multimedia data embedding and watermarking technologies," *IEEE Proceedings*, vol. 86, No. 6, pp 1064-1087, June 1998.
2. J. D. Foley, A. van Dam, S. K. Feiner, and J. F. Hughes, *Computer Graphics: Principles and Practice,* 2nd Edition, Addison-Wesley, 1990.
3. S. H. Low, N. F. Maxemchuk, and A. M. Lapone, "Document identification for copyright protection using centroid detection," *IEEE Trans. on Comm.,* vol. 46, no. 3, Mar 1998, pp. 372-83.
4. N. F. Maxemchuk, and S. H. Low, "Marking text documents," *Proc. IEEE Intl Conf. On Image Processing,* October 1997.
5. S. H. Low, and N. F. Maxemchuk, "Performance comparison of two text marking methods," *IEEE Journal on Selected Areas in Communications,* Vol. 16 No. 4, May 1998.
6. S. H. Low, N. F. Maxemchuk, J. T. Brassil, and L. O'Gorman, "Document marking and identification using both line and word shifting," *Infocom 95,* IEEE Computer Society Press, Los Alamitos, Calif., 1995.
7. S. H. Low, A. M. Lapone, and N. F. Maxmchuk, "Document identification to discourage illicit copying," *IEEE GlobeCom 95,* Nov. 13-17 1995, Singapore.
8. Y. Liu, J. Mant, E. Wong, and S. H. Low, "Marking and detection of text documents using transform-domain techniques," *Proc. SPIE Conf. on Security and Watermarking of Multimedia Contents,* pp. 317-328, Jan 1999, San Jose, California.
9. I. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure spread spectrum watermarking for multimedia," *Proc. First Int. Workshop Information Hiding,* R. Anderson, Ed., Cambridge, U. K.: Springer-Verlag, May/June 1996, pp. 183-206.
10. J. Brassil and L. O'Gorman, "Watermarking document images with bounding box expansion," *Proc. 1st Int'l Workshop on Information Hiding,* Newton Institute, Cambridge, UK, May 1996, pp. 227-235.
11. N. Chotikakamthorn, "Document image data hiding techniques using character spacing width sequence coding," *Proc. IEEE Intl. Conf. Image Processing,*1999, Japan.
12. M. Wu, E. Tang, and B. Liu, "Data hiding in digital binary images," *Proc. IEEE Int'l Conf. on Multimedia and Expo,* Jul 31-Aug 2, 2000, New York, NY.
13. E. Koch, J. Zhao, "Embedding robust labels into images for copyright protection", *Proc. International Congress on Intellectual Property Rights for Specialized Information, Knowledge & New Technologies,* Vienna, Aug. 1995.
14. H-K Pan, Y-Y Chen, Y-C Tseng, "A secure data hiding scheme for two-color images", *IEEE Symposium on Computers and Communications,* 2000
15. Y. Tseng, and H. Pan, "Secure and invisible data hiding in 2-color images," *IEEE Symposium on Computers and Communications,* 2000.
16. Q. Mei, E. K. Wong, and N. Memon, "Data hiding in binary text documents" *SPIE Proc Security and Watermarking of Multimedia Contents III*, San Jose, CA., Jan. 2001.
17. T. Amamo and D. Misaki, "Feature calibration method for watermarking of document images," *Proc. 5th Int'l Conf on Document Analysis and Recognition*, 1999, pp. 91-94, Bangalore, India.
18. A. K. Bhattacharjya, and H. Ancin, "Data embedding in text for a copier system," *Proc. IEEE International Conference on Image Processing,* Vol. 2*,* 1999, pp. 245-249.
19. K. Matsui and K. Tanaka, "Video-steganography: how to secretly embed a signature in a picture," *Proc. of IMA Intellectual Property Project,* v.1, no. 1, pp. 187-206, 1994.
20. Z. Baharav, D. Shaked, "Watermarking of dither half-toned images," *Proc. of SPIE Security and Watermarking of Multimedia Contents,* Vol. 1. pp. 307-313, Jan. 1999.

21. H-C A. Wang, “Data hiding techniques for printed binary images,” *The International Conference on Information Technology: Coding and Computing,* April 2-4, 2001.
22. M. S. Fu and O. C. Au, “Data hiding for halftone images,” *Proc of SPIE Conf. On Security and Watermarking of Multimedia Contents II,* Vol. 3971, pp. 228-236, Jan. 2000.
23. M. S. Fu and O. C. Au, “Data hiding by smart pair toggling for halftone images,” *Proc. of IEEE Int’l Conf. Acoustics, Speech, and Signal Processing,* Vol. 4, pp. 2318-2321, 5-9, June, 2000.
24. M. S. Fu, O. C. Au, “Improved halftone image data hiding with intensity selection,” *Proc. IEEE International Symposium on Circuits and Systems,* Vol. 5, 2001, pp. 243-246.
25. K. T. Knox, “Digital watermarking using stochastic screen patterns,” *United States Patent Number 5,734,752.*
26. S. G. Wang, “Digital watermarking using conjugate halftone screens,” *United States Patent Number 5,790,703*
27. I. Katsavounidis and C. C. Jay Kuo, “A multiscale error diffusion technique for digital half - toning,” *IEEE Trans. on Image Processing,* Vol. 6, no. 3, pp. 483-490, Mar 1997.

Table 1. Comparison of Techniques

Ref	Techniques	Robustness	Advantages (+) / Disadvantages (-)	Capacity	Limitations
[3-7]	Line shifting	High		Low	Formatted text only
[3-7]	Word shifting	Medium		Low/Medium	Formatted text only
[10]	Bounding box expansion	Medium	- Sensitive to document skewing	Low/Medium	Formatted text only
[11]	Character spacing	Medium	+ Can be applied to languages with no clear-cut word boundaries	Low/Medium	Formatted text only
[12]	Fixed partitioning -- Odd/Even pixels	None	+ Can be applied to binary images in general	High	
[13]	Fixed partitioning – Percentage of white/black pixels	Low/Medium	+ Can be applied to binary images in general - Image quality may be reduced	High	
[14][15]	Fixed partitioning –Logical invariant	None	+ Embed multiple bits within each block + Use of a secret key	High	
[16]	Boundary modifications	None	+ Can be applied to general binary images w. connected components + Direct control on image quality	High	
[17]	Modification of horizontal stroke widths	Medium		Low/Medium	Languages rich in horizontal strokes only
[18]	Intensity modulations of sub-character regions	Medium		Medium	Grayscale images of scanned documents only
[19]	Run-length modifications	None	- Image quality may be reduced	High	
[20]	Use two-dithering matrices	None			Half-tone images only
[22] to [24]	Embed data at pseudo-random locations	None		High	Half-tone images only
[21]	Modified ordered dithering	None		High	Half -tone images only
[21]	Modified error diffusion	None		High	Half-tone images only