World Academy of Science, Engineering and Technology
International Journal of Computer and Information Engineering
Vol:2, No:10, 2008

# Key Based Text Watermarking of E-Text Documents in an Object Based Environment Using Z-Axis for Watermark Embedding

Mussarat Abdullah, and Fazal Wahab



***Abstract*—**Data hiding into text documents itself involves pretty complexities due to the nature of text documents. A robust text watermarking scheme targeting an object based environment is presented in this research. The heart of the proposed solution describes the concept of watermarking an object based text document where each and every text string is entertained as a separate object having its own set of properties. Taking advantage of the z-ordering of objects watermark is applied with the z-axis letting zero fidelity disturbances to the text. Watermark sequence of bits generated against user key is hashed with selected properties of given document, to determine the bit sequence to embed. Bits are embedded along z-axis and the document has no fidelity issues when printed, scanned or photocopied.

***Keywords*—**Digital Watermarking, Object Based Environment, Watermark, z-ordering.

## I. Introduction

IN today's environment of digital world where digital data traveling in electronic form in our daily lives which is made possible due to technological advancements further making it quite feasible and very economical to store, share and transmit multimedia data where ever and when ever needed. An important role is being played by the commercial applications and e-businesses over the internet and in their steady growth these are becoming more famous and are highly appreciated. Distribution of digital contents is increased tremendously in the late of the past decade, which creates and opens new doors of problems and the terms like digital rights management came into existence. In digital data distribution no one knows where and how his data can be misused and by whom. Most of the office work is documented**,** distributed and recorded in text form. The unauthorized use of electronic confidential text documents is not a big deal. The idea of using some dependable[2] method of watermarking that is capable of uniquely identifying the authenticated source (original owner) of a text document has much focus of people involved in the field of electronic publishing and printing industries. Due to the war field challenges and possible counter measures its is therefore some of the basic requirements[9] of a watermark to be imperceptible, holds capacity to contain watermark bits, secure, dependable and must survive attacks that can be applied to a text document.

Mussarat Abdullah is with Department of Computer Science, COMSATS Institute of Information Technology, Quaid Avenue, The Mall, Wah Cantt 47040 Pakistan (e-mail: mussarat@cuonline.net.pk).

Fazal Wahab is with Department of Computer Science and Engineering, BAHRIA University Islamabad campus, Shangrila road, Sector E-8, Islamabad 44000 Pakistan (e-mail: fwahhab@bci.edu.pk).

## II. Related Work

The recent work done in the field of text watermarking and data hiding techniques for binary document images can be classified according to one of the following embedding methods:

### *A. Text Line, Word, or Character Shifting*

This [1] class of shits a line, group of words or a group of characters by a small amount to embed data. They [3 to 8] are only applicable to the electronic documents with proper formatted 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.

### *B. Boundary Modifications*

In this [6] approach the watermark is embedded in the bounding box enclosing a group of words. 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 descanters. 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. 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.

### *C. Fixed Partitioning of the Image into Blocks*

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.

*D. Modification of Character Features*

In this [6] approach 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.

*E. Modification of Run-Length Patterns*

In this [6] approach it is proposed to embed data in the run-lengths of facsimile images. 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.

*F. Modifications of Half-Tone 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.

## III. Drawbacks of Discussed Techniques

In the above section some text watermarking schemes have been discussed. All of them do work in spatial domain but some how proved not much better when you have a very strong pattern or output matching scripts. The companies which are professional with document manipulation even take care of 1/1000 shift in a character, word or line. So with such scripting all of the above schemes failed. Besides their nature of introducing shifts they are slow as well. For instance technique discussed in [4] makes vertical and horizontal profiles of the input document which is pretty time consuming as each pixel is being scanned for two times checking the watermark capacity.

The technique that has been proposed is different in each above discussed aspect from techniques discussed in section II.

## IV. The Proposed Technique

The theme of research carried out lies around the text watermarking in an object based environment.

All of the earlier spatial domain schemes intentionally or unintentionally disturbs the document fidelity (although at a very low level) which is not a very good and recommended technique, as companies that are professional in document composition and manipulation have strong scripts that can even detect a word, character or line shift of 1/14400. It is therefore tried to introduce zero distortion to the document fidelity and keep the watermark as strong as it was never before.

In this research a new robust, reliable and secure text watermarking technique is being proposed.

The main theme of the research is around the object based environment, in which each text string (entered at a time) is treated as a separate text object. Each text object holds its set of attributes and properties.

The differentiating nature of this technique from others is its embedding channel. All existing schemes as discussed above have conventional channels to embed watermark, like inter words, inter characters, line spacing etc. This technique uses z-axis for watermark embedding and all operations.

## V. Embedding Module

An object based text document can be represented in an object based environment with the following function.

*f (xi,zj)* Where i = 1,2,3,4……n, shows the number of text objects present in an object based text document at any given time and j = 1,2,3,4……n, shows their distance along z-axis.

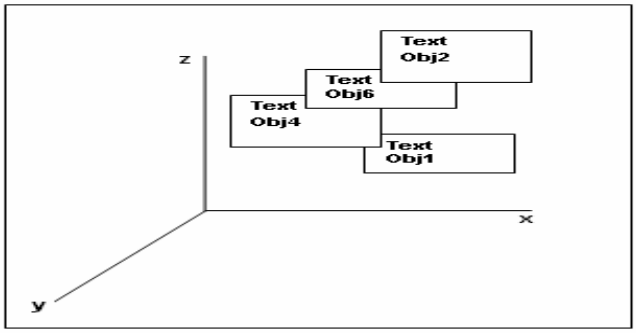


Fig. 1 Z- ordered text object view

Each input document has to go through a number of steps and phases to be completely watermarked and secured against the user input key. At each and every step some specific function is applied to the input document and the information is passed to the next phase for further processing. Key plays an important role to embed the watermark bits and to detect the watermark bits afterwards.

## VI. Analyzing Input

Analysis of input text document is the first and one of the most important steps of the text watermarking process. In this step the input document is analyzed for the nature of its contents and metadata is prepared about the input document according to certain requirements.

Basically the watermark engine is interested in the places where it can embed the watermark bits. This whole process is called analysis of input document. Here certain information about the input text document is obtained.

In the scheme proposed the information collected about the input document belongs to the z-ordering of the text objects in the input text document.

For instance metadata to be collected about the document in Fig. 1 can be graphically represented in Fig. 2.

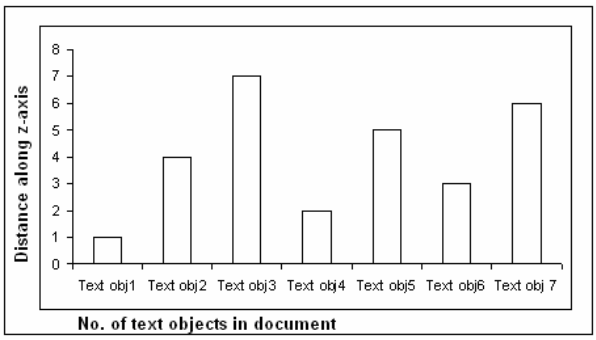


Fig. 2 Document analysis

Once this information is obtained the metadata is stored in internal structures and passed to the following phases for further purifications and processing.

## VII. Watermark Sequence Generation

Watermark bits are the pseudo random numbers generated against the key entered by the user. These bits are then passed through the parity check and hash of the objects to determine which bit to insert in which object.

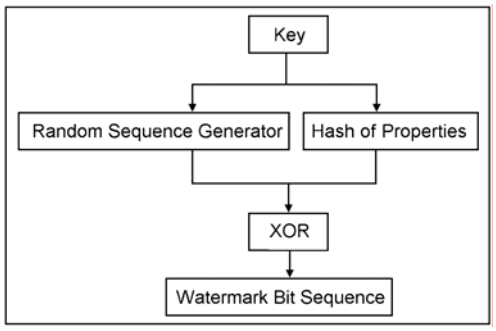


Fig. 3 Watermark sequence generator

After the hashing is applied and parity checks are performed the watermark is ready to embed.

## VIII. Embedding Engine

Watermark embedding engine works on the input calculated in some earlier steps. The input to embedding engine includes the followings.

1) Generated random sequence of bits
2) Document properties

The embedding engine then analyzes the input and embeds the watermark bits into the text document on the basis of properties it gets as input. The properties of the text document mainly include the z-ordering of the text objects besides other properties like size, position, content etc.

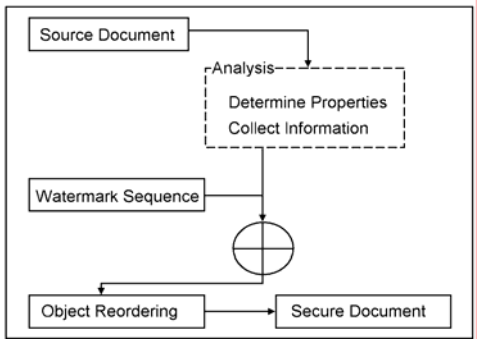


Fig. 4 Watermark embedding

The process of bit sequence embedding works on predefined reversible rules so that the bits can be recovered on detection process. The embedding rules are as under.

$EO(xi) = rem( f(xi,zj) /2 )$

$EK = rem( f(k) /2 )$

$E(xi) = if(EO(xi) \&\& EK) ? (f(xi,zj+1)) : (f(xi,zj))$

Where i = 1,2,3,4……n, shows the number of text objects present in an object based text document at any given time and z = 1,2,3,4……n, shows their distance along z-axis. Note that $(f(xi,zj+1)) <= n \&\& (f(xi,zj+1)) >= 0$.

Each text object's z-order is checked and determined is it even or odd. Then mod operation is performed on the key sum and z- order of the object. And it makes 4 possible combinations. On the basis of which it is decided whether a 1 or 0 can be inserted in current text object. If one of 1 and 0 cant be embedded the object is skipped and next object is analyzed.

TABLE I
Bits Embedding Criteria

| Serial | Object's z-order | Key sum mod z-order | Embed |
|---|---|---|---|
| 1 | Even | Even | 0 |
| 2 | Even | Odd | 1 |
| 3 | Odd | Even | 1 |
| 4 | Odd | Odd | 0 |

If result of processing lies in one of the 2, 3 categories 1 can be embedded in that object. In this case object's z-order is increased by 1. And if the results of processing lies in the category 1 or 4 it is assumed a can be embedded or is already embedded here.

Embedding can be shown mathematically as under.

$$E = \begin{cases} RE \varepsilon (1, 4) & 0 \\ RE \varepsilon (2, 3) & 1 \end{cases} \quad (1)$$

## IX. Detection Engine

In detection process steps VI, VII and other pre requisite steps are repeated. For the detection of embedded bits a similar kind of criteria is defined just as we did for bits embedding.

TABLE II
BITS EXTRACTION CRITERIA

| Serial | Object's z-order | Key sum mod z-order | Detect |
|---|---|---|---|
| 1 | Even | Even | 0 |
| 2 | Even | Odd | 1 |
| 3 | Odd | Even | 1 |
| 4 | Odd | Odd | 0 |

If result of processing lies in one of the 2, 3 categories 1 can be extracted from current object. In this case object's z-order is decreased by 1. And if the results of processing lie in the category 1 or 4 then 0 bit is detected.

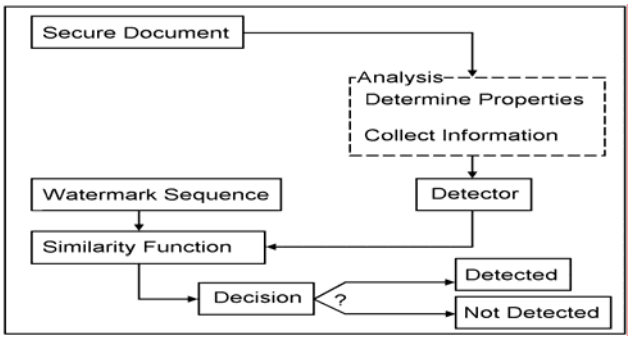


Fig. 5 Watermark extractor

After completing the bit sequence from all objects and combining it, it is passed to the proceeding module.

## X. Decision

The similarity function is used to determine the presence of watermark in the given document against the user key. If the generated bits and extracted bits are matched then it is considered that the watermark is present, otherwise not.

## XI. Conclusion

In this research a new area of the electronic text document watermarking is explored. The idea of object based document watermarking has its own appealing potential to work in. Watermark bits embedded through a secure robust and a recoverable mechanism makes this technique different from existing. Embedding of watermark along z-axis leaves no fidelity change to the electronic text document when printed, photocopied or displayed on monitor.

## References

[1] Huijuan Yang and Alex, C. Kot, "Text Document Authentication by Integrating Inter Character and Word Spaces Watermarking", The 2004 IEEE Inter-national Conference on Multimedia and Expo. June 26-30, 2004.

[2] M. Wu, E. Tang and B. Liu, "Data hiding in Digital Binary images", IEEE international Conf. on Multi-media, 2000

[3] S. Low and N.F. Maxemchuk, "Performance Comparison of two Text Marking Methods", IEEE, vol.16, pp.561-572, May 1998.

[4] Utilization of Maximum Data Hiding Capacity in Object-based Text Document Authentication "Imtiaz Awan, S.A.M. Gilani, and S.A. Shah" Faculty of Computer Science & Engineering GIK Institute of Engineering Sciences & Technology Topi, Pakistan.

[5] A Text Watermarking Algorithm based on Word Classification and Inter-word Space Statistics Young-Won Kim*, Kyung-Ae Moon, and Il-Seok Department of Computer Science, Chonbuk National University, KoreaS/W Contents Technology Department, Computer & Software Research Laboratory, ETRI,

[6] Recent Developments in Document Image Water-marking 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 Labora-tory.

[7] 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.

[8] Watermarking Electronic Text Documents Containing Justified Paragraphs and Irregular Line Spacing Adnan M. Alattar and Osama M. Alattar Digimarc Corporation.

[9] "Watermarks and Text Transformations in Visual Document Authentication" Thorsten Herfet Tele-communications Lab, Saarland University.