DOUBLE CROSS-RATIO METHODOLOGY FOR ENHANCING ZERO WATERMARK HIDING CAPACITY, ROBUSTNESS AND ORIGINAL-TEXT VERIFICATION PERFORMANCE OF MULTILANGUAGE TEXT-IMAGE WATERMARKING

WIYADA YAWAI

A THESIS SUBMITTED IN FULFILLMENT
OF THE REQUIREMENT FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY IN COMPUTER SCIENCE
FACULTY OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
2014

KMITL-2014-SC-D-002-014

หัวข้อวิทยานิพนธ์	วิธีครอสเรโชแบบสองเส้นสำหรับการเพิ่มความจุในการซ่อน ความทนทาน และสมรรถนะในการตรวจสอบความถูกต้องของข้อความต้นฉบับของการฝังลายน้ำแบบซีโร่บนภาพตัวอักษรแบบหลากหลายภาษา
นักศึกษา	นางสาววิยดา ยะไวทย์
รหัสประจำตัว	49062910
ปริญญา	ปริญญาดุษฎีบัณฑิต
สาขาวิชา	วิทยาการคอมพิวเตอร์
พ.ศ.	2557
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผศ.ดร.นวลสวาท หิรัญสกลวงศ์

บทคัดย่อ

งานวิจัยนี้มีจุดมุ่งหมายเพื่อค้นหาวิธีการฝังลายน้ำบนภาพข้อความทุกประเภททุกภาษาที่มีประสิทธิผล โดยมีเป้าหมายสูงสุดคือ หาวิธีฝังลายน้ำแบบไม่ถูกมองเห็นหรือถูกตรวจพบ หรือเป็นแบบซีโร่ที่การฝังไม่กระทบต่อต้นฉบับ เพื่อให้มีความทนทาน มีความจุในการซ่อนลายน้ำมากขึ้น และตรวจสอบการเปลี่ยนแปลงภาพข้อความต้นฉบับไม่ว่าด้วยวิธีใดๆ ได้ในเวลาเดียวกัน ซึ่งนี่เป็นข้อจำกัดของวิธีการฝังลายน้ำโดยส่วนใหญ่ การทดลองกับภาพข้อความของภาษาไทย อังกฤษ จีน และอาหรับแสดงให้เห็นว่าทฤษฎีครอสเรโชเป็นหนึ่งในวิธีที่มีประสิทธิผลที่สุดที่เอาชนะข้อจำกัดเหล่านี้ได้ โดยเฉพาะอย่างยิ่งการประยุกต์ใช้ครอสเรโชของจุดร่วม 4 จุดบนเส้นตรงเดียวกันกับเส้นแนวนอนและแนวตั้งซึ่งลากเส้นเสมือนผ่านเส้นโครงร่างตัวอักษร ซึ่งเรียกว่า “วิธีครอสเรโชแบบสองเส้น” เพื่อวางตำแหน่งและตรวจจับลายน้ำแบบซีโร่บนจุดตัดของเส้นลากกับตัวอักษรได้อย่างละเอียด วิธีนี้ได้พิสูจน์ให้เห็นอย่างชัดเจนว่ามีประสิทธิผลในการสร้างลายน้ำแบบซีโร่ที่มองไม่เห็นและช่วยทำให้การแกะรอยและรวบรวมข้อมูลตำแหน่งการฝังลายน้ำแบบซีโร่ได้แม่นยำมากยิ่งขึ้น และยังช่วยเสริมความทนทานของลายน้ำและเพิ่มสมรรถนะในการตรวจพิสูจน์ความจริงแท้ของภาพข้อความด้วยแม้จะถูกโจมตีและเปลี่ยนแปลงด้วยการบิดเบือนทางเรขาคณิตและแก้ไขต่อเติมข้อความด้วยวิธีการต่างๆ ก็ตาม ครอสเรโชแบบสองเส้นที่ประยุกต์ใช้กับเส้นลากแนวนอนและแนวตั้งนี้มีความแม่นยำในการฝังลายน้ำมากกว่าครอสเรโชแบบเส้นเดียว ทั้งนี้เพราะเส้นครอสเรโชสองเส้นสามารถใช้ตรวจสอบไขว้ซึ่งกันและกันได้ ในการฝังลายน้ำแบบซีโร่ จุดตัดของเส้นแนวนอนและแนวตั้งอ้างอิงตัดผ่านเส้นโครงร่างตัวอักษรจะถูกใช้เป็นจุดฝังลายน้ำแบบซี

โร่ โดยหนึ่งจุดตัดคือหนึ่งพิกเซลของลายน้ำแบบซีโร่ แต่ละจุดตัดถูกกำหนดให้มีค่าเท่ากับหนึ่งบิตของการซ่อนลายน้ำแบบซีโร่ ฉะนั้น ยิ่งลากเส้นแนวนอนและแนวตั้งอ้างอิงมากเท่าใดก็จะทำให้ยิ่งเพิ่มความจุในการซ่อนบิตสำหรับการฝังลายน้ำมากขึ้นเท่านั้น ซึ่งในที่นี้ได้คิดค้นเส้นตัดแบบ 11 ชั้น หรือ หลายชั้นขึ้นมาเพื่อเพิ่มจำนวนจุดตัดให้ได้มากยิ่งขึ้น ในขณะที่ความสามารถนี้เป็นจุดด้อยของวิธีฝังลายน้ำบนภาพข้อความส่วนใหญ่ นอกจากนี้แล้ว ถ้าตำแหน่งของจุดตัดถูกเก็บข้อมูลไว้ทั้งหมด จุดเหล่านี้ก็จะทำหน้าที่เป็นจุดอ้างอิงที่แสดงถึงตัวอักษรต้นฉบับสำหรับการพิสูจน์ถึงความจริงแท้ของตัวอักษรทั้งหมดหลังถูกแก้ไขอีกด้วย จากการทดลอง พบว่า เส้นตัดทั้ง 11 ชั้นนี้เหมาะอย่างยิ่งในการนำไปประยุกต์ใช้กับเอกสารสำคัญที่มีจำนวนหน้าไม่มากนัก และหนังสือสัญญาที่มีความสำคัญมาก ๆ แต่หากเป็นเอกสารทั่วไปที่มีจำนวนหน้ามาก การฝังและการพิสูจน์ทราบโดยใช้เพียง 5 ชั้น (ชั้นที่ 7 ถึง 11) ของเส้นหลักทั้งแนวตั้งและแนวนอนที่เป็นตาข่าย และการขยับตัวของตาข่าย ก็มีความไว้วางใจได้สูงถึง 99 เปอร์เซ็นต์แล้ว ทั้งนี้ไม่รวมถึงการแก้ไขลำดับ และการลบข้อมูล

คำสำคัญ : ครอสเรโช, ครอสเรโชแบบสองเส้น, ลายน้ำแบบซีโร่, ลายน้ำ, ภาพตัวอักษร

Thesis Title	Double Cross-ratio Methodology for Enhancing Zero Watermark Hiding Capacity, Robustness and Original-Text Verification Performance of Multilanguage Text-Image Watermarking
Student	Miss Wiyada Yawai
Student ID.	49062910
Degree	Doctor of Philosophy
Program	Computer Science
Year	2014
Thesis Advisor	Asst.Prof. Dr. Nualsawat Hiransakolwong

ABSTRACT

This research aims to find the most effective method to watermark any text images of any languages. The ultimate goal is to create the absolutely invisible or zero watermarks with more robustness, higher hiding capacity and simultaneously identifying any change of its original texts which generally are the limitations of most text-image watermarking methods. The testing with many text images of Thai, English, Chinese and Arabic languages shown that the cross-ratio theory is one of the most effective method to breakthrough these limitations, especially applying the cross ratio of four collinear points for both horizontal and vertical lines which virtually run across text-image character-skeleton lines, so called the double cross-ratio methodology, for finely marking and detecting the zero watermarks on these line-character intersection points. This method is strongly proved its effectiveness of creating invisible-zero watermarks and enhancement of precisely tracing and retrieving the zero-watermarking positions and additionally increasing the watermark robustness and text-image integrity verification performance, even though it has been attacked and tampered with some geometric distortions and randomly text modifications. These double cross-ratio applying for horizontal and vertical lines have watermarking precision more than the single cross-ratio line, because two cross-ratio lines can be used as the cross-checking lines of each other. To embed zero watermark, the intersection points which all reference horizontal and vertical lines virtually run across all character-skeleton lines will be used as the zero watermark embedding positions, one intersection for one specific zero watermark pixel. Each intersection point is defined as one hiding-bit of zero watermark, so that the more reference horizontal and vertical lines plotting, the more points of intersection would be got and increasing the hiding-bit capacity for watermark embedding while this capability is the disadvantage of many text-image watermarking methods. Therefore, the 11 or multiple layers of crossed lines were created for enormously increasing these intersection points. In addition, if all positions of intersection points are collected, these points will act as identity reference points of all original characters for proving their integrities whether they are modified. From our testing, it has suggested that 11 layers of crossed lines are suitable for applying for only a small amount of sensitive documents, especially the important contract documents, but in case of general documents with a lot of pages, only five layers (7th to 11th layers) of the main grid layer and shifted grid layers are enough to apply at the reliability level up to 99 percent, except for the text reordered and deleted manipulations.

Keywords: cross ratio, double cross-ratio methodology, zero watermark, watermarking, text image

ACKNOWLEDGEMENTS

First and foremost, I offer my sincerest gratitude to my adviser, Asst. Prof. Dr. Nualsawat Hiransakolwong, who has supported me throughout my thesis with her patience and knowledge whilst allowing me the room to work in my own way. I attribute the level of my Doctoral degree to her encouragement and effort and without her this thesis, too, would not have been completed or written.

Second, I would like to give my special thanks to all lecturers and officers of Department of Computer, Faculty of Science, King Mongkut's Institute of Technology Ladkrabang, who have strongly supported me to fully finish my thesis.

In addition, I would like to offer my sincerest gratitude to the director of Department of Computer, the dean of Faculty of Science and Technology and the president of Nakhon Ratchasima Rajabhat University who have funded and allowed me to leave for this study.

Finally, I thank my brothers, sisters, especially my parents for supporting me throughout all my studies.

Wiyada Yawai

TABLE OF CONTENTS

TABLE OF CONTENTS (cont.)

TABLE OF CONTENTS (cont.)

LIST OF TABLES

LIST OF FIGURES

LIST OF FIGURES (cont.)

Figures **Page**

LIST OF FIGURES (cont.)

Figures **Page**

LIST OF FIGURES (cont.)

CHAPTER 1

INTRODUCTION

1.1 Statements of Problem

At present, creating and embedding watermark in a text image is still has many limitations, especially the lower of its robustness against variety of attacks as well as its capacity to hide more volume of watermark data, according to the nature of content, language and text itself that mostly written in black color. This nature gives a little variation space to hide a big volume of invisible watermark data, hence let it so easy to be seen and destroyed by some simple attacks. A lot of research has tried to propose their algorithms for hiding a lot of watermark data in the text image but they could apply only with some specific language and still get only small hiding-bit data capacity and could not be strong enough to survive from those attacks. Therefore, this research aims to find a new concept for solving these problems and get rid of most sensitive weak points by applying the cross ratio theory together with the character-skeleton line intersection to build up the novel effective watermark robustness against some specific sensitive attacks, especially the projective-geometric distortion and text manipulation which rarely found in other research. Moreover, this applying also simultaneously increases watermark hiding-bit data capacity, makes watermark totally invisible without modifying the original text image, effectively watermarks any multi-language text image and sensitively identify any change of the original text integrity.

1.2 Research Objectives

The main objectives of this thesis are to

1. Find the novel method to effectively zero watermark any text images in any languages.
2. Get more remarkable hiding-bit data capacity of watermark.
3. Make zero watermark surviving from the specific sensitive attacks; affine transformation and text manipulation.
4. Identify any change of the original text image.

1.3 Scope and Limitation of The Study

The scopes and limitation of the study are:

1. Studying only the grayscale text image
2. Testing with Thai, English, Chinese and Arabic text images
3. Testing with text image size of 1240x1754 pixels and 150 dpi of resolution

1.4 Results

The benefits of the research are as follows:

1. To create the novel text-image zero watermark
2. To create the text-image zero watermark which robust to many possible attacks
3. To increase the hiding-bit data capacity of text-image zero watermark
4. To create the text-image zero watermark which can be applied to any text images in any languages
5. To create the text-image zero watermark which can be simultaneously detected the text copyright and integrity of some specific text images of sensitive documents such as contract papers, title deed or land certificates, etc.

1.5 Research Methodology

To apply the cross ratio theory of four collinear points for the horizontal and vertical lines which virtually run across text-image character-skeleton lines, so called the double cross-ratio methodology, for finely marking and detecting the zero watermarks on these line-character intersection points. This applying has watermarking precision more than the single cross-ratio line has, because two cross-ratio lines can be used as the cross-checking lines of each other. To embed zero watermark, the intersection points which all reference horizontal and vertical lines run across all character-skeleton lines will be used as the zero watermark embedding positions, one intersection for one specific zero watermark pixel. Each intersection point is defined as one hiding bit of zero watermark, so that the more reference horizontal and vertical lines plotting, the more points of intersection would be got and consequently increasing both robustness and hiding-bit data capacity for watermark embedding. In addition, if all positions of intersection points are collected, these points will act as identity reference points of all original characters for proving their integrities as well.

1.6 Organization of Thesis

The remaining chapters of this thesis are organized in the following way.

Chapter 2 provides background of digital text image watermarking and cross ratio of four collinear points.

Chapter 3 presents applying cross ratio for text image watermarking.

Chapter 4 contains experimental results; their results and analysis.

Chapter 5 presents conclusion and recommendation.

CHAPTER 2

LITERATURE REVIEWS

This chapter is divided into two sections, Section 2.1 provides background of digital watermark and cross ratio theory which mainly applied to this research and Section 2.2 is the related works reviewed for knowledge representation.

2.1 Backgrounds

This section addresses background of digital watermark and an overview of its methodology. In addition, theoretical background of cross ratio theory of four collinear points and Hough transform method are described. These definitions, theories and methods are applied in this research to enhance overall performances of the digital watermarking process.

2.1.1 Digital Watermark

Digital watermarking is one of the processes of hiding data for protecting copyright of digital media either in forms of audio, video, text, etc. There are two categories of watermarking; visible watermarking and invisible watermarking. The major purpose of watermarking is to protect copyright of media through creating various forms of obstructions to violators.

An illicit re-distribution and reproduction of information content (s) and copyright violations can be avoided by applying text watermarking methods. Watermarking solutions for image, audio and the video images are already in place and a number of research groups are working in these domains. However, the watermarking methods for text documents, especially text image, are very scarce. Text document watermarking is the most difficult kind of watermarking; this is due largely to the relative lack of redundant information in a text file, especially text image, as compared with a picture or a sound file. The structure of text documents is identical with what we observe, while in other types of documents such as in picture, the structure of document is different from what we observe. Therefore, in such documents, we can hide information by introducing changes in the structure of the document without making a notable change in the concerned output.

In image watermarking imperceptibility is achieved by exploiting the "redundancy" in images and the limitations of the human visual system. Similarly in case of audio and videos, redundancy of sound waves and video frames can be utilized to embed watermark. We need to find similar redundancy in text to hide information. The binary nature, block/line/word patterning, which provides demarcation between foreground and background, text meaning, structure, style, language rules, are some of the eminent addressed in any text watermarking algorithm. However, it still has the limitations of the hiding data process, especially for a black and white text image file. Main limitations are making digital watermark invisible on text and increasing a hiding-bit capacity to have enough volume for embedding all watermark data they want.

Mostly, text image is embedded with the visible watermarking for indicating the ownership of such media. In case of making invisible watermark, a text image must be embedded with slightly hiding data in order to difficultly observe. The ideal solutions for watermarking should be easily processed, robust, and imperceptible.

2.1.2 Watermarking Techniques for Text Documents [1]

A text, being the simplest mode of communication and information exchange, brings various challenges when it comes to copyright protection. Any transformation on text should preserve the meaning, fluency, grammaticality, writing style and value of the text. Short documents have low capacity for watermark embedding and are relatively difficult to protect. Text watermarking algorithms are also dependent on text size, its language, rules, grammar, conventions and writing styles. The description of work done in each category of text watermarking is as follows:

2.1.2.1 Image-based Techniques

In this approach towards digital text watermarking, the text image is used to embed the watermark bits. Text should be treated as text instead of an image. The approach is not resistive to text reproduction (re-typing) attack. With effective and increasing use of OCR (Optical Character Recognition) nowadays, these methods are totally a failure. The use of OCR can destroy the changes made by shifting words upward and downwards, to the document margins, to the fonts, serif and features of the text. Also, watermark can easily destroyed by a simple "copy paste to notepad" attack. In this way, all the changes made in text structure, feature and font to embed watermark will get destroyed.

2.1.2.2 Syntactic Techniques

Text is composed of sentences. Sentences are composed of words, and words can be nouns, verbs, articles, prepositions, adjectives, adverbs etc. Sentences have different syntactic structure which depends on language and its conventions. Applying syntactic transformations on text structure to embed watermark has also been one of the approaches towards text watermarking in the past. The syntactic solutions are more applicable to agglutinative languages like Korean, Turkish and Arabic than English language.

2.1.2.3 Semantic Techniques

The semantic watermarking schemes focus on using the semantic structure of text to embed the watermark. Text contents, like verbs, nouns, prepositions, words spelling, acronyms, sentence structure, grammar rules, etc. are exploited to insert watermark in the text.

The text watermarking, based on semantics, is language dependent where language is not something static. With the passage of time, language varies and hence the security and copyright solution provided by digital watermarking based on semantic will have limited strength. The semantic techniques for digital watermarking use natural language processing algorithms to analyze text meaning and to perform transformation. NLP is an immature area of research; hence, text watermarking using semantics does not give a practical and complete text watermarking solution. The synonym based techniques are not resilient to the random synonym substitution attacks. Also, synonyms may not always be giving the exact meaning of the word hence destroying the value of the text. The sensitive nature of some documents e.g. legal documents, poetry, and quotes do not allow us to make semantic transformations randomly
because in these forms of the text, a simple transformation sometimes destroys both the semantic connotation and the value of text.

2.1.2.4 Structural Techniques

Structural schemes of text watermarking are the recently used watermarking approach which uses text structures to embed watermarks. In this scheme, text is not modified when the watermark is embedded in to it. These types of text watermarking schemes are robust zero watermarking [2], [3]. A text watermarking technique to protect copyrights of text documents by using existence of double letter (aa-zz) in the text, have been proposed. The algorithm is a combination of encryption, steganography and watermarking, and provides a robust solution for text watermarking problem. To overcome the shortcoming of, another algorithm using double letters and preposition has been proposed. The structural algorithms are not applicable to all types of text documents and the algorithm use an alphabetical watermark [4].

2.1.3 Zero watermarking technique

A zero-watermarking algorithm is the one which does not change the characters of original data, but utilize the characters of original data to construct original watermark information. As a new digital watermark system, zero watermarking can resolve conflict between imperceptibility and robustness of digital watermark [5]. Thus, any watermarking algorithm that does not change the characters of original data, but utilize the characters of original data to construct original watermark information, such as syntactic and structural techniques mentioned above, can also be classified as a such type of zero-watermarking algorithm.

2.1.4 The Cross Ratio of Four Collinear Points

The cross-ratio is a basic invariance in projective geometry (i.e., all other projective invariance can be derived from it). Here brief introduction to the cross-ratio invariance property is given.

Let A, B, C, D be four collinear points (Three or more points A, B, C,... are said to be collinear if they lie on a single straight line [6]) as shown in Fig. 2.1. The cross-ratio is defined as the "double ratio" in Eq. (2.1).

$$(A,B;C,D)=\frac{AC\cdot BD}{BC\cdot AD} \tag{2.1}$$

where all the segments are thought to be signed. The cross-ratio does not depend on the selected direction of the line ABCD, but does depend on the relative position of the points and the order in which they are listed. Based on a fundamental theory, any homography preserves the cross-ratio. Thus central projection, linear scaling, skewing, rotation, and translation preserve the cross-ratio [7]. This is the main reason that it has been efficiently applied to embed and detect any watermark on any text image.

Figure 2.1 Collinear points A, B, C, and D

2.1.5 The Hough Transform Method

The Hough transform method involves transforming each of the figure points into a straight line in a parameter space. The parameter space is defined by the parametric representation used to describe lines in the picture plane. Hough chose to use the familiar slope-intercept parameters, and thus his parameter space was the two-dimensional slope-intercept plane [8].

The set of all straight lines in the picture plane constitutes a two-parameter family. If we fix a parameterization for the family, then an arbitrary straight line can be represented by a single point in the parameter space. For reasons that become obvious, we prefer the so-called normal parameterization. As illustrated in Fig. 2.2, this parameterization specifies a straight line by the angle θ of its normal and its algebraic distance ρ from the origin. The equation of a line corresponding to this geometry is defined as in Eq. (2.2).

$$\rho = x_i \cos\theta + y_i \sin\theta \qquad (2.2)$$

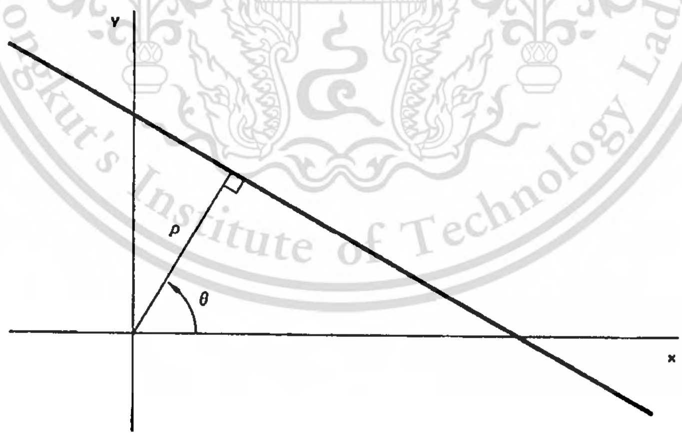


Figure 2.2 The normal parameters for a line.

Suppose now, that we have some set { $(x_1, y_1), \ldots, (x_k, y_k)$ } of n figure points and we want to find a set of straight lines that fit them, as shown in Fig. 2.3. Thus, this method can be applied to helpfully detect angles and edges of any scanned watermark-embedded text document during watermark detecting process.

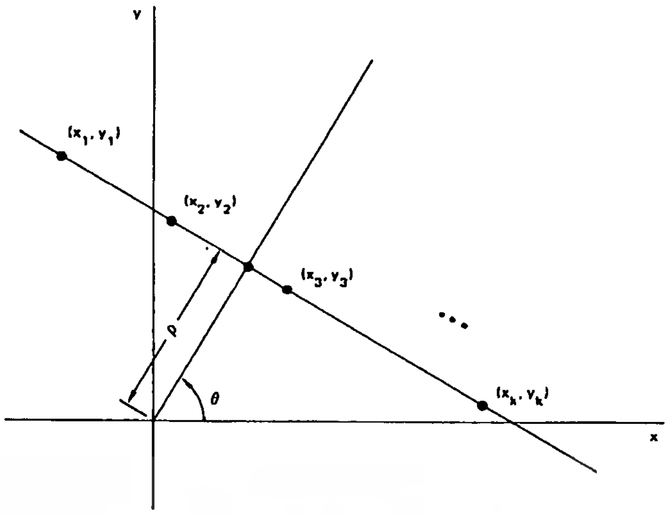


Figure 2.3 Projection of collinear points onto a line.

According to Eq. (2.2), therefore possible to associate with each line of the image a pair (ρ, θ) which is unique if $\theta \in [0, \pi)$ and $\rho \in \Re$, or if $\theta \in [0, 2\pi)$ and $\rho \geq 0$. The (ρ, θ) plane is sometimes referred to as Hough space for the set of straight lines in two dimensions.

Applying the Hough Transform to a set of edge points (x_i, y_i) results in an 2D function $C(\rho, \theta)$ that represents the number of edge points satisfying the linear equation $\rho = x\cos\theta + y\sin\theta$. In practical applications, the angles θ and distances ρ are quantized, and we obtain an array $C(\rho_k, \theta_l)$. The local maxima of $C(\rho_k, \theta_l)$ can be used to detect straight line segments passing through edge points.

Let us consider a rectangle with vertices $P_1 = (x_1, y_1)$, $P_2 = (x_2, y_2)$, $P_3 = (x_3, y_3)$ and $P_4 = (x_4, y_4)$, with P_1P_2 and P_3P_4 being parallel sides with length a, as well as P_2P_3 and P_4P_1 with length b. Also, let us assume that the origin of the coordinate system is located in the center of the rectangle, as shown in Fig. 2.4.

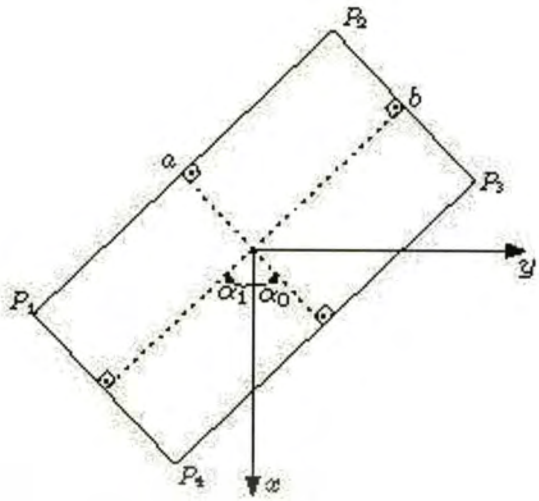


Figure 2.4 A rectangle centered at the origin of the coordinate system.

The image of this rectangle in the Hough Space as shown in Fig. 2.5. As expected, there are four peaks, labeled as $H_1 = (\rho_1, \theta_1)$ $H_2 = (\rho_2, \theta_2)$, $H_3 = (\rho_3, \theta_3)$ and $H_4 = (\rho_4, \theta_4)$, that correspond to the four sides of the rectangle (P_2P_3, P_1P_4, P_3P_4 and P_1P_2, respectively) [9].

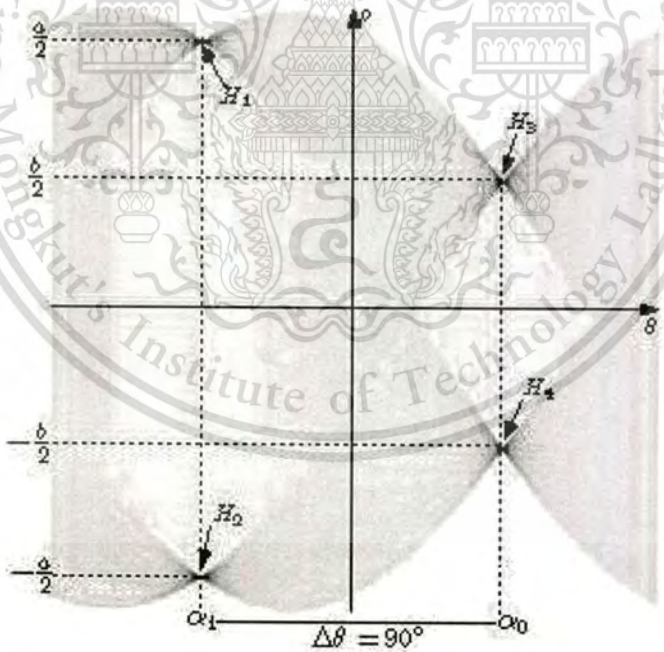


Figure 2.5 The Hough transform of a rectangle centered at the origin.

2.1.6 The Spread-spectrum Principle

The spread-spectrum principle has been applied to embed the watermark patterns by given the set of watermark embedding points $E_k = (x_k, y_k)$, $k = 1, \ldots M$, and each of the watermarking pattern bits w_k, $w_k \in \{1,-1\}$, $k = 1 \ldots. M$, each watermarking pattern bit is embedded to the original image as described in Eq. (2.3) [10].

$$I_e(x_m^k, y_n^k) = I(x_m^k, y_n^k) + \alpha w_k \qquad (2.3)$$

where $x_m^k = x_k + m$, m = -P, . . . , P, $y_n^k = y_k + n$, n = -Q, . . . , Q and α = strength of watermark.

2.1.7 The Correlation Coefficient Theory

The correlation coefficient theory, developed by Karl Pearson, was the index that use to measure correlation, Pearson's γ, of two variables, X_i and Y_i, as described in Eq. (2.4). The Pearson product-moment correlation coefficient is a dimensionless index, which is invariant to linear transformations of either variable [11].

$$\gamma = \frac{\sum (X_i - \overline{X})(Y_i - \overline{Y})}{\sqrt{\sum (X_i - \overline{X})^2 \sum (Y_i - \overline{Y})^2}} \qquad (2.4)$$

In the numerator, the raw scores are centered by subtracting out the mean of each variable, $\overline{X}$ or $\overline{Y}$, and the sum of cross-products of centered variables is accumulated. The denominator adjusts the scales of the variables to have equal units. Thus Eq. (2.4) describes γ as the centered and standardized sum of cross-product of two variables, X_i and Y_i.

2.1.8 The Aspect Ratio of a Rectangle

An aspect ratio is simply a numerical way of describing a rectangular shape. The aspect ratio of the standard television, for example, is 4:3. This means that the picture is 4 "units" wide and 3 "units" high. Interestingly, professional cinematographers tend to prefer a single number to describe screen shapes and reduce the familiar 4:3 television ratio down to 1.33:1, or just 1.33

[12], as shown in Fig. 2.6. Roughly speaking, the aspect ratio of an object is the ratio of its longest dimension to its shortest [13].

By default, the image is drawn as large as possible while still respecting its native aspect ratio. Otherwise, the image is positioned according to the arguments x and y and sized via width and height. If only one of width or height is given, then the aspect ratio of the image is preserved (and the image may extend beyond the current viewport) [14], as described by Eq. (2.5).

$$r = x : y = \frac{x}{y} \tag{2.5}$$

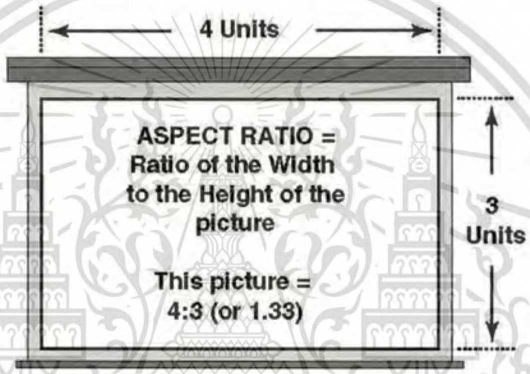


Figure 2.6 The example of the aspect ratio; 4:3, of a picture.

Where *r* stands for the aspect ratio, *x* represents the width and *y* represents the height of an image. Here, the aspect ratio would be applied for identifying type of language in order to enhance efficiency of the watermark detecting process.

2.2 Knowledge Representation

This research is particularly emphasized on applying the four-collinear-point cross ratio theory to create the robust watermark data embedded in the grayscale text image which must be survived and easily detected even it has been attacked in many possible ways, especially geometric distortion attacks which mostly not been explored in other text image watermarking researches. Most existing researches are focused on watermarking an electronic text or document file, instead of text image, of one specified language, instead of multi-language, and emphasized the watermark embedding technique instead of watermark robustness and its integrity verification. These existing document watermarking researches can be categorized, by their watermarking technique, into 3 techniques (see Table 2.1) as follows.

Table 2.1 Performance comparison of existing watermarking techniques

Technique	Principle	Applied to	Imperceptibility	Language	Approximately Capacity	Robustness	Integrity Verification?
Technique 1: Watermark embedding with text document physical layout/pattern/ structure rearrangement or space shifting	Text line shifting [15] , [16], [17],	Digital Text document, Text image	Visible	Multi-language	30 bits/A4 page	Easy to observe and re-manipulate low robust to document passing through document processing or OCR	No
	Inter-word space shifting (to left or right) [18], [19]	Digital Text document, Text image	Visible	Multi-language	720 bits/ A4 page	Easy to observe and re-manipulate or low robust to document passing through document processing or OCR	No
Technique 2: Embedding text watermark by text character/ letter feature modifying	Reducing or increasing length of letter for feature coding [18]	Text image	Visible	English	4,000 bits/A4 page	Low robust to document passing through document processing or OCR processing	No
	Adjusting character size [20]	Digital Text Document, Text image	Visible	Chinese	650 bits/A4 page	Low robust to document passing through document processing or OCR processing	No
	Character peak point distinction [21]	Text image	Visible	Persian and Arabic	900 bits/A4 page	Low robust to brightness and noise signal adding	
	Point shifting of letter "i" and "j" [21], [22]	Text image	Visible	English	600 bits/A4 page	Low robust to brightness and noise signal adding	No

Table 2.1 Performance comparison of existing watermarking techniques *(cont.)*

Technique	Principle	Applied to	Imperceptibility	Language	Approximately Capacity	Robustness	Integrity Verification?
	Changing the low-4 bits of RGB color components of characters [23]	Digital plain text	Invisible	Chinese	40,000 bits/A4 page	Robustness to resist deletion, modification attack etc.	No
Technique 3: Watermarking with semantic schemes or word/vowel substitution and using text structure itself.	Existence of double letter (aa-zz) in the text [4]	Digital plain text	Invisible (Zero Watermark)	English	50 bits/ A4 page	Could not be applied to all types of text documents	No
	Particular vowel creating sequent changing [24]	Digital plain text	Invisible	Thai	1.93 bits/text line or 57.9 bits/30 text-line/A4 page	Very low robust to document passing through document processing	No
	Substituting words in the text by synonyms [25]	Digital plain text	Invisible	English	870 bits/A4 page	Very low robust to document passing through document processing	No

First Technique: Watermark embedding with text document physical layout/pattern /structure rearrangement, such as shifting of lines [16] and words, particularly binding of word spacing, word shift coding or word classification [17], [18], [19]. This technique can be applied to both watermark electronic document file and text image. However, it has some disadvantages, for instance, line shifting technique of Brassil et al. will be low robust to document passing through document processing, page skewing/rotation; between -3° - $+3^{\circ}$, noise signal adding attack and a short text line. Another limitation of this process is that it can only apply to document with spaces between words, spacing of letters, shifting of baselines or line shift coding. Word shifting algorithm has also developed by Huang et al. [18], it's based on adjusting inter-word spaces that replace one sine wave. Signals have been encoded in phrases, waves, and frequencies of sine waves. The algorithm can do both non-blind modes for inserting signals, spaces between letter should also been changed, feature coding.

Min Du et al. [23] proposed a text digital watermarking algorithm based on human visual redundancy. According to that the human eye is not sensitive to the slight change for text color, watermarks were embedded by changing the low-4 bits of RGB color components of characters. This proposed method has good invisibility and robustness which depending on its redundancy. However, this research tested its robustness against word deleting and modifying only.

Second Technique: Embedding text watermark by text character/letter feature modifying, for example, Brassil, et al. [17] have used the letter adjusting by reducing or increasing length of letter; such as, increasing length of letters b, d, or h. For principle applying to extract hidden data out of document can be done by comparing hidden data document against original document. The limitation of this process is that the hidden data will be so little robust to document passing through document processing.

Applying arithmetic expression to replace characteristic of letter with close component has developed by W. Zhang et al. [19] which has applied arithmetic expression to replace characteristic of letter with close component (in square form) which the hiding is done by adjusting sizes of those characters in document file. This process is robust against attacks or destruction and unable to observe. The test referred that the mentioned hiding is durable and more difficult to observe than those of the line-shift coding, word-shift coding, and character coding but has not presented the robustness testing information in the research document. However, this research has only been applied to Chinese characters and subject to be further researched since the process has tested only the character replacement attack, has not yet been tested against durability to various forms of watermark attacks.

Shirali-Shahreza et al. [21] has applied changing of characters in Persian that a number most characters have their distinction in their peaks (Persian letter NOON) of these characters for hiding. Due to defects in using OCR in reading Persian and Arabic text image; therefore, reading of printed text from these characters for extracting hidden data is considered complicate, especially after attacking that has not yet been tested.

Suganya et al. [22] proposed to modify perceptually significant portions of an image to make the algorithm that watermark is hidden in the point's location of the English letter i and j.

first few bits are used to indicate the length of the hidden bits to be stored. Then the cover medium text is scanned to store a one, the point is slightly shifted up else it remains unchanged. However, this research did not refer to any robustness testing result.

Third Technique: Watermarking with semantic schemes or word/vowel substitution and using text structure itself; Topkara et al. [25] has developed a technique for embedding secret data without changing the meaning of the text by replacing words in the text by synonyms. This method deteriorates the quality of the document and a large synonym dictionary is needed.

Samphaiboon et al. [24] proposed a steganographic scheme for electronic Thai plain text documents that exploits redundancies in the way particular vowel, diacritical, and tonal symbols are composed in TIS-620, the standard Thai character set. The scheme is blind in that the original carrier text is not required for decoding. However, it can be used with only Thai text document and its watermark data is so easy to be destroyed by reediting with a word processing program.

The following presenting research has been focused on the text image, scanned, copied or saved from an original document paper, watermarking, with the cross ratio theory in collinear point type applying, in order to overcome the above mentioned limitations of text image watermarking, especially the hiding-bit capacity, robustness and language dependency. Thus, there are four main objectives of this study, the first is to study the new way to make zero watermark on text image that is not depending on some specific languages, the second is to increase remarkable hiding-bit capacity for virtually embedding digital watermark, the third is to test its robustness against various types of attack, particularly geometric distortion such as scaling, shearing and rotating and other manipulations; data compression, noise signal adding and brightness, contrast, scale, sharpness and blur adjusting, and the fourth is to find the way to verify its integrity.

CHAPTER 3

ANALYSIS OF DIGITAL WATERMARKING FOR TEXT IMAGE

3.1 The Experiment Concept and Plan

3.1.1 The Experiment Concept

According to the statements of problem and research objectives, in the Section 1.1 and 1.2 respectively, the concept of this watermarking experiment can be defined as follows.

1. Find the novel way to embed watermark with slightly modifying or without modifying any original text images of any languages, the later so called zero watermarking technique. Here, we plan to use space between text lines, words and characters and intersection points which reference lines run across the skeleton lines of each character.

2. Explore watermarking strength contributions, such as hiding bit capacity, robustness and integrity verification, of some specific crossed or grid lines; main and shifted horizontal (H), vertical (V) and double (H & V) lines, which run across the skeleton lines of each character and create considerable reference intersection points for effectively marking and detecting some zero watermarks throughout a text image page.

3. Prove the crossed-line intersection watermarking efficiency and effectiveness of hiding bit capacity and integrity verification and robustness after making some geometric attacks, such as sharpness, JPEG compression, blur, contrasting, shearing, rotation, noise signal adding and scaling, and text manipulations, including text adding/inserting, reordering and deleting of each type of crossed line.

4. Close some vital loopholes or weaknesses of each crossed-line intersection and build up their performance by combining all types of crossed line to create the multi-layers of zero watermarks which act like the finely crosscheck mesh.

3.1.2 The Experiment Plan

Regarding to the experiment concept mentioned above, In this section, a method for deriving such reference crossed-intersection points is detailed and the experiment plan has been defined as shown in Table 3.1.

3.2 The Cross Ratio Theory Definition and Application

To apply the cross ratio to digital image watermarking, three reference points are required. According to the Eq. (2.1) and Fig. 2.1, we can specify the definition for applying the four-collinear point cross ratio to efficiently embed and detect watermark as follows.

C_aC_b is line from an origin point C_a to a destination point C_b where $a = 1, 2, 3,4$ and $b = 1, 2, 3, 4$.

$Cr = (AC/BC)/(AD/BD) = (AC*BD)/(BC*AD)$

$R = (AC/AD)/Cr$

$BA = AD - CD/(1 - R)$

DsB is distance of *BA/AD* is equal the value of *BA/DA*.

Table 3.1 Experiment plan

Method	Watermark Type	Process	Measurement	Evaluation
Method 1: Applying the actual watermark embedded between text lines and the cross ratio theory to build up the robustness of text-image watermark	Actual watermark	Actually embedded between text lines	Four-Collinear-Point Cross ratio, Correlation coefficient	-Sharpness -JPEG compression -Blur -Contrasting -Shearing -Rotation - Noise signal adding -Scaling - Actual watermark embedding and detecting

Table 3.1 Experiment plan *(cont.)*

Method	Watermark Type	Process	Measurement	Evaluation
Method 2: Applying the single reference-line intersection of horizontal line and the cross ratio theory to increase the hiding-bit capacity and original-text verification performance of text-Image watermark with zero watermark embedded	Zero watermark	Virtually embedded along horizontal line run across character skeleton line	Four-Collinear-Point Cross ratio, Matching Percentage	-Sharpness -JPEG compression -Blur -Contrasting -Shearing -Rotation - Noise signal adding -Scaling -Text adding, deleting and reordering -Hiding-bit capacity - Zero watermark embedding and detecting

Table 3.1 Experiment plan *(cont.)*

Method	Watermark Type	Process	Measurement	Evaluation
Method 3: Applying the single reference-line intersection of additional vertical line and the cross-ratio theory to increase more watermark-hiding capacity and original-text verification performance of text-image watermark with zero watermark embedded	Zero Watermark	Virtually embedded along vertical line run across character skeleton line	Four-Collinear-Point Cross ratio, Matching Percentage	-Sharpness -JPEG compression -Blur -Contrasting -Shearing -Rotation - Noise signal adding -Scaling -Text adding, deleting and reordering -Text integrity verification - Watermark hiding-bit capacity - Zero watermark embedding and detecting

Table 3.1 Experiment plan *(cont.)*

Method	Watermark Type	Process	Measurement	Evaluation
Method 4: Applying the double reference-line intersection of vertical and horizontal lines and the double cross-ratio method to increase maximum watermark-hiding capacity and original-text verification precision of text-image watermark with zero watermark embedded	Zero Watermark	Virtually embedded along horizontal and vertical line which run across character skeleton line	Four-Collinear-Point Cross ratio, Matching Percentage	-Sharpness -JPEG compression -Blur -Contrasting -Shearing -Rotation -Noise signal adding -Scaling -Text adding, deleting and reordering -Text integrity verification - Watermark hiding-bit capacity - Zero watermark embedding and detecting

Table 3.1 Experiment plan *(cont.)*

Method	Watermark Type	Process	Measurement	Evaluation
Method 5: Applying the single horizontal, vertical and double reference-line intersection and the double cross-ratio method to create the multi-layer watermarking for increasing maximum watermark-hiding capacity and original-text verification precision of text-image watermark with zero watermark embedded.	Zero Watermark	Virtually embedded along horizontal and vertical line which run across character skeleton line	Four-Collinear-Point Cross ratio, Matching Percentage	-Sharpness -JPEG compression -Blur -Contrasting -Shearing -Rotation - Noise signal adding -Scaling -Text adding, deleting and reordering -Text integrity verification - Multi-layer hiding-bit capacity - Zero watermark embedding and detecting

3.3 The Aspect Ratio Definition and Application

Language type identification of any scanned watermark-embedded text document is very helpful for efficiently watermark detecting process. Here, the aspect ratio can be improvised for identifying our four-tested language characters; English, Thai, Chinese, and Arabic, of our tested text images. Based on the standard aspect ratio of the rectangle as described in Eq. (2.6), it can be improvised by reversing its ratio as shown in Fig. 3.1 and described by Eq. (3.1).

Figure 3.1 The example of width (x) and height (y) for calculating the aspect ratio.

$$Rr = \frac{1}{r} = y : x = \frac{y}{x} \tag{3.1}$$

where Rr stands for the reversed aspect ratio, x represents the width and y represents the height of each scanned character image of each language.

To certainly identify the specific language, the average reversed aspect ratio, ARr, of each character of each language must be separately clarified, as its identity range, by Eq. (3.2) below.

$$ARr = \frac{1}{n} * \sum_{i=1}^{n} Rr_i \tag{3.2}$$

where ARr is the average reversed aspect ratio and n is the number of reversed aspect ratios, Rr_i, of each character of each language.

3.4 Method 1: Applying The Actual Watermark Embedded between Text Lines and The Cross Ratio Theory to Build up The Robustness of Text-image Watermark

3.4.1 Embedding Scheme

This section shows how to use cross ratio for embedding watermark in text image. The method is described algorithmically below.

3.4.1.1 Predefine the set of cross-ratio values, to be used in subsequent steps.

3.4.1.2 Find the image center, as denoted by D_c, by using the line intersection formula [26] of the two diagonal lines of the image which the first one drawn from the point (x_1, y_2) to the point (x_4, y_4) and the second one drawn from the point (x_3, y_3) to the point (x_2, y_2), as described by Eqs.(3.3) ~ (3.4) below,

$$x_c = \frac{\begin{vmatrix} \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} & x_1 - x_2 \\ \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix} & x_3 - x_4 \end{vmatrix}}{\begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix}} \tag{3.3}$$

$$y_c = \frac{\begin{vmatrix} \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} & y_1 - y_2 \\ \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix} & y_3 - y_4 \end{vmatrix}}{\begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix}} \tag{3.4}$$

where $(x_1,y_1) = C_1$, $(x_2,y_2) = C_4$, $(x_3,y_3) = C_3$, $(x_4,y_4) = C_2$. In addition, C_i is the coordinate of the point $C_i, i = 1,\ldots,4$ (see Fig. 3.2).

3.4.1.3 From the above equations, x_c is the x-axis value of the point D_c of two-line intersection; C_1C_4 intersect C_2C_3, and y_c is the y-axis value of the same point and $|\,|$ denotes a determinant operator as shown in Fig. 3.2.

3.4.1.4 Find each of the primary-level watermark embedding points ($D_{LU,i}$ and $D_{LD,i}$) on the left diagonal line (see Fig. 3.2 (a)) as described by Eqs.(3.5) ~ (3.6) below. Those points can be identified by using two corner points of the left diagonal line (C_1 and C_4), in combination with the image center point D_c as shown in Fig. 3.2 (a) and the predefined cross-ratio values (C_r).

$$x = x_s + DsB \times (x_d - x_s) \tag{3.5}$$

$$y = y_s + DsB \times (y_d - y_s) \tag{3.6}$$

where (x,y) = ($x_{LU,i}, y_{LU,i}$), i = 1, 2, 1, ... M_{LU} , is the coordinate of the point $D_{LU,i}$, $A = C_1$, C_1 =(x_s , y_s), $B = D_{LU,i}$, $C = D_c$ and $D = C_4$, C_4 = (x_d , y_d). In addition (x,y) = ($x_{LD,i} , y_{LD,i}$), I = 1, ... M_{LD} , is the coordinate of the point $D_{LD,i}$, $A = C_1$, C_1 =(x_s , y_s), $B = D_{LD,i}$, $C = D_c$ and $D = C_4$, C_4 = (x_d , y_d).

3.4.1.5 Find each of the watermark embedding points ($D_{RU,i}$ and $D_{RD,i}$) on the right diagonal line (see Fig. 3.2(b)) by following the steps and equation similar to those detailed in Step 3. However, now the point A in Eqs. (3.5) ~ (3.6) represents the point C_2 while the point D now represents the point C_3. By using these substitutions, those embedding points are given by (X,Y) = ($x_{RU,i} , y_{RU,i}$), i = 1, ... M_{RU} , is the coordinate of the point $D_{RU,i}$, $A = C_2$, C_1 =(x_s , y_s), $B = D_{RU,i}$, $C = D_c$ and $D = C_3$, C_3 = (x_d , y_d). In addition (x,y) = ($x_{RD,i} , y_{RD,i}$), i = 1, ... M_{RD} , is the coordinate of the point $D_{RD,i}$, $A = C_2$, C_2 =(x_s , y_s), $B = D_{RD,i}$, $C = D_c$ and $D = C_3$, C_3 = (x_d , y_d).

3.4.1.6 For each pair of $D_{LU,i}$, $D_{RU,i}$, and $D_{LD,i}$, $D_{RD,i}$ levels, find an intersection, x_i , y_i , of crossed line of each level drawn across left side; $L_{LU,1}...L_{LD,1}$ and right side; $L_{RU,1}...L_{RD,1}$ of text image borders (see Fig. 3.3(a)); C_1C_3 and C_2C_4 , by applying Eqs. (3.3) ~ (3.4), where $(x_1,y_1) = D_{LU,i}$, $(x_2,y_2) = D_{RU,i}$, $(x_3,y_3) = D_{LD,i}$, $(x_4,y_4) = D_{RD,i}$

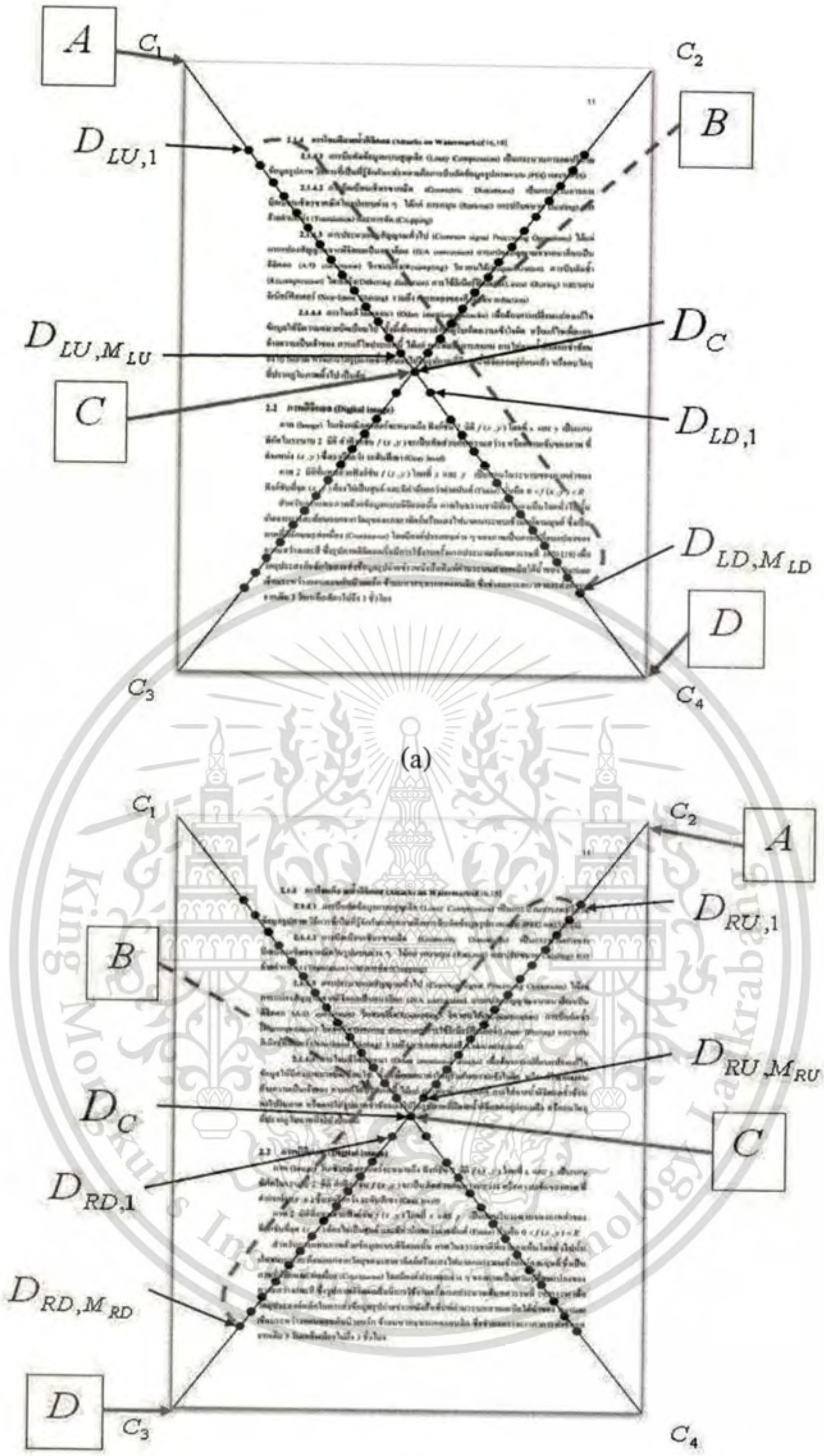
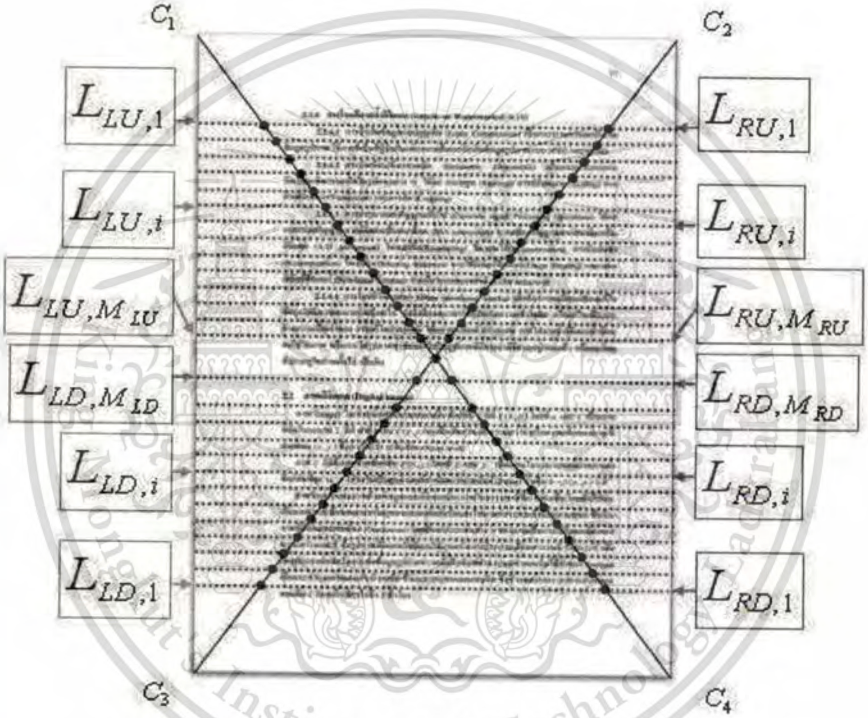


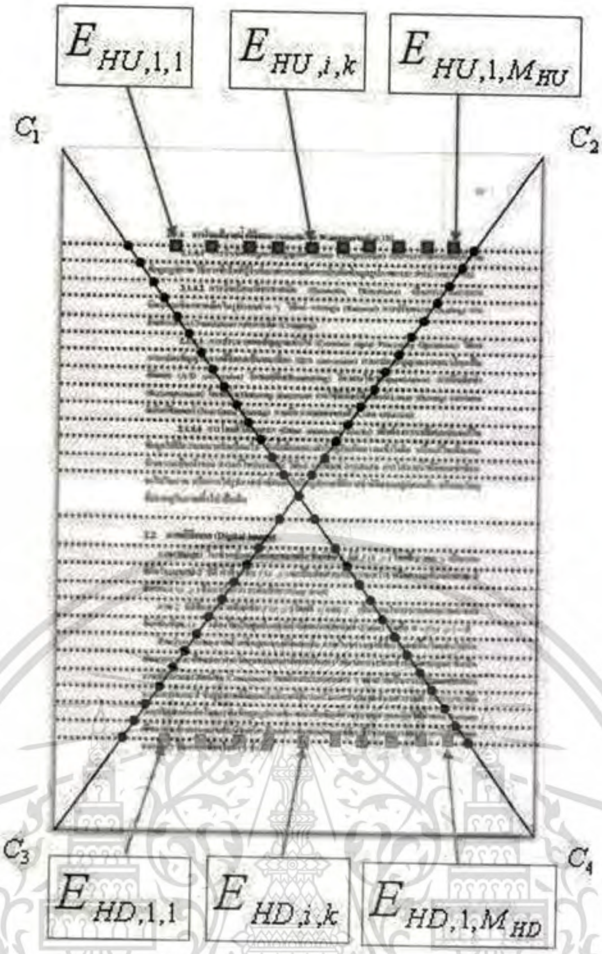
Figure 3.2 Notations of collinear points A, B, C, and D on the left (a) and right (b) diagonal line of the text image.

3.4.1.7 Find each of the watermark embedding points ($E_{HU,i,k}$ and $E_{HD,i,k}$) on the watermarked embedding lines (see Fig. 3.3 (b)). Eqs. (3.5) ~ (3.6) represents the embedding points ($E_{HU,i,k}$ and $E_{HD,i,k}$) where *(x,y)* = ($x_{LU,i}, y_{LU,i}$), i = 1, …. M_{LU} , is the coordinate of the point $D_{HU,ik}$, $A = L_{LU,i}$, $L_{LU,i}$ =(x_s, y_s), $B = E_{HU,i,k}$, $C = D_{RU,i}$ and $D = L_{RU,i}$, $L_{RU,i}$ = (x_d, y_d). In addition (X,Y) = ($x_{LD,i}, y_{LD,i}$), i = 1, …. M_{LD} , is the coordinate of the point $D_{HD,ik}$, $A = L_{LD,i}$, $L_{LD,i}$ =(x_s, y_s), $B = E_{HD,i,k}$, $C = D_{RD,i}$ and $D = L_{RD,i}$, $L_{RD,i}$ = (x_d, y_d).

3.4.1.8 From all watermark embedding points, embed the watermark patterns by means of a spread-spectrum principle [10] as described in Section 2.1.6.



(a)



(b)

Figure 3.3 (a) Notations of horizontal lines intersect with two diagonal lines and left and right border lines. (b) Notations of collinear points used for embedding watermark pattern bits.

3.4.2 Detecting Scheme

3.4.2.1 Identify Angle and Edge of a Watermark-embedded Text Image

To detect a watermark from the text image, the four image corner points must first be detected. This can be achieved by using the Hough transform method, as described under the Section 2.1.5, by which the four content borders, instead of the borders of a text image, would be detected. Three main steps for identifying angle and edge with MATLAB program are as follows.

1. Reverse a color of the original scanned watermarked text image by which the black color, grayscale lesser than 245, of text foreground would be changed into white color, 255

grayscale, and white color, grayscale equal or higher than 245, of text image background would be changed into black color at zero grayscale.

2. After that dilate all text character pixels until they have joined together and turned into the white-color strips. Then, erode or fill in the black color into the white-color strips for compensating the dilated pixels which expanded out of the actual boundary of text content borders.

3. Use Hough transform function to detect four angles and edges of four side borders of a text content and then find the Hough peaks for identifying the coordinates of four corners.

This method will give us the certain text image angles and edges so that identify all four reference corners, sides and its reference center point, D_C of a text image where the left and right diagonal lines intersected, as described under the Section 3.4.1.1. This main reference center would bring us to trace back to other reference lines and finally all watermark embedded lines, as explained in the Section 3.4.1.2, 3.4.1.3 3.4.1.4, 3.4.1.5, 3.4.1.6 and 3.4.1.7.

3.4.2.2 Embedded Watermark Identifying

Once the four corner points are detected watermark embedding points, on the text image $I_e^{'}$, must be identified. Each point can be calculated by using the method similar to that of the embedding stage (see Eqs (2.3)). To extract the values of the pixels corresponding to those watermark embedding points, a watermark can be detected by using the correlation coefficient detector as described in the Section 2.1.7.

3.5 Method 2: Applying The Single Reference-line Intersection of Horizontal Line and The Cross Ratio Theory to Increase The Hiding-bit Capacity and Original-text Verification Performance of Text-image Watermark with Zero Watermark Embedded

Since the Method 1 is based on the actual watermark embedded between text lines, so that they have given us considerably low hiding bit capacity of watermark secret data and so easy to be destroyed by copying process. Thus, in the Method 2, we have tried to solve these weak points by applying the horizontal crossed-line intersection which virtually ran across the skeleton lines of each character. Under this method, the text watermarking process is based on three sub-processes; the first one is to define watermark marking positions and its hiding-bit capacity with line intersection, the second one is to match the cross-ratio reference points with zero watermark marking positions for easily tracking the zero watermark marking points after a text image has been attacked and the last one is to detect the zero watermarks and verify its integrity with the matching percentage method.

3.5.1 Hiding-bit Capacity for Watermark Marking

Able to calculate the watermark hiding-bit capacity by following the procedure as described below.

3.5.1.1 Identify and draw the lines for the top, $L_{Top-Border}$, and bottom, $L_{Bottom-Border}$, borders of each text line (see Fig. 3.4).

3.5.1.2 Find and draw the main-middle line, $L_{Main-Middle}$, between the top and bottom border lines of each text line.

3.5.1.3 Find and draw the top-middle line, $L_{Top-Middle}$, between the top-border line and the main-middle line.

3.5.1.4 Find and draw the bottom-middle line, $L_{Bottom-Middle}$, between the bottom-border line and the main-middle line.

3.5.1.5 Detect each prominent intersection position of each horizontal line which runs across each text-character skeleton line or blank area where its tone is less or equal and higher than 245 grayscale level. These prominent positions would be specified as the embedded bits; one or zero, of invisible zero watermarks if their tones are lower than 245 or equal and higher than 245 respectively.

3.5.1.6 Count all intersection positions on each page of each language.

3.5.1.7 Define each intersection position as 1 bit so that total of intersection position of each page means the capacity of hiding bit for hiding watermark data.

3.5.2 Invisible Zero Watermark Marking Scheme

To apply the cross ratio to text image watermarking, three reference points are required. The watermark marking process under this method can be proceeded by applying the embedding scheme as described in the Section 3.4.1.

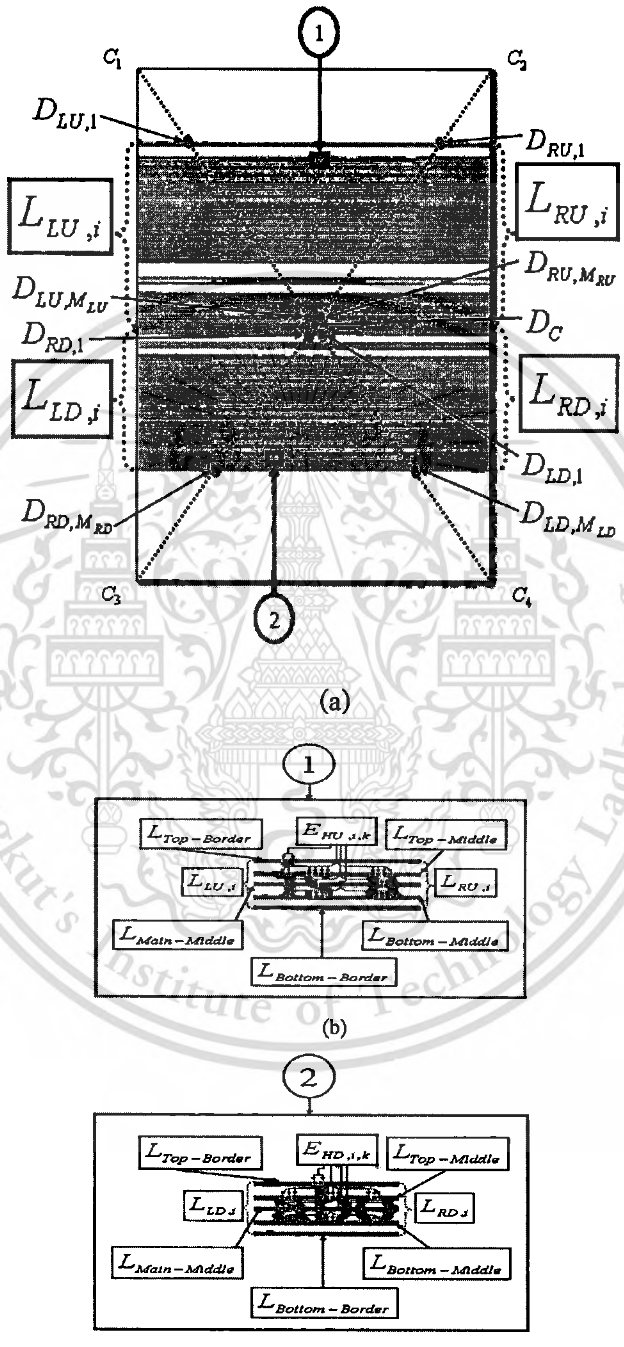


Figure 3.4 (a) Notations of horizontal lines intersect two diagonal lines. (b) and (c) Notations of virtual five lines intersect text character lines.

However, under this method the watermark virtually marking at the selected intersection points which the horizontal crossed lines run across the skeleton lines of each character, as shown in Fig. 3.4 (a), (b) and (c).

3.5.3 Invisible Zero Watermark and Integrity Detection Scheme

3.5.3.1 Identify Angle and Edge of a Watermark-embedded Text Image

The angle and edge of a watermark-embedded text image can be identified by the process as described in the Section 3.4.2.1 This process will give us the certain text image angles and edges so that identify all four reference corners, sides and its reference center point, $D_{C,}$ of a text image where the left and right diagonal lines intersected, as described under the Section 3.5.1. This main reference center would bring us to trace back to other reference lines and finally all watermark embedded lines, as explained in the Section 3.5.2 and then eventually detects the zero watermark embedded points.

3.5.3.2 Identify the language type of a watermark-embedded text image

According to the watermark embedding patterns are depending on the specific type of language so that after the main reference lines of a text image detecting, we have to identify the type of language by applying the reversed aspect ratio as described under the Section 3.3.

Based on this ratio application, the average reversed aspect ratio, as shown in Eq. (3.25), of each language would be explored in advance in order to use it as the reference identity of each specific language by following the brief procedure specified below.

1. Split each character of each language text image sample set; English, Thai, Chinese, and Arabic, by applying the histogram technique.

2. Select only the text row of each language that its height is higher than the mean height value but lesser than the highest value.

3. Capture only characters of each language that have grayscale value lesser than, mean value, 250. Then, we would get the results as shown in Figs. 3.5 ~ 3.8.

4. Detect height and width of each selected captured character of each language and then calculate its reversed aspect ratio, *Rr*, by applying the Eq. (3.1).

5. Calculate the average reversed aspect ratio, *ARr*, of each language by applying the Eq. (3.2) and find out the reference identity of each specific language; English, Thai, Chinese, and Arabic.

3.5.3.3 Zero Watermark Detection

After the four corner points are detected, watermark marking points must be identified. Each point can be calculated by using the method similar to that of the marking stage (see Eq. (3.7)). By extracting the values of the pixels corresponding to those watermark marking points, a watermark can be detected by using any of the existing watermark detectors. Here, we adopt the basic matching percentage method as described in the Section 3.5.3.4 to compare each detected watermark positions with its originally marked positions, one-by-one, till cover all specified positions. Predefine the set of cross-ratio values, to be used in subsequent steps.

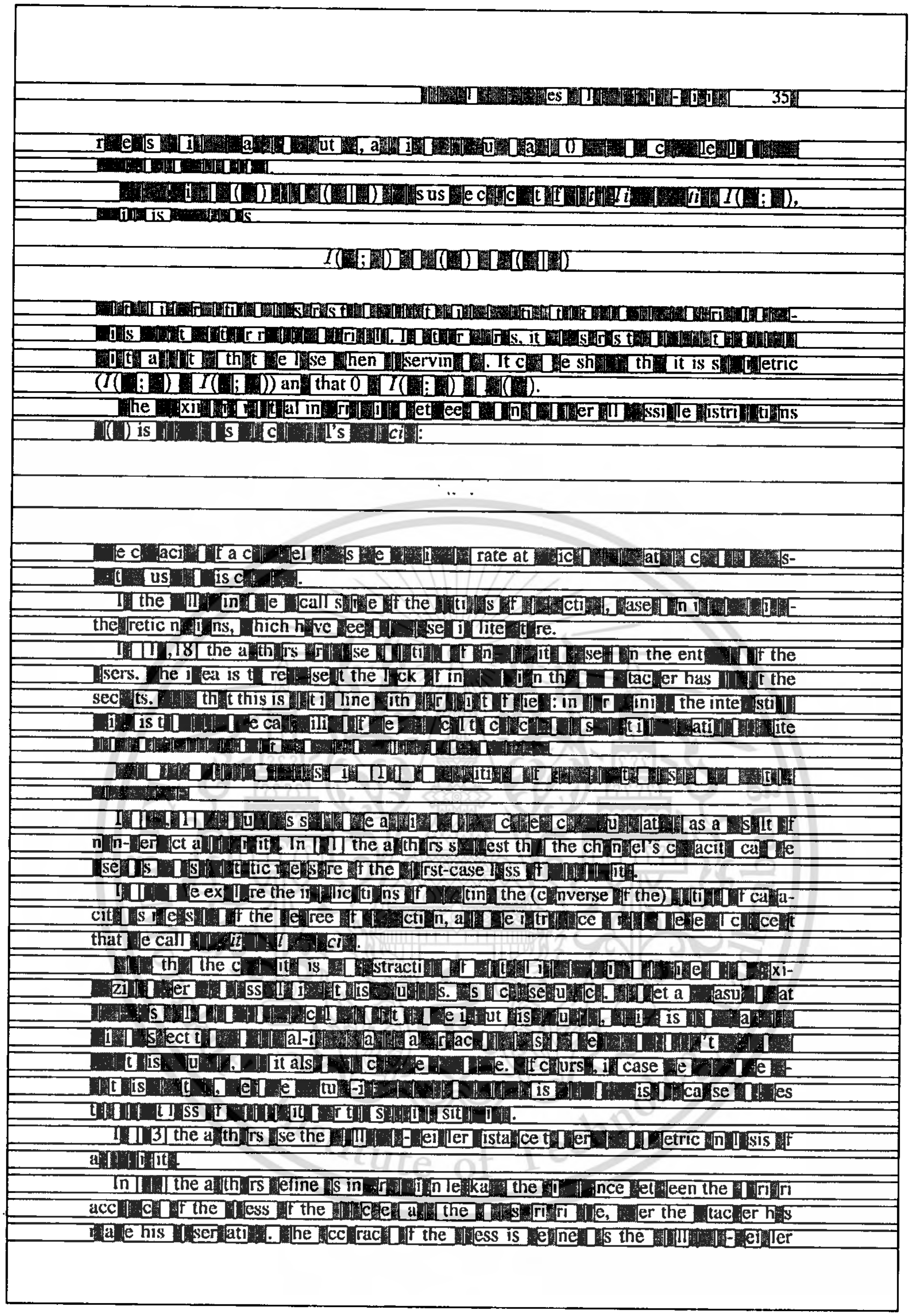


Figure 3.5 The example of the captured characters of the watermarked English text image.

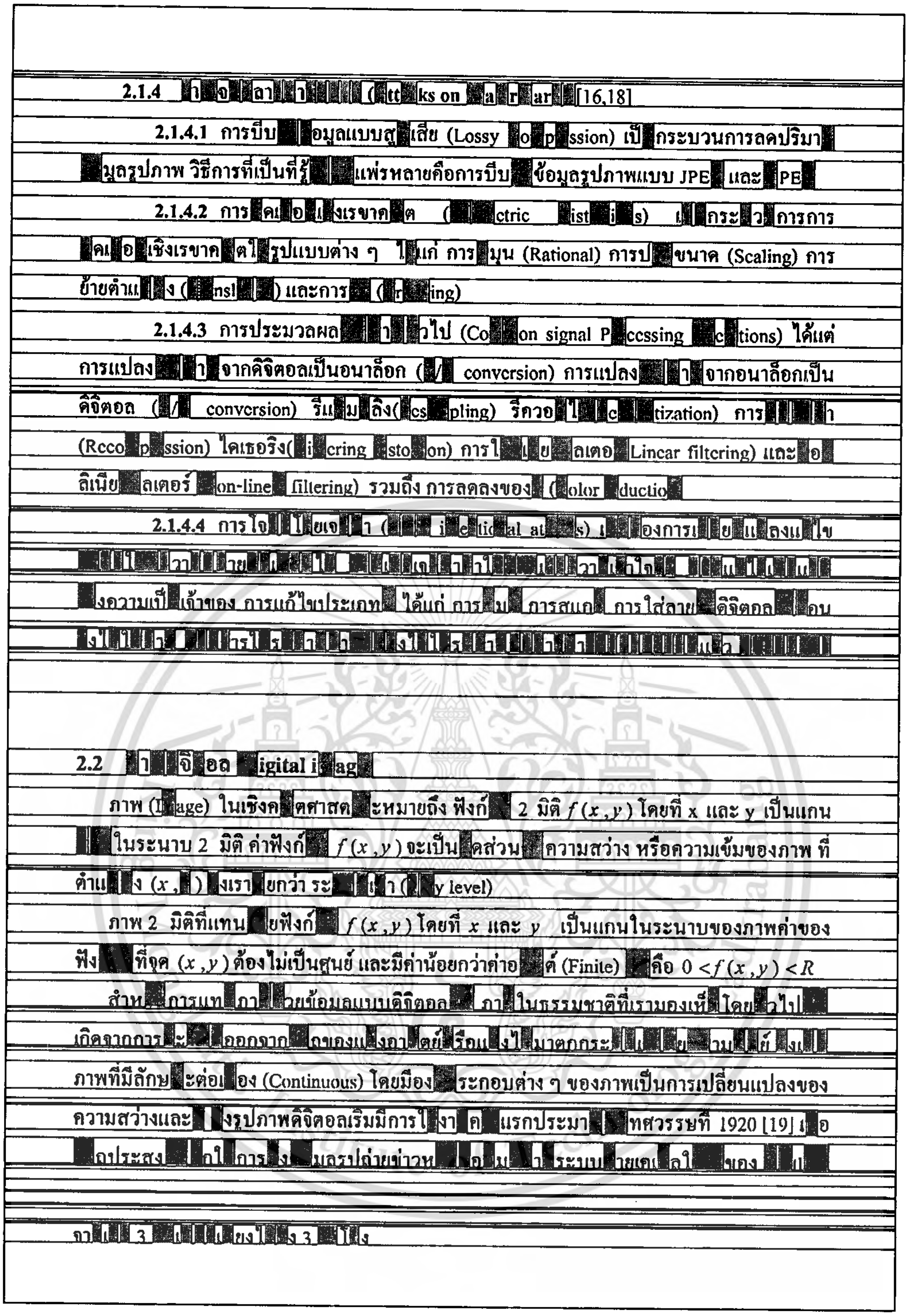


Figure 3.6 The example of the captured characters of the watermarked Thai text image.

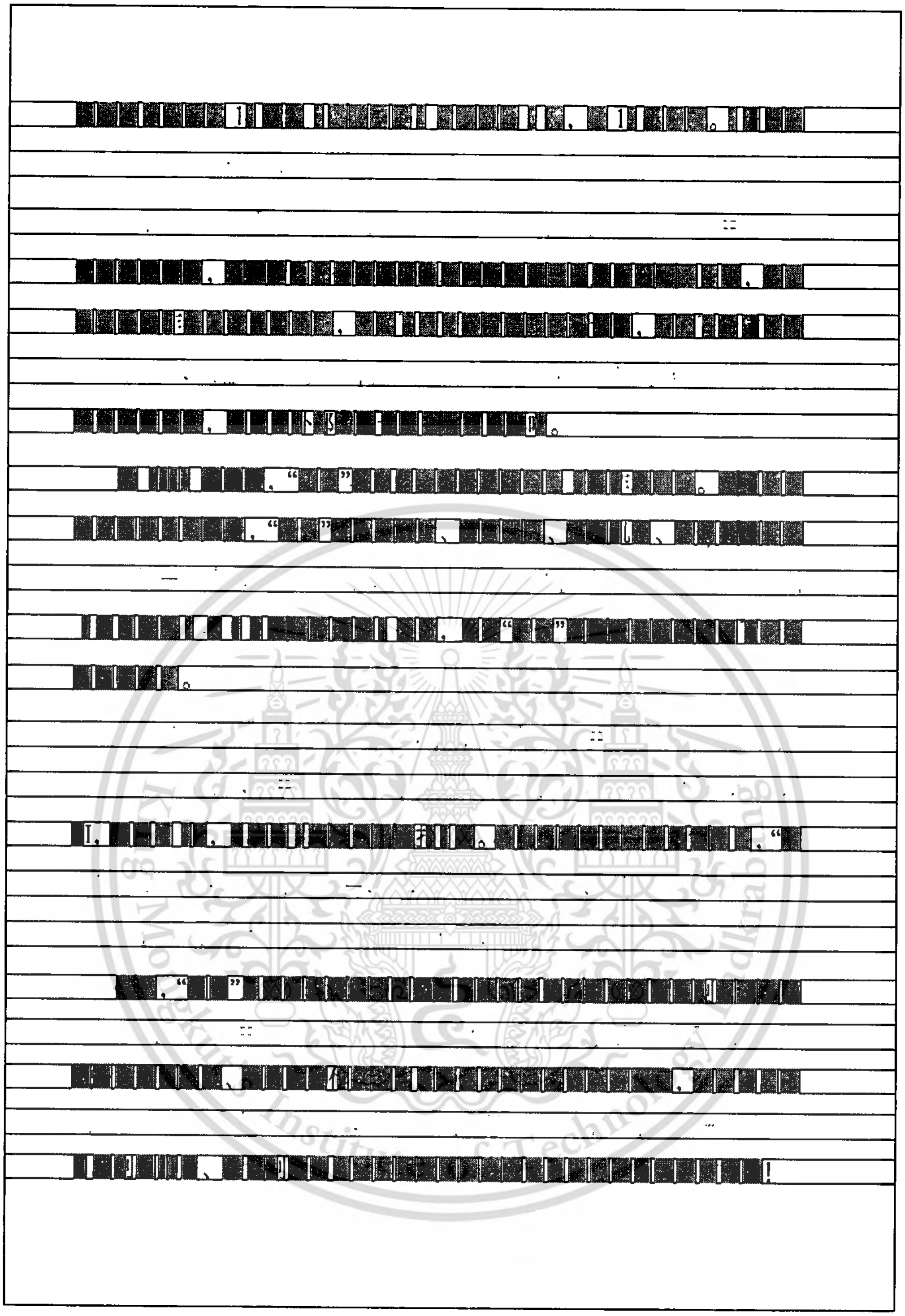


Figure 3.7 The example of the captured characters of the watermarked Chinese text image.

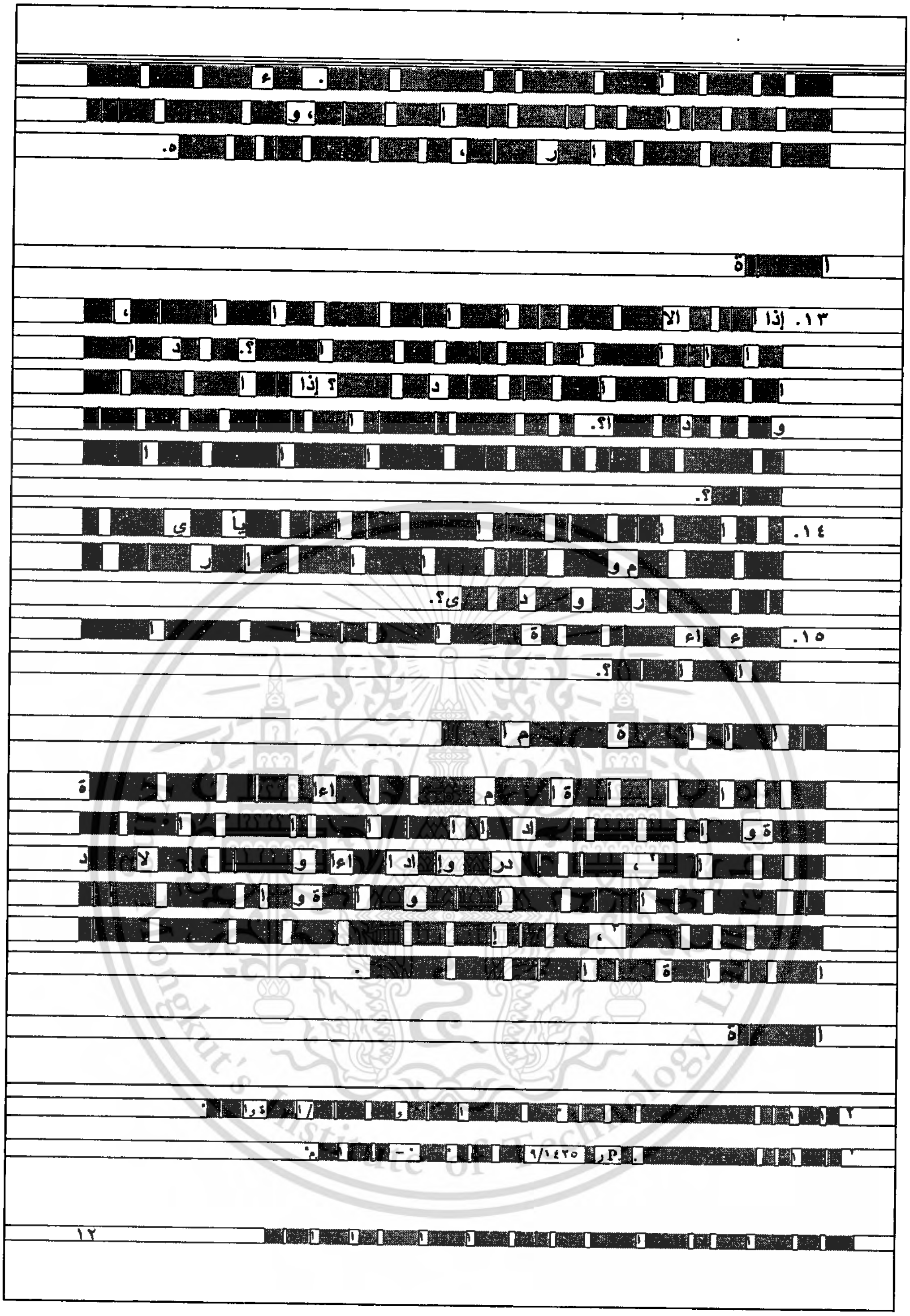


Figure 3.8 The example of the captured characters of the watermarked Arabic text image.

3.5.3.4 The Matching Percentage

The matching percentage is applied for comparing the watermark embedded position with the watermark detected position, one-to-one, and then count all the matched positions to calculate the percentage of matching by using Eq. (3.7).

$$MPW = \frac{\sum D_{wm}(x_i, y_i)}{\sum P_{ip}(x_i, y_i)} \times 100 \qquad (3.7)$$

where MPW = Matching percentage of watermark embedded position detection which corresponding to the selected intersection point for embedding watermark.

D_{wm} = Detected watermark at position x_i, y_i.

P_{ip} = Prominent Intersection points at position x_i, y_i which were selected to embed watermark.

3.6 Method 3: Applying The Single Reference-line Intersection of Additional Vertical Line and The Cross-ratio Theory to Increase More Watermark-hiding Capacity and Original-text Verification Performance of Text-image Watermark with Zero Watermark Embedded

After studying the structure of character skeleton lines of English, Thai, Chinese and Arabic languages, in the Method 2, we had found that some character skeleton lines of some languages, such as English and Thai mostly have their skeleton lines in the vertical direction which the horizontal crossed line could not run and cut along them. This limitation of the horizontal line means some text manipulations, such as text adding, reordering and deleting, would not be detected. Thus, the Method 3 has been designed for exploring the advantages of the vertical crossed line. Like two methods described above, the text watermarking process is based on two sub-processes; the first one is to match the cross-ratio reference points with zero watermark marking positions for easily tracking the zero watermark marking points after a text image has been attacked and the last one is to detect the zero watermarks and verify its integrity with the matching percentage method.

3.6.1 Invisible Zero Watermark Marking Scheme

To apply the cross ratio to text image watermarking, three reference points are required. The watermark marking process under this method can be predefined the set of cross-ratio values and find the image center by following the steps and equations as described in the Section 3.4.1. After that, follow the steps as described below.

3.6.1.1 Find each of the primary-level watermark marking points (TCr_1 and TCr_2) on the right diagonal line (see Fig. 3.9 (a)) as described by Eqs. (3.5) ~ (3.6) below. Those points can be identified by using two corner points of the left diagonal line (C_2 and C_3), in combination with the image center point D_c and the predefined cross-ratio values (C_r). Here, (x, y) = ($x_{TCr,i}$, $y_{TCr,i}$), $i = 1, 2$, $A = C_2$, $C_2 = (x_s, y_s)$, $B = TCr_i$, $C = D_c$ and $D = C_3 = (x_d, y_d)$.

3.6.1.2 Find each of the primary-level watermark marking points (TCr_3) on the right diagonal line (see Fig. 3.9(a)) as described by Eqs. (3.5) ~ (3.6). Those points can be identified by using two corner points of the left diagonal line (C_1 and C_4), in combination with the image center point D_c and the predefined cross-ratio values (C_r).

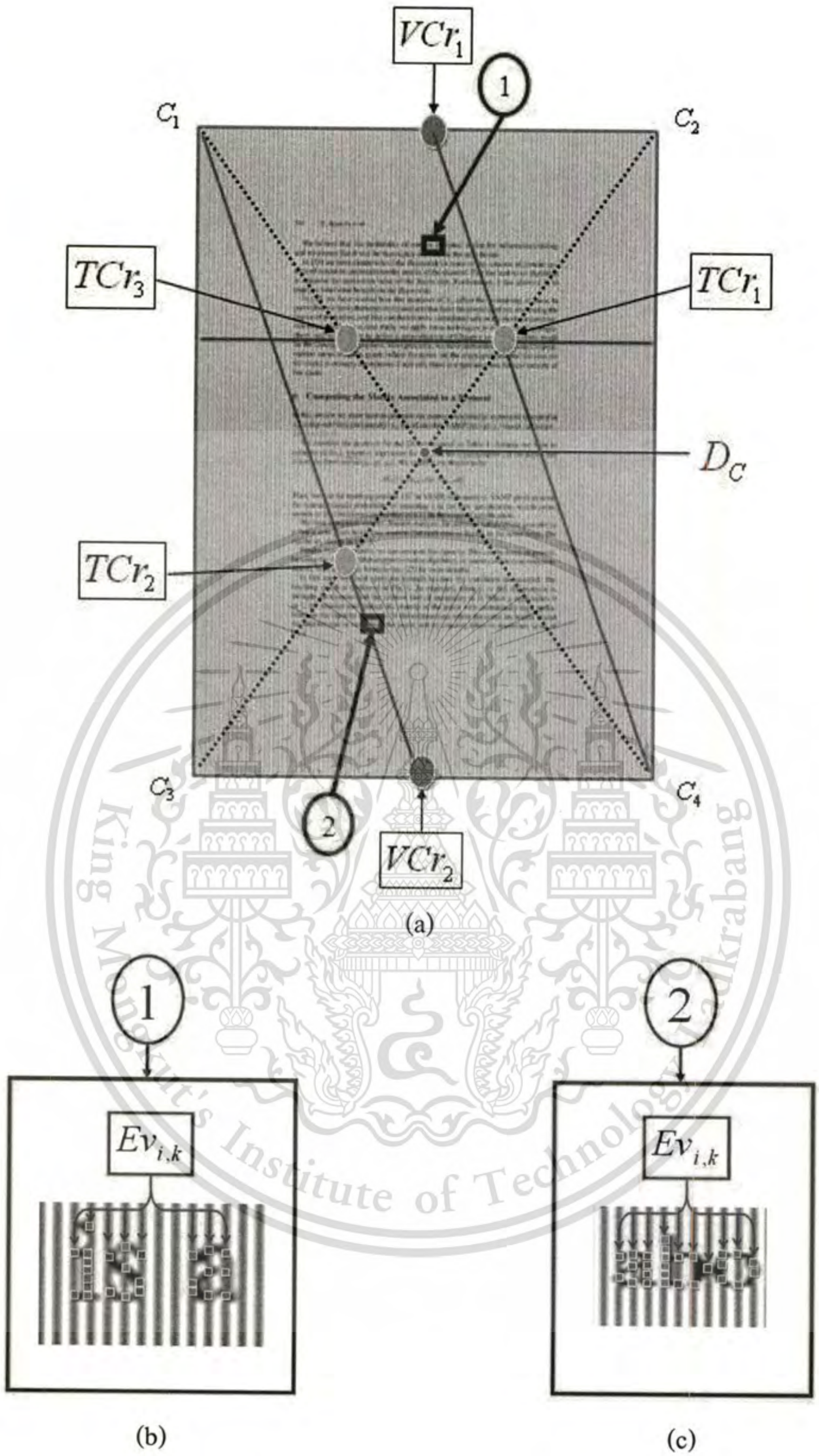


Figure 3.9 (a) Notations of vertical lines intersect text character lines. (b) and (c) Notations of virtual lines intersect text character lines.

Here, $(x,y) = (x_{TCr,3}, y_{TCr,3})$, $A = C_2$, $C_2 = (x_s, y_s)$, $B = TCr_3$, $C = D_c$ and $D = C_3$, $C_3 = (x_d, y_d)$.

3.6.1.3 For each pair of (x_1, y_1) (x_2, y_2) level find an intersection, (x_i, y_i), of crossed line of each level drawn across (X_3, y_3) (x_4, y_4); by applying Eqs. (3.3) ~ (3.4). Here, for each pair of C_4, TCr_1 find an intersection, $x_{VCr,1}, y_{VCr,1}$, of crossed line of each level drawn across C_1C_2, $(x_1, y_1) = (x_{C,1}, y_{C,1})$, $(x_2, y_2) = (x_{C,2}, y_{C,2})$, $(x_3, y_3) = (x_{C,4}, y_{C,4})$, $(x_4, y_4) = (x_{TCr,1}, y_{TCr,1})$, for each pair of C_1, TCr_2 find an intersection, $x_{VCr,2}, y_{VCr,2}$, of crossed line of each level drawn across C_3C_4, $(x_1, y_1) = (x_{C,1}, y_{C,1})$, $(x_2, y_2) = (x_{TCr,2}, y_{TCr,2})$, $(x_3, y_3) = (x_{C,3}, y_{C,3})$, $(x_4, y_4) = (x_4, y_4)$

3.6.1.4 Find each of the secondary-level watermark marking points ($V_{Cr1,i}$) on the line (see Fig. 3.9 (b)) as described by Eqs. (3.5) ~ (3.6) below. Those points can be identified by using two corner points of the line $C_1 C_2$, in combination with the image center point VCr_1 and the predefined cross-ratio values (C_r). Here, $(x, y) = (x_{VCr1,j}, y_{VC1,j})$, $A = C_1$, $C_1 = (x_s, y_s)$, $B = V_{Cr1,i}$, $C = VCr_1$ and $D = C_2$, $C_2 = (x_d, y_d)$.

3.6.1.5 Find each of the secondary-level watermark marking points ($V_{Cr2,i}$) on the line (see Fig. 3.9(a)) as described by Eqs. (3.5) ~ (3.6) below. Those points can be identified by using two corner points of the line $C_3 C_4$, in combination with the image center point VCr_2 and the predefined cross-ratio values (C_r). Here, $A = C_1$, $C_1 = (x_s, y_s)$, $B = V_{Cr1,i_1}$, $C = VCr_1$ and $D = C_2$, $C_2 = (x_d, y_d)$.

3.6.1.6 For each pair of $(x_{VCr1,i}, y_{VCr1,i})$ $(x_{VCr2,i}, y_{VCr2,i})$ level find an intersection, (x_i, y_i), of crossed line of each level drawn across $(X_{TCr,3}, y_{TCr,3})$ $(x_{TCr,1}, y_{TCr,1})$; by applying Eqs. (3.3) ~ (3.4). For each pair of VCr_1, VCr_2 find an intersection, $x_{SCr,i}, y_{SCr,i}$, of crossed line of each level drawn across TCr_3TCr_1, $(x_1, y_1) = (x_{VCr1,i}, y_{VCr1,i})$, $(x_2, y_2) = (x_{VCr2,i}, y_{VCr2,i})$, $(x_3, y_3) = (x_{TCr,3}, y_{TCr,3})$, $(x_4, y_4) = (x_{TCr,1}, y_{TCr,1})$,

3.6.1.7 Detect each prominent intersection position of each vertical line which runs across each text-character skeleton line or blank area where its tone is less or equal and higher than 245 grayscale level. These detected prominent positions would be specified as the embedded bits; one or zero, of invisible zero watermarks ($Ev_{i,k}$) if their tones are lower than 245 or equal and higher than 245 respectively, as shown in Fig. 3.9(a) and (b).

3.6.2 Invisible Zero Watermark and Integrity Detection Scheme

3.6.2.1 Identify Angle and Edge of a Watermark-embedded Text Image

To detect a watermark from the text image, the four image corner points must first be detected. These angle and edge of a watermark-embedded text image can be identified by following the process as described in the Section 3.4.2.1

3.6.2.2 Identify The Language Type of a Watermark-embedded Text Image

The language type of a watermark-embedded text image can be identified by following the procedure as described in the Section 3.5.3.2.

3.6.2.3 Zero Watermark Detection

Like the Method 2, the matching percentage method, as described in the Section 3.5.3.3, will be used to detect the marked zero watermark.

3.7 Method 4: Applying The Double Reference-line Intersection of Vertical and Horizontal Lines and The Double Cross-Ratio Method to Increase Maximum Watermark-Hiding Capacity and Original-text Verification Precision of Text-image Watermark with Zero Watermark Embedded.

According to the Method 2 and Method 3, even each horizontal and vertical line would be fixed with the specific interval, but the intersected points run across the character skeleton lines and space could not be fixed, depending on text written on each page, so that we got the random intersection point pattern which is difficult to embed and detect watermarks. Moreover, it could not be sure that these crossed-line intersection points have covered throughout a text image page. Hence, we thought that the double or grid lines of horizontal and vertical lines would help us to solve this problem. Under this experiment, the intersection points of horizontal and vertical lines which exactly lied on the character skeleton lines or space would be counted as the marking point of the zero watermarks.

3.7.1 Invisible Zero Watermark Marking Scheme

The watermark marking process under this method can be predefined the set of cross-ratio values and find the image center by following the steps and equations as described in the Section 3.4.1. After that, follow the steps as described below.

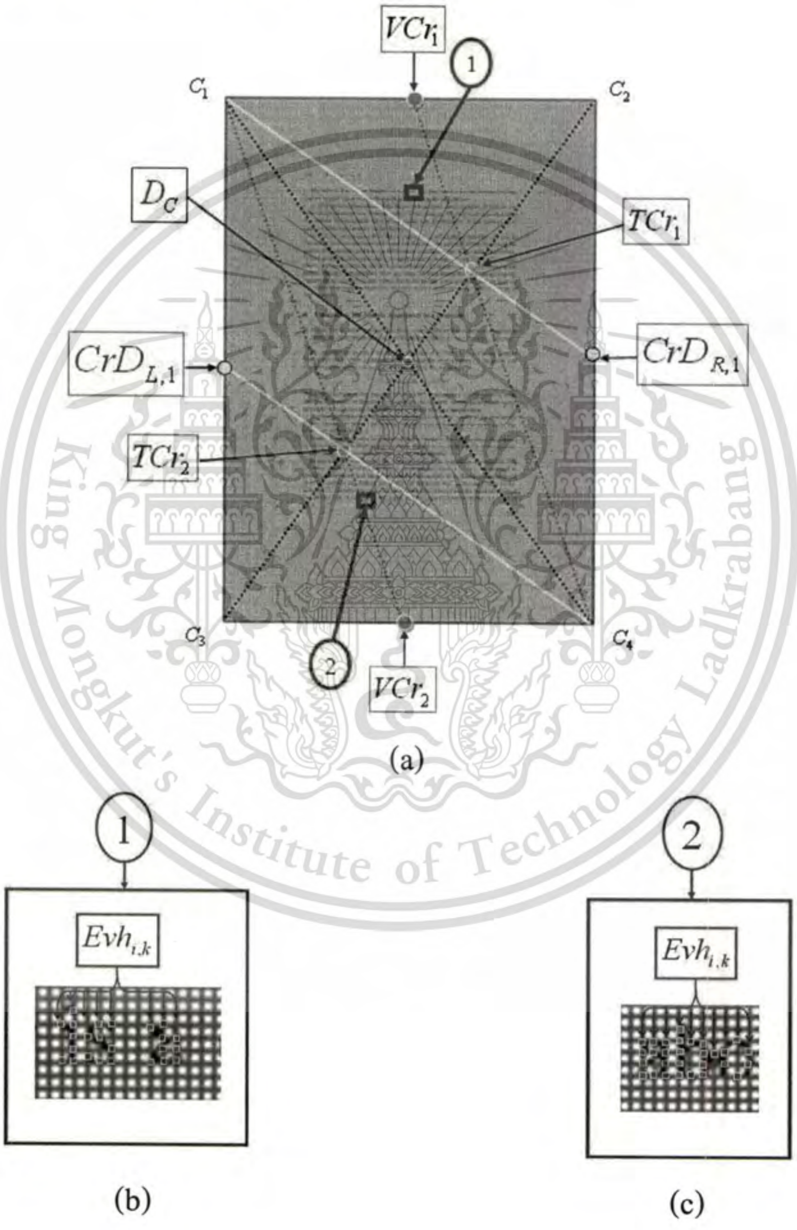


Figure 3.10 (a) Notations of double reference-line intersection of vertical and horizontal lines. (b) and (c) Notations of the double lines intersect text character lines.

3.7.1.1 Find each of the primary-level watermark marking points (TCr_1 and TCr_2) on the right diagonal line (see Fig. 3.10) as described in the Section 3.6.1.3.

3.7.1.2 For each pair of (x_1, y_1) (x_2, y_2) level find an intersection, (x_i, y_i) ,of crossed line of each level drawn across (X_3, y_3) (x_4, y_4); by applying Eqs. (3.3) ~ (3.4). Here, for each pair of TCr_2, C_4 find an intersection, $x_{CrDl,1}, y_{CrDl,1}$, of crossed line of each level drawn across C_1C_3 , $(x_1, y_1) = (x_{TCr,2}, y_{TCr,2})$, $(x_2, y_2) = (x_{C,4}, y_{C,4})$, $(x_3, y_3) = (x_{C,3}, y_{C,3})$, $(x_4, y_4) = (x_{C,1}, y_{C,1})$, for each pair of C_1 , TCr_1 find an intersection , $x_{CrDr,1}, y_{CrDr,1}$,of crossed line of each level drawn across C_4C_2 , $(x_1, y_1) = (x_{C,1}, y_{C,1})$, $(x_2, y_2) = (x_{TCr,1}, y_{TCr,1})$, $(x_3, y_3) = (x_{C,4}, y_{C,4})$, $(x_4, y_4) = (x_{C,2}, y_{C,2})$

3.7.1.3 Find each of the secondary-level watermark marking points ($H_{CrL,i}$) on the line as described by Eqs. (3.5) ~ (3.6) below. Those points can be identified by using two corner points of the line C_1 C_3, in combination with the image center point $H_{CDL,1}$ and the predefined cross-ratio values (C_r). Here, $A = C_l$, $C_l = (x_s, y_s)$, $B = V_{HCrL,i}$, $C = H_{CDL,1}$ and $D = C_3$, $C_3 = (x_d, y_d)$.

3.7.1.4 Find each of the secondary-level watermark marking points ($H_{CrR,i}$) on the line as described by Eqs. (3.5) ~ (3.6) below. Those points can be identified by using two corner points of the line C_2 C_4, in combination with the image center point $H_{CDR,1}$ and the predefined cross-ratio values (C_r). Here, $A = C_2$, $C_2 = (x_s, y_s)$, $B = H_{HCrR,i}$, $C = H_{CDR,1}$ and $D = C_4$, $C_4 = (x_d, y_d)$.

3.7.1.5 Repeat step 3.7.1.5 to 3.7.1.7 for finding each of the secondary-level watermark marking points $(x_{VCr1,i}, y_{VCr1,i})$ and $(x_{VCr2,i}, y_{VCr2,i})$.

3.7.1.6 Each pair of $(x_{VCr1,i}, y_{VCr1,i})$ $(x_{VCr2,i}, y_{VCr2,i})$ level find an intersection, (x_i, y_i) ,of crossed line of each level drawn across $(X_{HCrL,i}, y_{HCrL,i})$ $(X_{HCrR,i}, y_{HCrR,i})$; by applying Eqs. (3.3) ~ (3.4). For each pair of $(x_{VCr1,i}, y_{VCr1,i})$ $(x_{VCr2,i}, y_{VCr2,i})$ find an intersection, $(X_{Evh,i,k}, y_{Evh,i,k})$, of crossed line of each level drawn across $(X_{HCrL,i}, y_{HCrL,i})$ $(X_{HCrR,i}, y_{HCrR,i})$, $(x_1, y_1) = (x_{VCr1,i}, y_{VCr1,i})$, $(x_2, y_2) = (x_{VCr2,i}, y_{VCr2,i})$, $(x_3, y_3) = (x_{HCrL,i}, y_{HCrL,i})$, $(x_4, y_4) = (x_{HCrR,i}, y_{HCrR,i})$.

3.7.1.9 Detect each prominent intersection position of each double (H & V) line which runs across each text-character skeleton line or blank area where its tone is less or equal and higher than 245 grayscale level. These detected prominent positions would be specified as the embedded bits; one or zero, of invisible zero watermarks ($Evh_{i,k}$) if their tones are lower than 245 or equal and higher than 245 respectively, as shown in Fig. 3.10 (a) and (b).

3.7.2 Invisible Zero Watermark and Integrity Detection Scheme

3.7.2.1 Identify Angle and Edge of a Watermark-embedded Text Image

To detect a watermark from the text image, the four image corner points must first be detected. These angle and edge of a watermark-embedded text image can be identified by following the process as described in the Section 3.4.2.1

3.7.2.2 Identify The Language Type of a Watermark-embedded Text Image

The language type of a watermark-embedded text image can be identified by following the procedure as described in the Section 3.5.3.2.

3.7.2.2 Zero Watermark Detection

Like the Method 2, the matching percentage method, as described in the Section 3.5.3.3, will be used to detect the marked zero watermark.

3.8 Method 5: Applying The Single Horizontal, Vertical and Double Reference-line Intersection and The Double Cross-ratio Method to Create The Multi-Layer Watermarking for Increasing Maximum Watermark-hiding Capacity and Original-text Verification Precision of Text-image Watermark with Zero Watermark Embedded.

To find the new effective technique for increasing remarkable hiding-bit capacity to multi-language text image watermark is the main target of this research. This breakthrough is the most challenge for all text-image watermarking researches. Thus, we would like to achieve this target by combining three types of crossed-line zero watermarking techniques of Method 2, Method 3 and Method 4 together and created the 11 layers of crossed-line intersections, with the cross ratio theory applying for effectively controlling the watermark embedding and detecting processes, were tested with multi-language text images; Thai, English, Chinese and Arabic languages. This experiment revealed that 11 layers of 11 patterns of crossed-line intersections on each page of multi-language text images, by which horizontal(H), vertical(V) and double (H & V) lines were drawn and used to create multiple layers of intersection points for embedding watermark secret data bits, could effectively built up a huge watermark hiding-bit capacity. This means the more unique intersection points and layers you have created, the more hiding-bit capacity you would get, so that a lot of watermark secret data could be embedded. Moreover, even though these enormous bits were embedded, nobody could observed them because it used only their virtual intersection points, with defined with the exact bit values, as the reference positions for embedding the corresponding watermark data bit, designated one bit for one point with the same bit value, without any physically marking or modifying the original text images, so called zero watermarking. Nevertheless, if all points have been embedded, it would very helpful to owners to verify their integrities and prevent them to be forced to accept the fake text images if their physical watermarks have been copied and reproduced.

3.8.1 Multi-layer Watermark Embedding Scheme

3.8.1.1 Create The Watermark-hiding Bits Capacity with The Multiple Layers of Crossed-line Intersection

The big volume of watermark-hiding bit capacity of the text image can be effectively created by using the multiple layers of crossed-line intersection of horizontal(H), vertical(V) and double (H & V) lines which run across one another under the specific pattern, as shown in Fig. 3.11 Each layer can be created by following the procedure belows.

1st, 2nd and 3rd Crossed-line intersection layers of the main and shifted horizontal lines

1. Set the first layer of main horizontal crossed-line intersection by first identifying and drawing the horizontal lines for the top borders of the first text line of each page of text image.

2. Draw the second main horizontal line, next to the first main horizontal line, the third and the rest till the last text line (see Fig. 3.12 (a)), with 3-pixel interval, the suitable gap for geometric variation buffer which has been found from our previous experiment.

3. Set the second layer of lower-shifted horizontal crossed-line intersection with 2-pixel lower shifting (see Fig. 3.12 (a-1)) and then draw the other lower shifted horizontal lines till cover all a text image page, like drawing the main horizontal line.

4. Set the third layer of upper-shifted horizontal crossed-line intersection with 2-pixel upper shifting (see Fig. 3.12 (a-2)) and then draw the other upper shifted horizontal lines till cover all a text image page, like drawing the main horizontal line.

5. Select each prominent crossed-line intersection position of each main, lower-shifted or upper-shifted horizontal line which runs across each text-character skeleton line and blank area, as shown some samples in Fig. 3.13 (a) and (a-1). If its tone less than 245 grayscale level or in gray/black tone, it would be specified the bit value as 1. If its grayscale value is equal and higher than 245 grayscale level or in the white tone, it would be specified the bit value as 0, as shown in Fig. 3.13 (a) and (a-1) and Table 3.2. These selected prominent white and black positions would be specified as the marking points of zero digital watermark bits for hiding secret data and these points represent the main hiding-bit capacity of the first, second and third crossed-line layers.

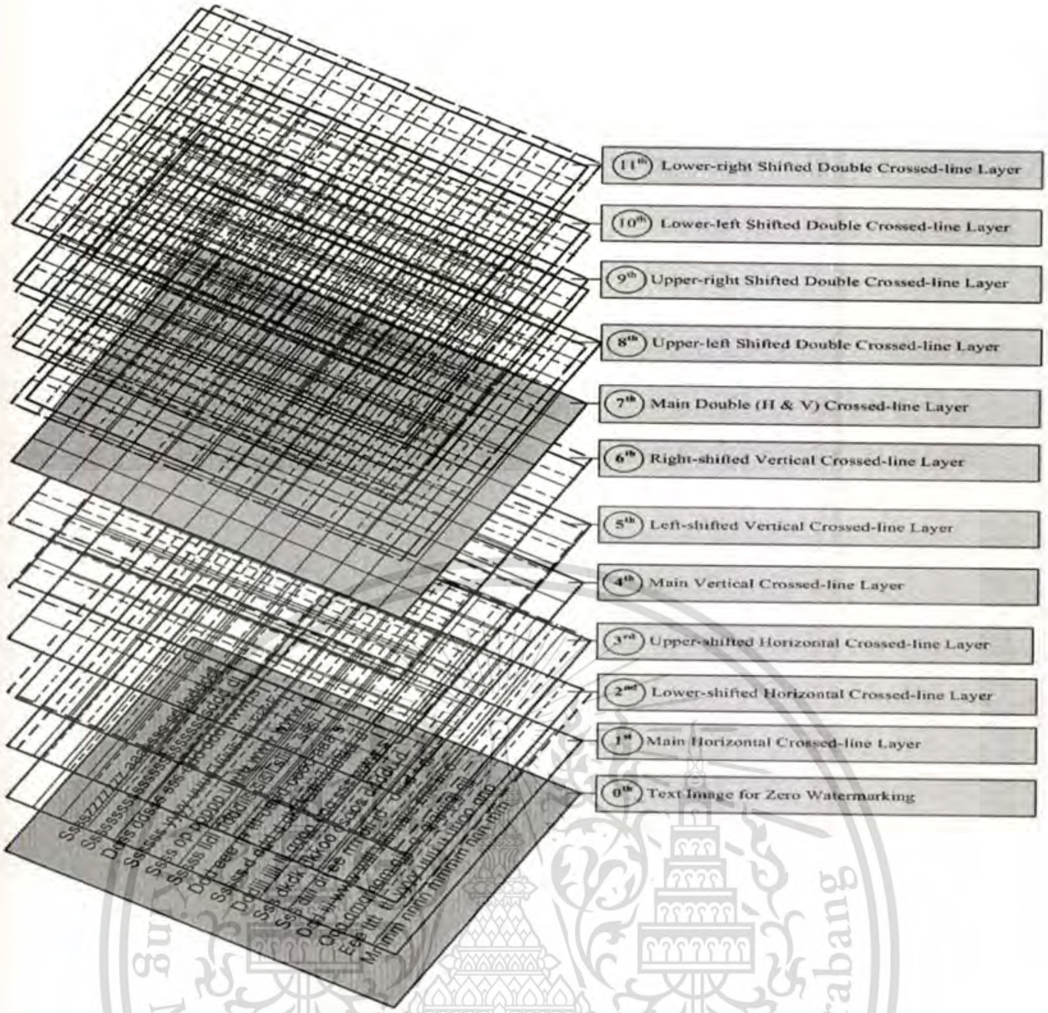


Figure 3.11 Eleven layers of crossed-line intersection of horizontal (H), vertical (V) and double (H & V) lines.

6. If we need to get more hiding-bit capacity on each layer, these horizontal crossed lines can be shifted to the new positions and drawn the new shifted horizontal lines for creating the new intersection points which run across these text-character skeleton lines and blank area. These shifting lines would create more intersection points on the new positions which give us more hiding-bit capacities. Moreover, these new intersection points, surrounding its main original intersection points, can be used as the crosscheck points for identifying the exact point of its originality and verifying the changing of its integrity as well.

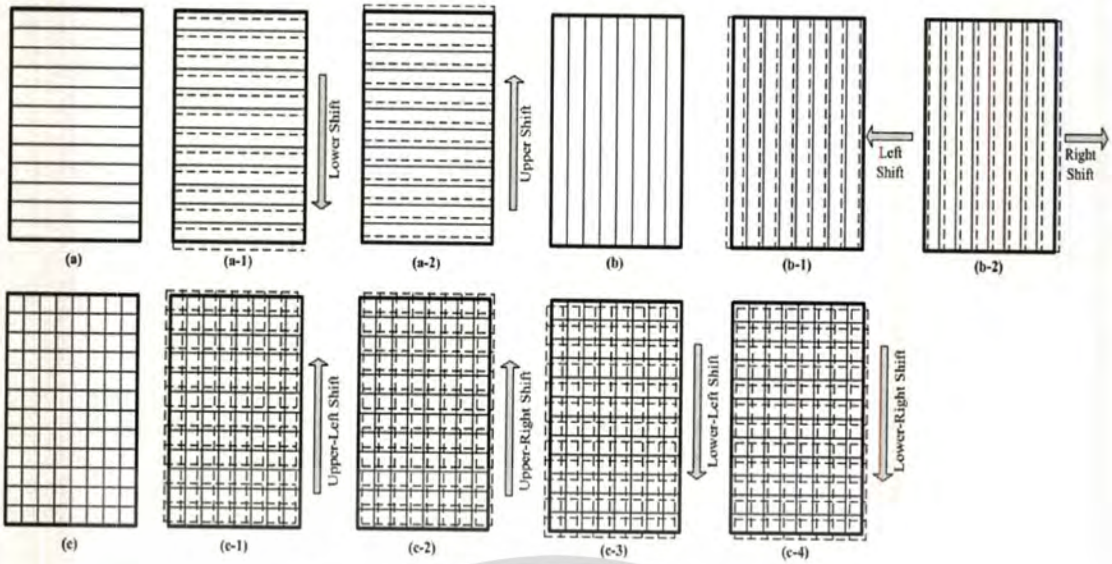


Figure 3.12 The structure of three main layers and their variations which use the main and shifted (dash-line) (a) horizontal (H), (b) vertical (V) and (c) double (H & V) lines run across text-character skeleton lines and blank area.

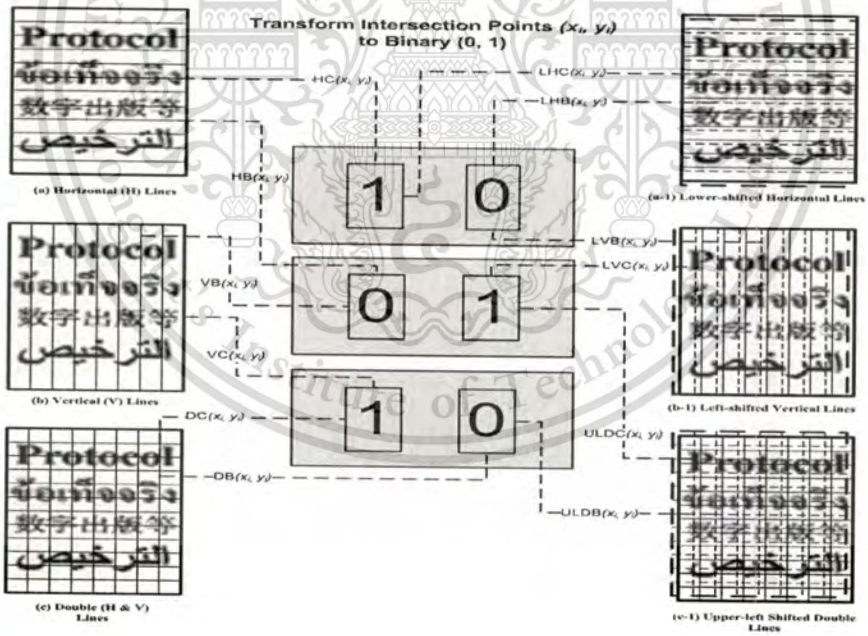


Figure 3.13 The technique to transform intersection points (x_i, y_i), run across text-character skeleton lines and blank area with (a) the horizontal (H), (b) vertical (V), (c) double H & V and their shifted lines; (a-1), (b-1) and (c-1), into binary or bit; 0, 1, data.

Table 3.2: The mapping table of watermark data bit values and intersection points

Layer No.	**Type of Crossed Line**	**Variable Name and Intersection Points (Position)**		**Watermark Data Bit Value**
		Across Character (C) Skeleton Line	Across Blank (B) Area	
1st	Main Horizontal (H)	HC(x_i, y_i)		1
			HB(x_i, y_i)	0
2nd	Lower-shifted H.	LHC(x_i, y_i)		1
			LHB(x_i, y_i)	0
3rd	Upper-shifted H.	UHC(x_i, y_i)		1
			UHB(x_i, y_i)	0
4th	Main Vertical (V)	VC(x_i, y_i)		1
			VB(x_i, y_i)	0
5th	Left-shifted V.	LVC(x_i, y_i)		1
			LVB(x_i, y_i)	0
6th	Right-shifted V.	RVC(x_i, y_i)		1
			RVB(x_i, y_i)	0
7th	Main Double (H & V)	DC(x_i, y_i)		1
			DB(x_i, y_i)	0

Table 3.2: The mapping table of watermark data bit values and intersection points *(cont.)*

Layer No.	Type of Crossed Line	Variable Name and Intersection Points (Position)		Watermark Data Bit Value
		Across Character (C) Skeleton Line	Across Blank (B) Area	
8th	Upper-left Shifted Double (H & V)	ULDC(x_i, y_i)		1
			ULDB(x_i, y_i)	0
9th	Upper-right Shifted Double (H & V)	URDC(x_i, y_i)		1
			URDB(x_i, y_i)	0
10th	Lower-left Shifted Double (H & V)	LLDC(x_i, y_i)		1
			LLDB(x_i, y_i)	0
11th	Lower-right Shifted Double (H & V)	LRDC(x_i, y_i)		1
			LRDB(x_i, y_i)	0

4th, 5th and 6th Crossed-line intersection layers of the main and shifted vertical lines

1. Set the fourth layer of main vertical crossed-line intersection by identifying the right and left borders of text body on a text image and draw the first main vertical lines along the left border, after that draw the second main vertical line, the third and the rest till the end of the right border (see Fig. 3.12 (b)) with 3-pixel interval the suitable gap for geometric variation buffer which has been found from our previous experiment.

2. Set the fifth layer of left-shifted vertical crossed-line intersection with 2-pixel left shifting (see Fig. 3.12 (b-1)) and then draw the other left-shifted horizontal lines till cover all a text image page, like drawing the main vertical line.

3. Set the sixth layer of right-shifted vertical crossed-line intersection with 2-pixel right shifting (see Fig. 3.12 (b-2)) and then draw the other right-shifted vertical lines till cover all a text image page, like drawing the main vertical line.

4. Select each prominent crossed-line intersection position of each main, left-shifted or right-shifted vertical line which runs across each text-character skeleton line and blank area, as shown some samples in Fig. 3.13 (b) and (b-1). If its tone less than 245 grayscale level or in gray/black tone, it would be specified the bit value as 1. If its grayscale value is equal and higher than 245 grayscale level or in the white tone, it would be specified the bit value as 0, as shown in Fig. 3.13 (b) and (b-1) and Table 3.2. These selected prominent white and black positions would be specified as the marking points of zero digital watermark bits for hiding secret data and these points represent the main hiding-bit capacity of the fourth, fifth and sixth crossed-line layers.

5. If we need to get more hiding-bit capacity on each layer, these vertical crossed lines can be shifted to the new positions and drawn the new shifted vertical lines for creating the new intersection points which run across these text-character skeleton lines and blank area. These shifting lines would create more intersection points on the new positions which give us more hiding-bit capacities. Moreover, these new intersection points, surrounding its main original intersection points, can be used as the crosscheck points for identifying the exact point of its originality and verifying the changing of its integrity as well.

7th, 8th, 9th, 10th and 11th Crossed-line intersection layer of the main and shifted double (horizontal and vertical) lines

1. Set the seventh layer by drawing both main horizontal(H) and vertical(V) lines (see Fig. 3.12 (c)), so called the main double (H & V) lines, with 3-pixel interval, on the same text-image page in order to create the seventh layer of main double-line intersection where both main horizontal line and main vertical line run across either text-character skeleton line or blank area and create their intersection points for embedding watermark data bits.

2. Set the eigth, nineth, tenth and eleventh layers by drawing 2-pixel shifted double (H & V) lines to upper-left, upper-rigth, lower-left and lower-right direction (see Fig. 3.12 (c-1) – (c-4)) in order to create the eigth, nineth, tenth and eleventh layers, respectively, of shifted double-line intersection where both main horizontal line and main vertical line run across either text-character skeleton line or blank area and create their intersection points for embedding watermark data bits.

3. Select each prominent crossed-line intersection position, as shown some samples in Fig. 3.13 (c) and (c-1). If its tone less than 245 grayscale level or in gray/black tone, it would be specified the bit value as 1. If its grayscale value is equal and higher than 245 grayscale level or in the white tone, it would be specified the bit value as 0, as shown in Fig. 3.13 (c) and Table 3.2. These selected prominent white and black positions would be specified as the marking points of zero digital watermark bits for hiding secret data and these points represent the hiding-bit capacity.

4. If we need to get more hiding-bit capacity on these layers, these double horizontal and vertical crossed lines which intersect with either text-character skeleton lines or blank area can be shifted to the new positions and drawn the new shifted double (H & V) lines for creating the new intersection points which run across these text-character skeleton lines and blank area. These shifting lines would create more intersection points on the new positions which give us more hiding-bit capacities. Moreover, these new intersection points, surrounding its main original intersection points, can be used as the crosscheck points for identifying the exact point of its originality and verifying the changing of its integrity as well.

These 11 crossed-line intersection layers, described above, are designed to generate remarkably enormous watermark hiding-bit capacity to each text-image page. Under this experiment, we have tested with 35 pages of Thai, English, Chinese and Arabic language text images by totally counting all crossed-line intersection points of all layers which one intersection point means one bit of watermark hiding data.

3.8.1.2 Define The Redundant Watermark Data Block

Before embedding watermark secret data, it needs to define the redundant data block for repeatedly buffering its watermark secret data on each text-image page, as shown in Fig. 3.14. This means the more redundant data we have, the more accuracy detected secret data we get. There are two options to make the redundant data block, depending on the volume of watermark secret data, as follow.

Option 1: Set multiple redundant watermark data blocks on each embedded layer

This option is suitable for a small volume of embedded secret data or in case of limitted availibility of the hiding-bit capacity. Thus, the number of redundant watermark-data block, on each text-image page, can be calculated by the following equation.

$$nWDB = \frac{Ahb}{(Sdb + Gbb)} \qquad (3.8)$$

where *nWDB* stands for number of redundant watermark-data block which can be specified their *ID* afterward, *Ahb* stands for available hiding bits; selected prominent intersection points, of each crossed-line intersection layer, *Gbb* stands for bits spared for gap between each redundant block and *Sdb* stands for secret data bits which we would like to embed.

Option 2: Set one embedded layer as one redundant watermark data block

This option is suitable for a big volume of embedded secret data or if a lot of hiding-bit capacity is needed. Thus, we have to dedicate all selected prominent intersection points on one layer for the watermark secret data embedding. However, for this case, the other layers of this text-image page would be used as the redundant data blocks for the dedicated layer.

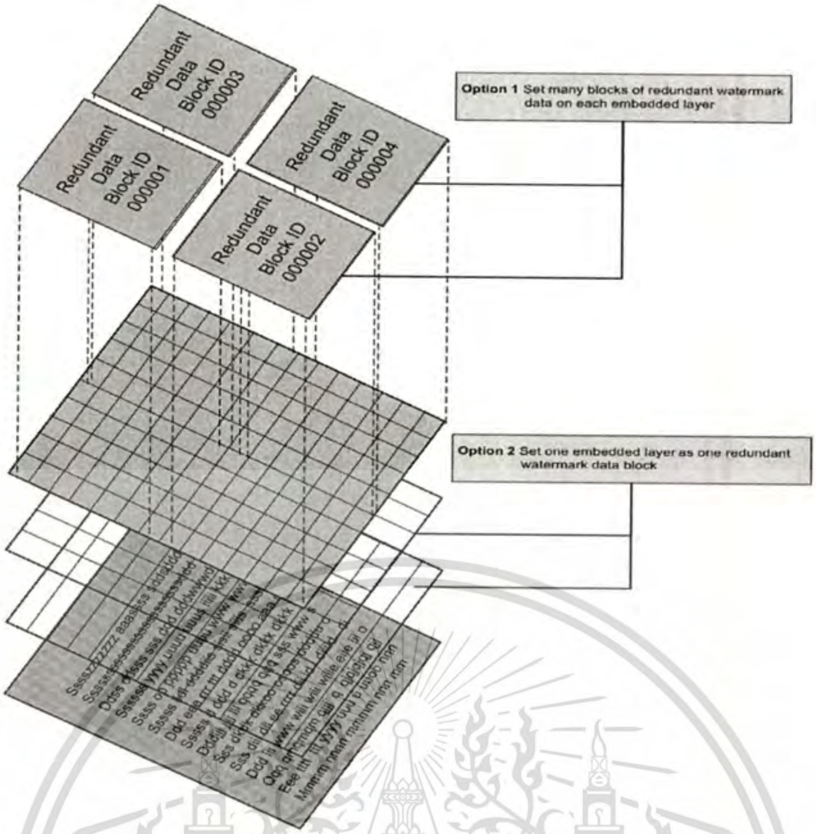


Figure 3.14 The two options of redundant watermark data block creation.

3.8.1.3 Transform The Watermarked Secret Data into Binary and Embedding Positions

After the redundant watermark data blocks have been completely specified, the next steps to do are described below.

1. Define the secret data to be embedded as the watermark on each redundant block or layer.

2. Transform the watermarked secret data, for example the COPYRIGHT word, into binary or bit, 0, 1, as shown in Figs. 3.15 and 3.16.

3. Select the prominent intersection points or their locations, (x_i, y_i), which corresponding to specific bit 1or bit 0 data of the transformed secret data, as specified in Table 3.2, for assigning as the watermark embedding positions.

3.8.1.4 Embed Watermarks under Cross-ratio Directing

Under this experiment, the watermark embedding technique shown in Fig. 3.15 (Step 3) and Fig. 3.16 is actually based on two sub-processes; the first one is to match the cross-ratio reference points with zero watermark marking positions for easily tracking the zero watermark marking points after a text image has been attacked and the last one is to detect the zero watermarks and verify its integrity with the matching percentage method.

The watermark marking process under this method can be predefined the set of cross-ratio values and find the image center by following the steps and equations as described in the Section 3.4.1 and Section 3.7.1. After that, follow the steps as described below.

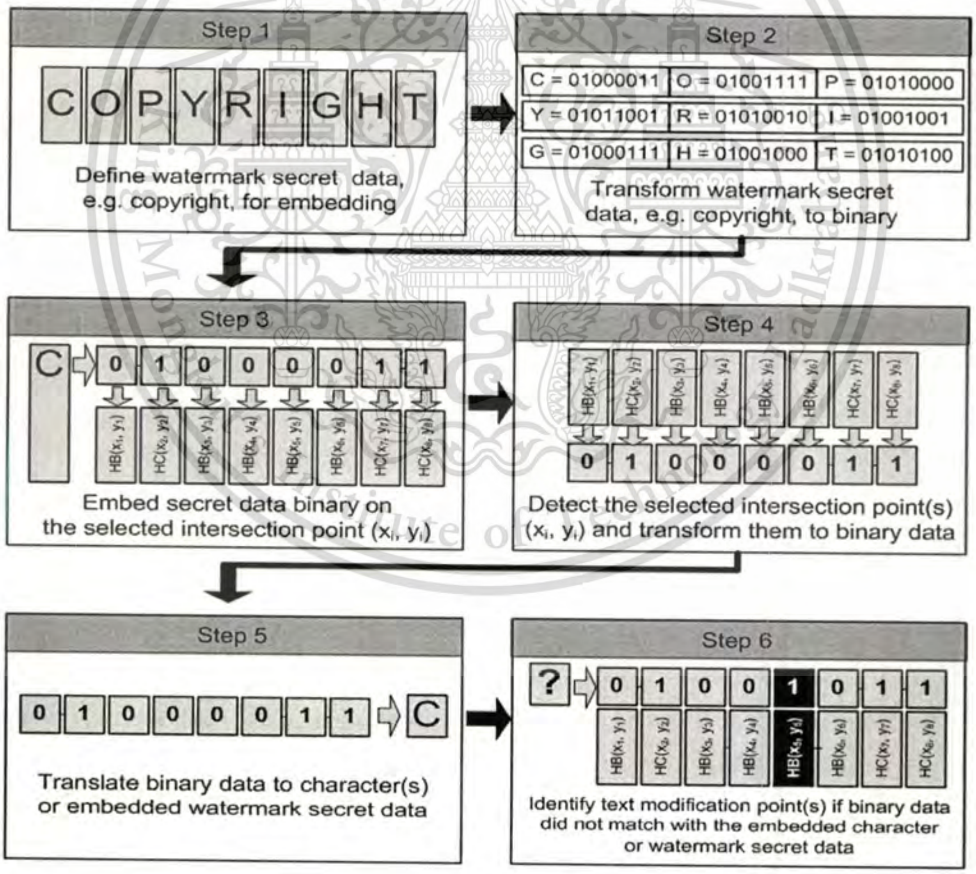


Figure 3.15 The six steps of embedding and detecting watermark secret data.

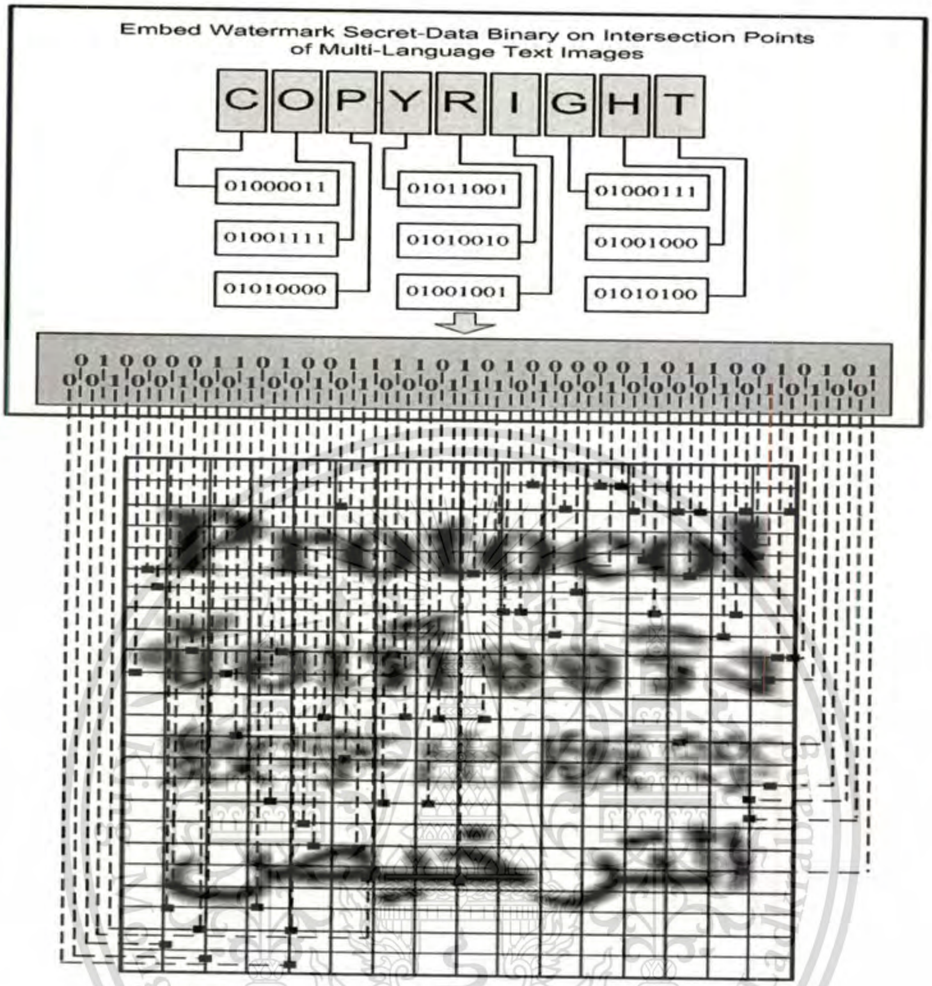


Figure 3.16 The technique to embed the secret data bits (0, 1) of watermark.

1. Find each of the primary-level watermark marking points (TCr_1 and TCr_2) on the right diagonal line (see Fig. 3.17, 3.18 and 3.19) as described in the Section 3.6.1.3

2. For each pair of (x_1, y_1) (x_2, y_2) level find an intersection, (x_i, y_i), of crossed line of each level drawn across (x_3, y_3) (x_4, y_4), as described in the Section 3.7.1.2, by applying Eqs (3.3) ~ (3.4).

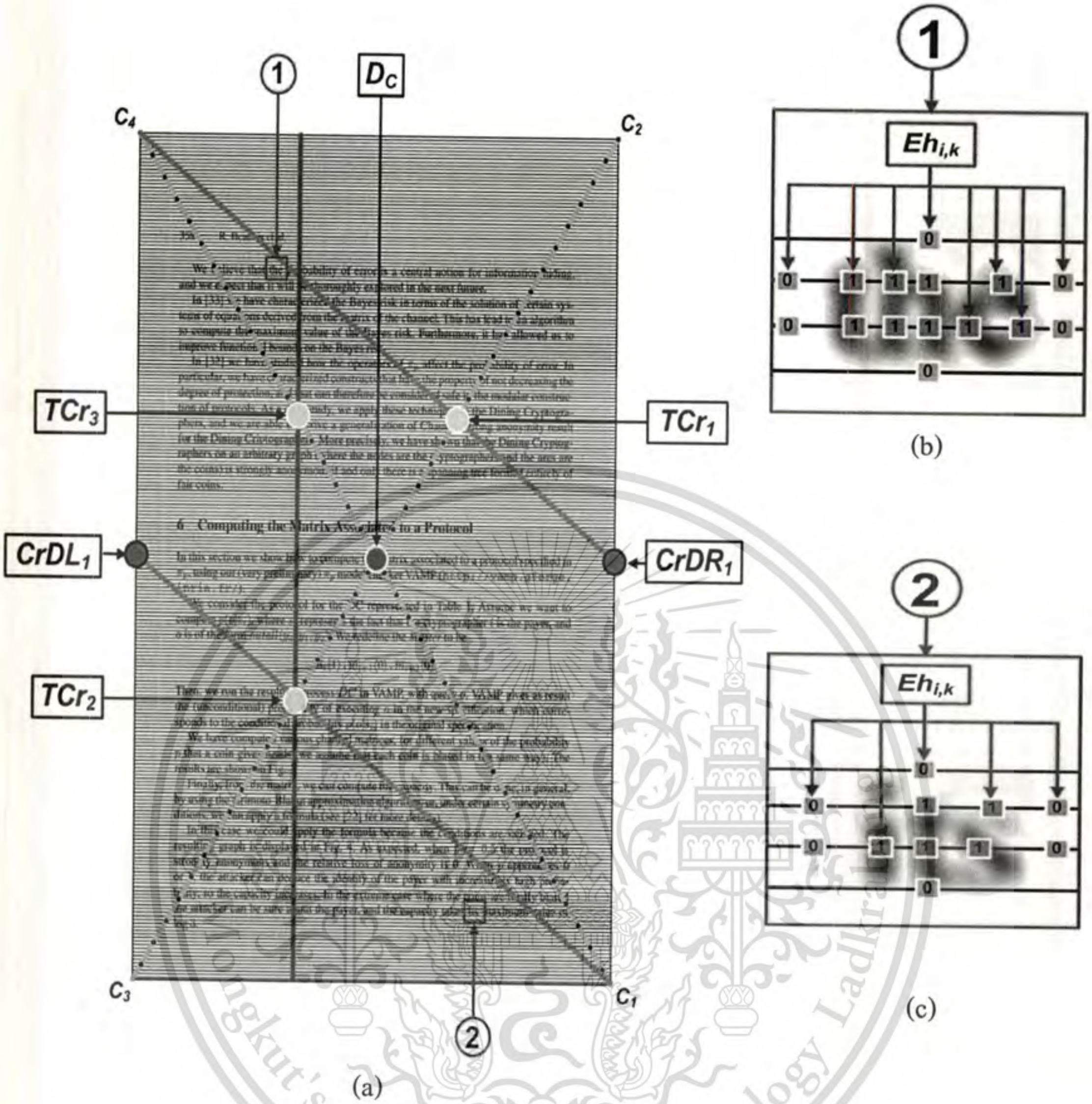
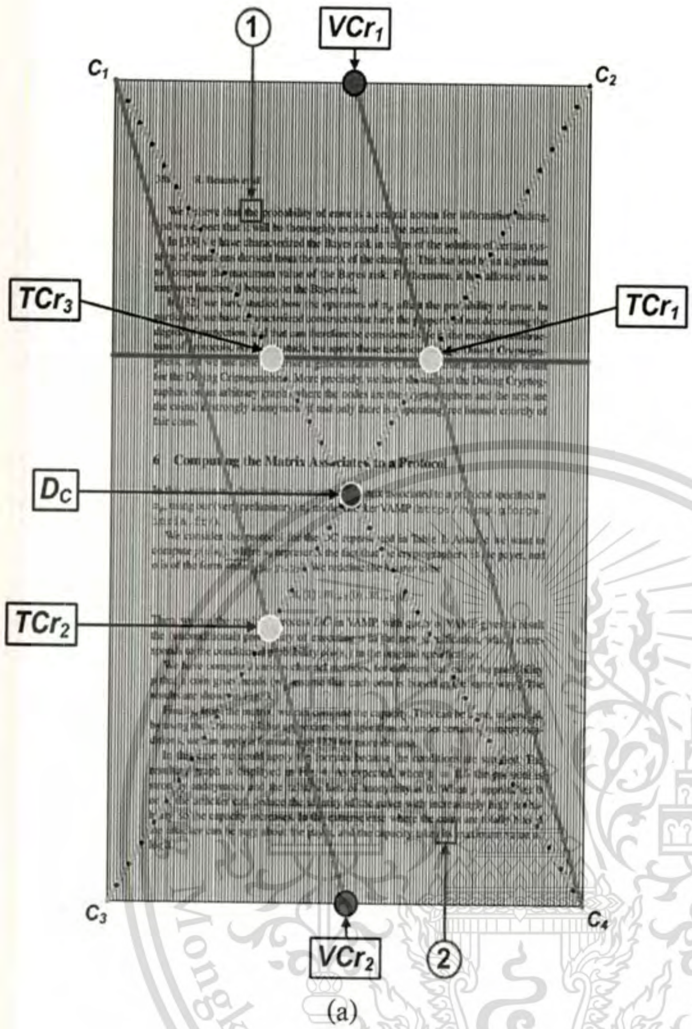
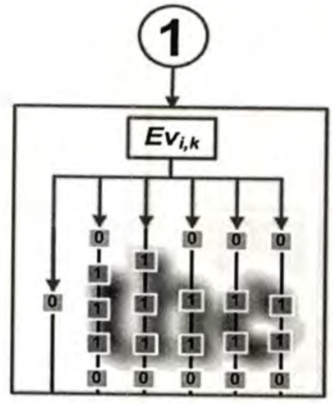


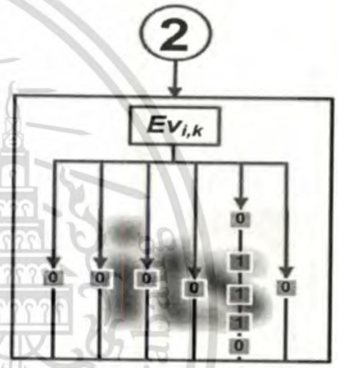
Figure 3.17 (a) Notations of main horizontal lines intersect two diagonal lines. (b) and (c) Notations of virtual main horizontal lines intersect text character lines.



(a)



(b)



(c)

Figure 3.18 (a) Notations of main vertical lines intersect text character lines. (b) and (c) Notations virtual main vertical lines intersect text character lines.

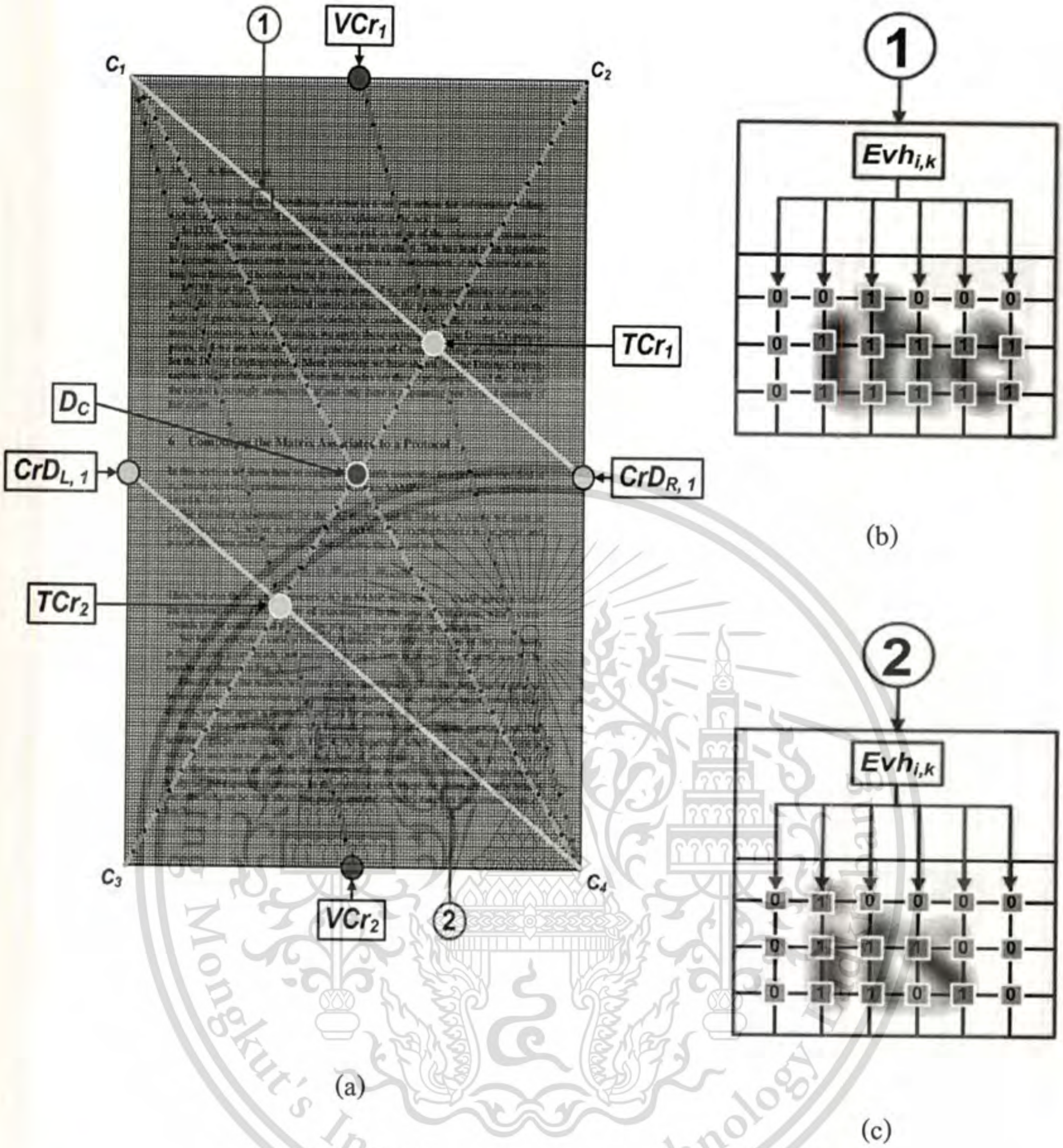


Figure 3.19 (a) Notations of main double (H & V) crossed-line intersection. (b) and (c) Notations of the main double (H & V) lines intersect text character lines.

3. Repeat steps, as described in the Section 3.7.1, for finding each secondary-level watermark marking point ($Eh_{i,k}$) at the prominent horizontal crossed-line intersection position where each horizontal line runs across a text-character skeleton line or blank area and its tone is less or equal and higher than 245 grayscale level if we need more effectively embedding and detecting. These prominent positions would be specified as the embedded bits; one or zero, of invisible zero watermarks ($Eh_{i,k}$) if their tones are lower than 245 or equal and higher than 245 respectively, as shown in Figs 3.17(b)-(c).

4. Find each of the primary-level watermark marking points (TCr_3) on the right diagonal line as described by Eqs. (3.5) ~ (3.6) below. Those points can be identified by using two corner points of the left diagonal line (C_1 and C_4), in combination with the image center point D_c and the predefined cross-ratio values (C_r). Here, $A = C_1$, $C_1 = (x_s, y_s)$, $B = TCr_3$, $C = D_c$ and $D = C_4$, $C_4 = (x_d, y_d)$.

5. Find each of the secondary-level watermark marking points ($V_{Cr,i}$) on the line (see Fig. 3.18) as described by Eqs. (3.5) ~ (3.6) below. Those points can be identified by using two corner points of the line C_1 C_2, C_3 C_4, in combination with the image central point VCr_i and the predefined cross-ratio values (C_r).Here, $A = C_1$ OR $C_3 = (x_s, y_s)$, $B = V_{Cr,i}$, $C = VCr_1$ OR VCr_2 and $D = C_2$ OR $C_4 = (x_d, y_d)$.

6. In case of vertical line, detect each prominent intersection position of each vertical line which runs across each text-character skeleton line or blank area where its tone is less or equal and higher than 245 grayscale level if we need more effectively embedding and detecting. These detected prominent positions would be specified as the embedded bits; one or zero, of invisible zero watermarks ($Ev_{i,k}$) if their tones are lower than 245 or equal and higher than 245 respectively, as shown in Fig. 3.18 (b)-(c).

7. In case of double (H & V) lines, each pair of $(x_{VCr1,i}, y_{VCr1,i})$ $(x_{VCr2,i}, y_{VCr2,i})$ level find an intersection, (x_i, y_i), of crossed line of each level drawn across $(X_{HCrL,i}, y_{HCrL,i})$ $(X_{HCrR,i}, y_{HCrR,i})$; by applying Eqs. (3.3) ~ (3.4). For each pair of $(x_{VCr1,i}, y_{VCr1,i})$ $(x_{VCr2,i}, y_{VCr2,i})$ find an intersection , $(X_{Evh,i,k}, y_{Evh,i,k})$, of crossed line of each level drawn across $(X_{HCrL,i}, y_{HCrL,i})$ $(X_{HCrR,i}, y_{HCrR,i})$, $(x_1, y_1) = (x_{VCr1,i}, y_{VCr1,i})$, $(x_2, y_2) = (x_{VCr2,i}, y_{VCr2,i})$, $(x_3, y_3) = (x_{HCrL,i}, y_{HCrL,i})$, $(x_4, y_4) = (x_{HCrR,i}, y_{HCrR,i})$.

8. Detect each prominent intersection position of each double (H & V) line which runs across each text-character skeleton line or blank area where its tone is less or equal and

higher than 245 grayscale level if we need more effectively embedding and detecting. These detected prominent positions would be specified as the embedded bits; one or zero, of invisible zero watermarks ($Evh_{i,k}$) if their tones are lower than 245 or equal and higher than 245 respectively, as shown in Fig. 3.19(b)-(c).

3.8.2 Watermark Detection Technique

3.8.2.1 Identify Angle and Edge of a Watermark-embedded Text Image

To detect a watermark from the text image, the four image corner points must first be detected. These angle and edge of a watermark-embedded text image can be identified by following the process as described in the Section 3.4.2.1.

3.8.2.2 Identify The Language Type of a Watermark-embedded Text Image

The language type of a watermark-embedded text image can be identified by following the procedure as described in the Section 3.5.3.2.

3.8.2.3 Detect The Watermark Bits at The Embedded Intersection Points

Like the Method 2, the matching percentage method, as described in the Section 3.5.3.3, will be used to detect the marked zero watermark.

3.8.2.4 Translate The Watermark Bits into The Embedded Secret Data

After getting the extracted bit values of each redundant block or layer, the next step is to translate them into characters or embedded watermark secret data, as shown in Fig. 3.15 (Step 5), and then compare them with the original embedded secret data. These comparisons will be done block by block or layer by layer and then check the percentage of matching.

3.8.2.5 Verify Text Image Integrity and Changing

If the extracted data does not match with the original embedded secret data, the unmatched bits would lead us to their positions, (x_i, y_i), where we can check their modifications, as shown in Fig. 3.15 (Step 6), such as some new text adding, reordering or deleting. This would help the copyright owners to trace back any change which be done with their original text images. At the same time, this technique can also prevent the copyright owners to be forced to accept their ownership with any fake text images which have been proved that embedded their verified watermark secret data. This situation may happen if they have applied the physical watermarking technique such as inter-word or line shifting, which some hackers may observe and know how to manipulate them by avoid touching the watermark embedding positions. That means the right watermarked data is still detected and extracted even them have been modified. This mistake acceptance would rarely occur under this new technique because all text characters are finely marked, stored and detected, so that a little bit change can eventually be identified.

CHAPTER 4

EXPERIMENTAL EVALUATION

4.1 Experimental Results of Method 1:

Applying The Actual Watermark Embedded Between Text Lines and The Cross Ratio Theory to Build up The Robustness of Text-image Watermark

Under this computer simulation experiment, 35 grayscale multi-language text images, size of 1240x1754 pixels and 150 dpi of resolution, were used to add 20 different invisible actual watermark patterns of length 100 bits, α is 3, block size of watermark 5x5 pixels/watermark pattern bit, as described in Section 3.4 of Chapter 3. The cross-ratio values used for watermark embedding and detecting were 120 values.

The results of experiment for digital embedding in 35 grayscale text images of four tested languages; English, Thai, Chinese and Arabic, applied with 20 various watermark patterns (see Figs. 4.1 ~ 4.5), through measuring of watermark values from correlation coefficient by fixing threshold value equal to 0.5 (if there are watermarks in text images with threshold value from 0.5 onward, if there is no watermarks its value must be less than 0.5) revealed the reasonable watermark robustness enhancement of the cross-ratio applying.

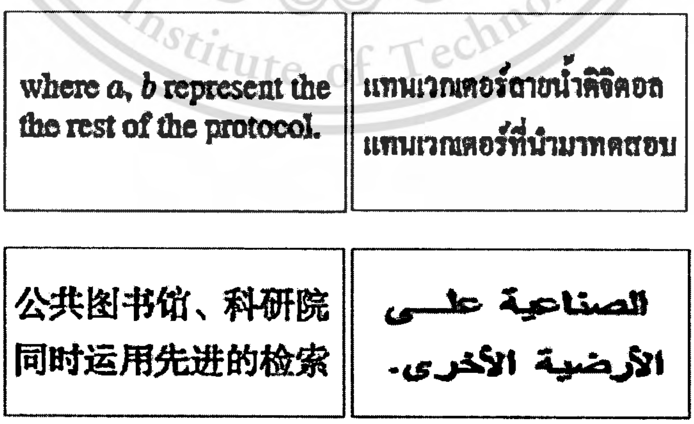


Figure 4.1 Show some invisible actual watermark pattern bits which were embedded between each text line of English, Thai, Chinese, and Arabic text images.

We believe that the probability of error is a central notion for information-hiding, and we expect that it will be thoroughly explored in the next future.

In [33] we have characterized the Bayes risk in terms of the solution of certain systems of equations derived from the matrix of the channel. This has lead to an algorithm to compute the maximum value of the Bayes risk. Furthermore, it has allowed us to improve functional bounds on the Bayes risk.

In [32] we have studied how the operators of π_p affect the probability of error. In particular, we have characterized constructs that have the property of not decreasing the degree of protection, and that can therefore be considered safe in the modular construction of protocols. As a case study, we apply these techniques to the Dining Cryptographers, and we are able to derive a generalization of Chaum's strong anonymity result for the Dining Criptographers. More precisely, we have shown that the Dining Cryptographers on an arbitrary graph (where the nodes are the cryptographers and the arcs are the coins) is strongly anonymous if and only there is a spanning tree formed entirely of fair coins.

6 Computing the Matrix Associated to a Protocol

In this section we show how to compute the matrix associated to a protocol specified in π_p, using our (very preliminary) π_p model checker VAMP (`http://vamp.gforge.inria.fr/`).

We consider the protocol for the DC represented in Table 1. Assume we want to compute $p(o|s_i)$, where s_i represents the fact that the cryptographer i is the payer, and o is of the form $\overline{outall}\langle y_o, y_1, y_2\rangle$. We redefine the *Master* to be

$$\overline{m_i}\langle 1\rangle \,.\, \overline{m_{i+1}}\langle 0\rangle \,.\, \overline{m_{i+2}}\langle 0\rangle$$

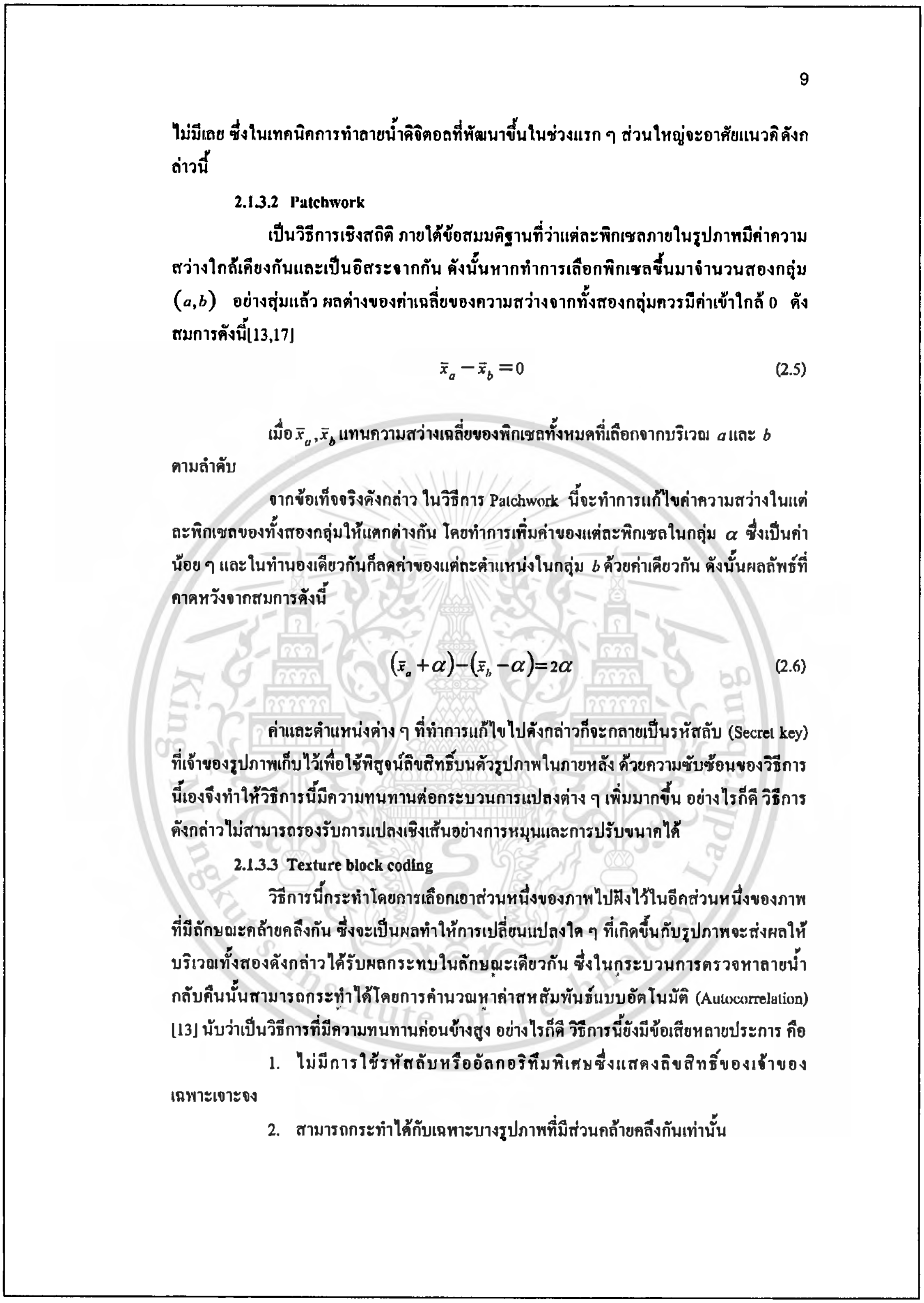
Then, we run the resulting process DC in VAMP, with query o. VAMP gives as result the (unconditional) probability of executing o in the new specification, which corresponds to the conditional probability $p(o|s_i)$ in the original specification.

We have computed various channel matrices, for different values of the probability p that a coin gives heads (we assume that each coin is biased in the same way). The results are shown in Fig. 3.

Finally, from the matrix, we can compute the capacity. This can be done, in general, by using the Arimoto-Blahut approximation algorithm, or, under certain symmetry conditions, we can apply a formula (see [22] for more details).

In this case we could apply the formula because the conditions are satisfied. The resulting graph is displayed in Fig. 4. As expected, when $p = 0.5$ the protocol is strongly anonymous and the relative loss of anonymity is 0. When p approaches 0 or 1, the attacker can deduce the identity of the payer with increasingly high probability, so the capacity increases. In the extreme case where the coins are totally biased the attacker can be sure about the payer, and the capacity takes its maximum value of $\log 3$.

Figure 4.2 Show some invisible actual watermark pattern bits which were embedded between each text line of English text image.



9

ไม่มีเลย ซึ่งในเทคนิคการทำลายน้ำดิจิตอลที่พัฒนาขึ้นในช่วงแรก ๆ ส่วนใหญ่จะอาศัยแนวคิดดังกล่าวนี้

2.1.3.2 Patchwork

เป็นวิธีการเชิงสถิติ ภายใต้ข้อสมมติฐานที่ว่าแต่ละพิกเซลภายในรูปภาพมีค่าความสว่างใกล้เคียงกันและเป็นอิสระจากกัน ดังนั้นหากทำการเลือกพิกเซลขึ้นมาจำนวนสองกลุ่ม (a,b) อย่างสุ่มแล้ว ผลต่างของค่าเฉลี่ยของความสว่างจากทั้งสองกลุ่มควรมีค่าเข้าใกล้ 0 ดังสมการดังนี้[13,17]

$$\bar{x}_a - \bar{x}_b = 0 \tag{2.5}$$

เมื่อ $\bar{x}_a, \bar{x}_b$ แทนความสว่างเฉลี่ยของพิกเซลทั้งหมดที่เลือกจากบริเวณ a และ b ตามลำดับ

จากข้อเท็จจริงดังกล่าว ในวิธีการ Patchwork นี้จะทำการแก้ไขค่าความสว่างในแต่ละพิกเซลของทั้งสองกลุ่มให้แตกต่างกัน โดยทำการเพิ่มค่าของแต่ละพิกเซลในกลุ่ม α ซึ่งเป็นค่าน้อย ๆ และในทำนองเดียวกันก็ลดค่าของแต่ละตำแหน่งในกลุ่ม b ด้วยค่าเดียวกัน ดังนั้นผลลัพธ์ที่คาดหวังจากสมการดังนี้

$$(\bar{x}_a + \alpha) - (\bar{x}_b - \alpha) = 2\alpha \tag{2.6}$$

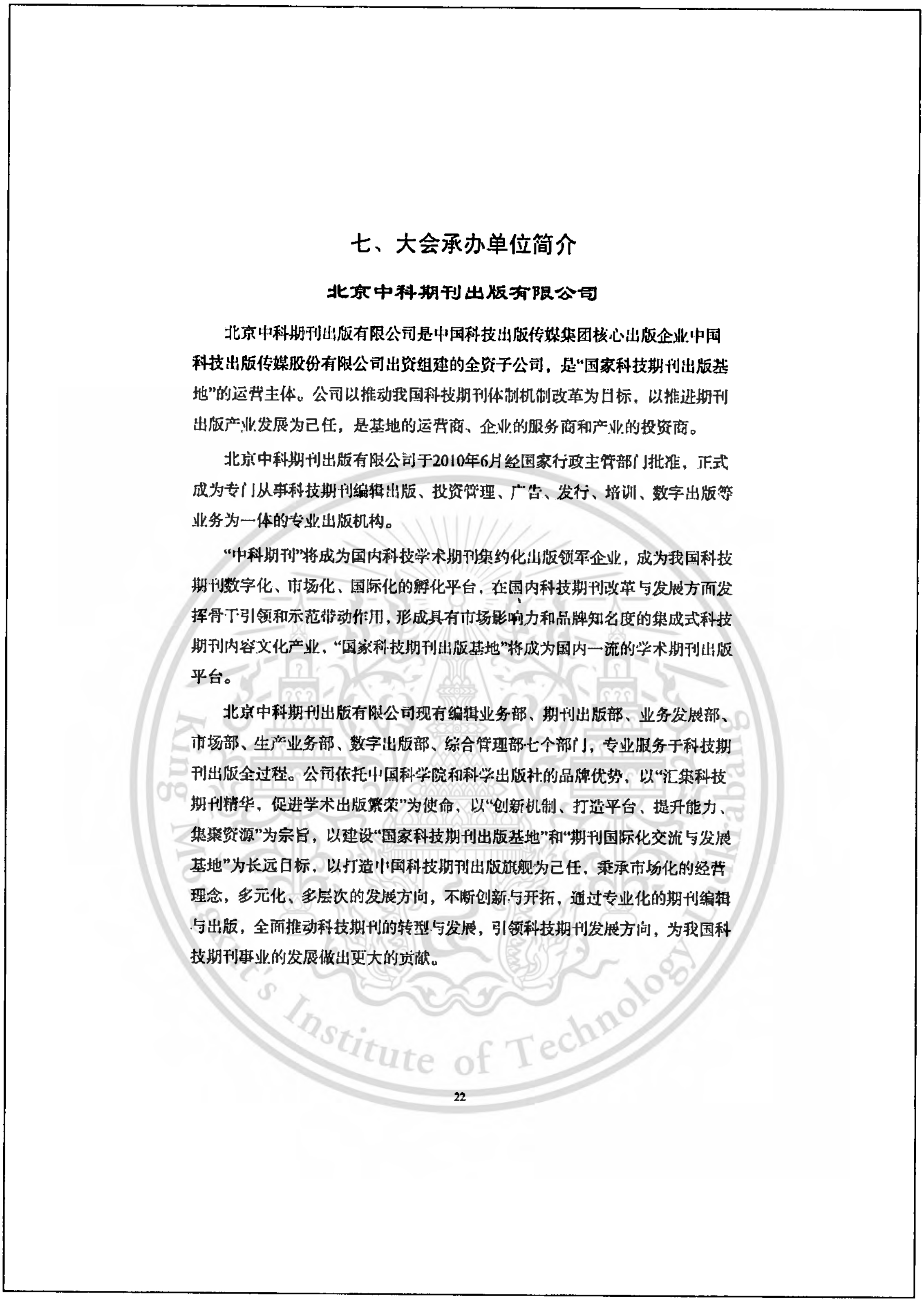
ค่าและตำแหน่งต่าง ๆ ที่ทำการแก้ไขไปดังกล่าวก็จะกลายเป็นรหัสลับ (Secret key) ที่เจ้าของรูปภาพเก็บไว้เพื่อใช้พิสูจน์ลิขสิทธิ์บนตัวรูปภาพในภายหลัง ด้วยความซับซ้อนของวิธีการนี้เองจึงทำให้วิธีการนี้มีความทนทานต่อกระบวนการแปลงต่าง ๆ เพิ่มมากขึ้น อย่างไรก็ดี วิธีการดังกล่าวไม่สามารถรองรับการแปลงเชิงเส้นอย่างการหมุนและการปรับขนาดได้

2.1.3.3 Texture block coding

วิธีการนี้กระทำโดยการเลือกเอาส่วนหนึ่งของภาพไปฝังไว้ในอีกส่วนหนึ่งของภาพที่มีลักษณะคล้ายคลึงกัน ซึ่งจะเป็นผลทำให้การเปลี่ยนแปลงใด ๆ ที่เกิดขึ้นกับรูปภาพจะส่งผลให้บริเวณทั้งสองดังกล่าวได้รับผลกระทบในลักษณะเดียวกัน ซึ่งในกระบวนการตรวจหาลายน้ำกลับคืนนั้นสามารถกระทำได้โดยการคำนวณหาค่าสหสัมพันธ์แบบอัตโนมัติ (Autocorrelation) [13] นับว่าเป็นวิธีการที่มีความทนทานค่อนข้างสูง อย่างไรก็ดี วิธีการนี้ยังมีข้อเสียหลายประการ คือ

1. ไม่มีการใช้รหัสลับหรืออัลกอริทึมพิเศษซึ่งแสดงลิขสิทธิ์ของเจ้าของเฉพาะเจาะจง
2. สามารถกระทำได้กับเฉพาะบางรูปภาพที่มีส่วนคล้ายคลึงกันเท่านั้น

Figure 4.3 Show some invisible actual watermark pattern bits which were embedded between each text line of Thai text image.



七、大会承办单位简介

北京中科期刊出版有限公司

北京中科期刊出版有限公司是中国科技出版传媒集团核心出版企业中国科技出版传媒股份有限公司出资组建的全资子公司，是“国家科技期刊出版基地”的运营主体。公司以推动我国科技期刊体制机制改革为目标，以推进期刊出版产业发展为己任，是基地的运营商、企业的服务商和产业的投资商。

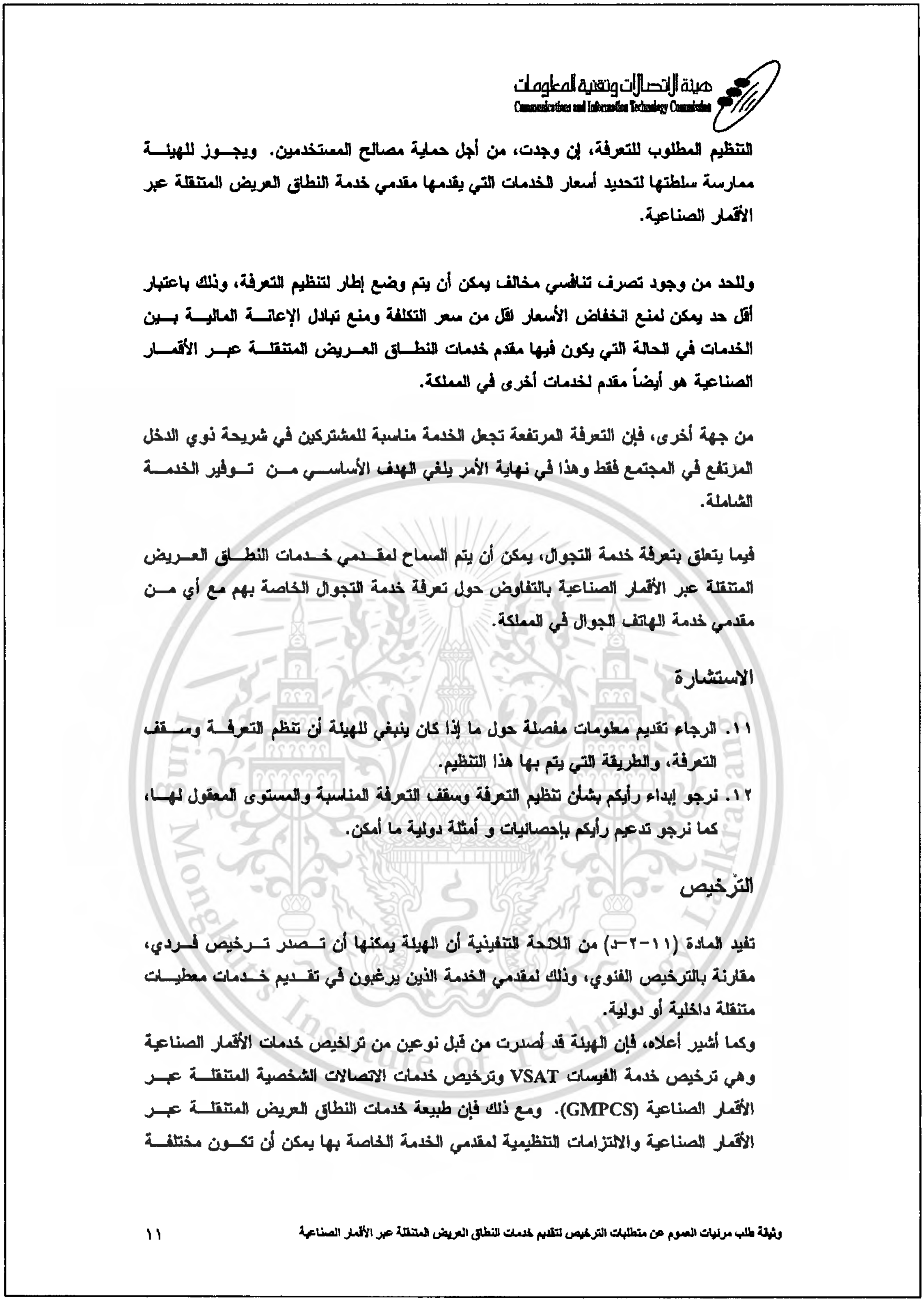
北京中科期刊出版有限公司于2010年6月经国家行政主管部门批准，正式成为专门从事科技期刊编辑出版、投资管理、广告、发行、培训、数字出版等业务为一体的专业出版机构。

“中科期刊”将成为国内科技学术期刊集约化出版领军企业，成为我国科技期刊数字化、市场化、国际化的孵化平台，在国内科技期刊改革与发展方面发挥骨干引领和示范带动作用，形成具有市场影响力和品牌知名度的集成式科技期刊内容文化产业，“国家科技期刊出版基地”将成为国内一流的学术期刊出版平台。

北京中科期刊出版有限公司现有编辑业务部、期刊出版部、业务发展部、市场部、生产业务部、数字出版部、综合管理部七个部门，专业服务于科技期刊出版全过程。公司依托中国科学院和科学出版社的品牌优势，以“汇集科技期刊精华，促进学术出版繁荣”为使命，以“创新机制、打造平台、提升能力、集聚资源”为宗旨，以建设“国家科技期刊出版基地”和“期刊国际化交流与发展基地”为长远目标，以打造中国科技期刊出版旗舰为己任，秉承市场化的经营理念，多元化、多层次的发展方向，不断创新与开拓，通过专业化的期刊编辑与出版，全面推动科技期刊的转型与发展，引领科技期刊发展方向，为我国科技期刊事业的发展做出更大的贡献。

22

Figure 4.4 Show some invisible actual watermark pattern bits which were embedded between each text line of Chinese text image



هيئة الاتصالات وتقنية المعلومات
Communications and Information Technology Commission

التنظيم المطلوب للتعرفة، إن وجدت، من أجل حماية مصالح المستخدمين. ويجوز للهيئة ممارسة سلطتها لتحديد أسعار الخدمات التي يقدمها مقدمي خدمة النطاق العريض المتنقلة عبر الأقمار الصناعية.

وللحد من وجود تصرف تنافسي مخالف يمكن أن يتم وضع إطار لتنظيم التعرفة، وذلك باعتبار أقل حد يمكن لمنع انخفاض الأسعار اقل من سعر التكلفة ومنع تبادل الإعانة المالية بين الخدمات في الحالة التي يكون فيها مقدم خدمات النطاق العريض المتنقلة عبر الأقمار الصناعية هو أيضاً مقدم لخدمات أخرى في المملكة.

من جهة أخرى، فإن التعرفة المرتفعة تجعل الخدمة مناسبة للمشتركين في شريحة ذوي الدخل المرتفع في المجتمع فقط وهذا في نهاية الأمر يلغي الهدف الأساسي من توفير الخدمة الشاملة.

فيما يتعلق بتعرفة خدمة التجوال، يمكن أن يتم السماح لمقدمي خدمات النطاق العريض المتنقلة عبر الأقمار الصناعية بالتفاوض حول تعرفة خدمة التجوال الخاصة بهم مع أي من مقدمي خدمة الهاتف الجوال في المملكة.

الاستشارة

١١. الرجاء تقديم معلومات مفصلة حول ما إذا كان ينبغي للهيئة أن تنظم التعرفة وسقف التعرفة، والطريقة التي يتم بها هذا التنظيم.

١٢. نرجو إبداء رأيكم بشأن تنظيم التعرفة وسقف التعرفة المناسبة والمستوى المعقول لها، كما نرجو تدعيم رأيكم بإحصائيات و أمثلة دولية ما أمكن.

الترخيص

تفيد المادة (١١-٢-د) من اللائحة التنفيذية أن الهيئة يمكنها أن تصدر ترخيص فردي، مقارنة بالترخيص الفئوي، وذلك لمقدمي الخدمة الذين يرغبون في تقديم خدمات معطيات متنقلة داخلية أو دولية.
وكما أشير أعلاه، فإن الهيئة قد أصدرت من قبل نوعين من تراخيص خدمات الأقمار الصناعية وهي ترخيص خدمة الفيسات VSAT وترخيص خدمات الاتصالات الشخصية المتنقلة عبر الأقمار الصناعية (GMPCS). ومع ذلك فإن طبيعة خدمات النطاق العريض المتنقلة عبر الأقمار الصناعية والالتزامات التنظيمية لمقدمي الخدمة الخاصة بها يمكن أن تكون مختلفة

وثيقة طلب مرئيات العموم عن متطلبات الترخيص لتقديم خدمات النطاق العريض المتنقلة عبر الأقمار الصناعية ١١

Figure 4.5 Show some invisible actual watermark pattern bits which were embedded between each text line of Arabic text image

Firstly, tested the controlled text image without watermark by comparing with watermark pattern could obtain value of correlation coefficient = 0, while the image with watermark pattern obtained value of correlation coefficient = 1.

After that testing the cross-ratio watermarking robustness with nine attacks, including three projective-geometric distortions; shearing, scaling and rotating and five image manipulations; compression, sharpness, contrast, blur masking and noise signal adding. Under this testing, all watermark embedded text images had been routinely detected their edges and angles by applying the Hough transform method, as described in the Section 3.2, before watermark carefully detecting. The actual watermark detecting results can be classified their effects into three groups as follows:

Group I: No effect on actual watermark robustness has been found under attacking of sharpness which shown the correlation coefficient = 1, for all percentage of sharpness filtering variation, range 0 – 100% (see Table 4.1).

Group II: Very low effect on actual watermark robustness has been found under attacking of compression, at the range 60 – 100 % of JPEG compression quality, scaling, at the range 11 – 60% of scaling factor (see Fig. 4.6), blur, at the range 3x3 – 13x13 of blur filtering mask size, contrasting, at the range 1 – 45%, shearing, at the range 0 – 0.05 and rotating, at the range of angle between 1- 4 degrees (see Fig. 4.7) which shown the acceptable correlation coefficient values between 0.5 - 1, for all kinds of attacking values specified above.

Group III: High effect on actual watermark robustness has been found under attacking of Salt & Pepper noise signal adding, at the range higher than 1.5%, and Gaussians noise signal adding, at all ranges shown unacceptable correlation coefficient values which mostly near "0". It has shown that noise disturbing signals is the most complicated factor to affect watermark detecting, if there are more disturbing signals, detecting for watermarks can be lower to be done.

Table 4.1 Experiment results

Technique	Process	Applied to	Imperceptibility	Language	Approximately Hiding Bit Capacity	Correlation Coefficient / Watermark Matching Percentage After Attacking	Integrity Verification?
Method 1: Applying the actual watermark embedded between text lines and the cross ratio theory to build up the robustness of text-image watermark	Actual watermark was embedded between text lines	Text Image	Invisible	Multi-language (Thai, Chinese, English and Arabic)	12,500 bits/A4 page	-Sharpness (0-100 %): $\gamma = 1$ -JPEG compression (60-100 %): $\gamma = 0.79 - 1.0$ -Blur(3x3-13x13 of mask size): $\gamma = 0.6 - 1.0$ -Contrasting(1-45 %): $\gamma = 0.5 - 1.0$ -Shearing(0 - 0.05 %): $\gamma = 0.5 - 1.0$ -Rotation(0 - 4 degree): $\gamma = 0.5 - 1.0$ -Noise signal adding (Salt & Pepper 0-1.5 %): $\gamma = 0.5$ -Scaling(11 - 60 %): $\gamma = 0.5 - 1.0$ - Brightness(5 %): $\gamma = 0.5$ -Shearing(0 - 0.05): $\gamma = 0.5 - 1.0$	No

Table 4.1 Experiment results *(cont.)*

Technique	Process	Applied to	Imperceptibility	Language	Approximately Hiding Bit Capacity	Correlation Coefficient / Watermark Matching Percentage After Attacking	Integrity Verification?
Method 2: Applying the single reference-line intersection of horizontal line and the cross ratio theory to increase the hiding-bit capacity and original-text verification performance of text-Image watermark with zero watermark embedded	Zero watermark was embedded along horizontal line run across character skeleton line	Text Image	Invisible	Multi-language (Thai, Chinese, English and Arabic)	Up to 33,500 bits/ A4 page	-Sharpness (0 – 100%): 98.25 – 99.50% -JPEG compression (0 – 100%): 98 – 100% -Blur (3x3-15x15 of mask size): 99.10% -Contrasting (10 – 50%): 100% -Shearing (Shift 0.00 – 0.05): 98.25 – 100% -Rotation (5-35 degree): 95.50% -Noise signal adding (Salt & Pepper: 1 – 15%): 99.89 – 100%): -Scaling (10 - 120%): 94% -Text adding (1 – 10 Text groups): 98.75 % -Text deleting (1 – 10 Text groups): 97.00 % -Text reordering (1 – 10 Text groups): 97.80 %	Yes

Table 4.1 Experiment results *(cont.)*

Technique	Process	Applied to	Imperceptibility	Language	Approximately Hiding Bit Capacity	Correlation Coefficient / Watermark Matching Percentage After Attacking	Integrity Verification?
Method 3: Applying the single reference-line intersection of additional vertical line and the cross-ratio theory to increase more watermark-hiding capacity and original-text verification performance of text-image watermark with zero watermark embedded	Zero watermark was along vertical line run across character skeleton line	Text Image	Invisible	Multi-language (Thai, Chinese, English and Arabic)	Up to 38,053 bits/ A4 page	-Sharpness (0 – 100%): 100% -JPEG compression (40 – 100%): 96 – 100% -Blur (3x3-15x15 of mask size): 12.5 – 46.0% -Contrasting (10 – 50%): 99.59 -99.72 % -Shearing (Shift 0.00 – 0.05): 86 – 100% -Rotation (5-35 degree): 81 – 87% -Noise signal adding: (Salt & Pepper: 1 – 15%): 99.73 – 99.97% -Scaling (10 - 120%): 75 – 100% - Text adding(1 – 10 Text groups): 99.68% - Text deleting(1 – 10 Text groups): 98.70% - Text reordering(1 – 10 Text groups): 99.40%	Yes

Table 4.1 Experiment results *(cont.)*

Technique	Process	Applied to	Imperceptibility	Language	Approximately Hiding Bit Capacity	Correlation Coefficient / Watermark Matching Percentage After Attacking	Integrity Verification?
Method 4: Applying the double reference-line intersection of vertical and horizontal lines and the double cross-ratio method to increase maximum watermark-hiding capacity and original-text verification precision of text-image watermark with zero watermark embedded	Zero watermark was embedded along horizontal and vertical lines which run across character skeleton lines	Text Image	Invisible	Multi-language (Thai, Chinese, English and Arabic)	Up to 240,608 bits/ A4 page	-Sharpness (0 – 100%): 100% -JPEG compression (0 – 100%): 99.53 – 100% -Blur (3x3 - 5x5 of mask size): 97.0 – 99.5% -Contrasting (10 – 50%): 100% -Shearing (Shift 0.00 – 0.05): 99.3 - 100% -Rotation (5-35 degree): 99.14 – 99.85% -Noise signal adding (Salt & Pepper: 1 – 15%): 99.33 – 99.97% -Scaling (10 – 120%): 97.95 -100% -Text adding(1 – 10 Text groups): 99.72 % -Text deleting(1 – 10 Text groups): 99.64 % -Text reordering(1 – 10 Text groups): 99.62 %	Yes

Table 4.1 Experiment results *(cont.)*

Technique	Process	Applied to	Imperceptibility	Language	Approximately Hiding Bit Capacity	Correlation Coefficient / Watermark Matching Percentage After Attacking	Integrity Verification?
Method 5: Applying the single horizontal, vertical and double reference-line intersection and the double cross-ratio method to create the multi-layer watermarking for increasing maximum watermark-hiding capacity and original-text verification precision of text-image watermark with zero watermark embedded.	Zero watermark was embedded along the single horizontal (H), vertical (V) and double (H & V) lines which run across character skeleton lines and run across one another	Text Image	Invisible	Multi-language (Thai, Chinese, English and Arabic)	Up to 1,545,738 bits/ A4 page (11 layers)	-Sharpness (0 – 100%): 100% (H or V) -JPEG compression (0 – 100%): 99.53 – 100% (HV) -Blur (3x3 - 5x5 of mask size): 97.0 – 99.5% (HV) -Contrasting (10 – 50%): 99.59 – 100% (H, V or HV) -Shearing (Shift 0.00 – 0.05): 99.3 – 100% (HV) -Rotation (5-35 degree): 99.14 – 99.85% (HV) -Noise signal adding: (Salt & Pepper 1 – 15%): 99.73 – 99.97% (H or V) -Scaling (10 – 120%): 97.95 – 100% (HV) -Text adding(1 – 10 Text groups): 99.33% -Text deleting(1 – 10 Text groups): 98.63% -Text reordering(1 – 10 Text groups): 99.24%	Yes

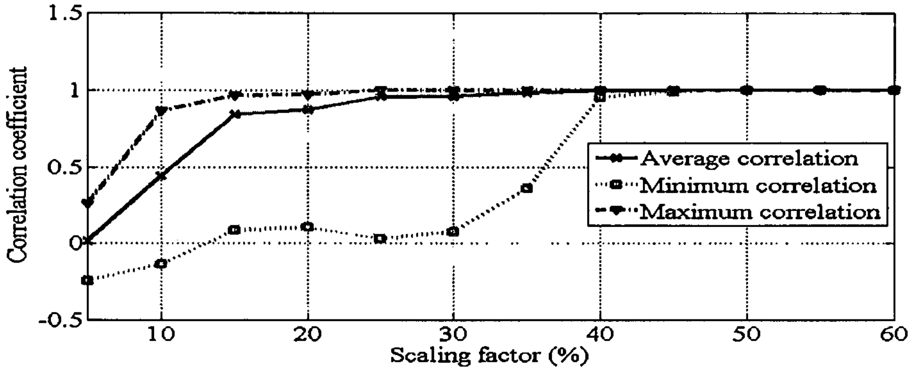


Figure 4.6 Correlation coefficient of watermark detection after scaling attack.

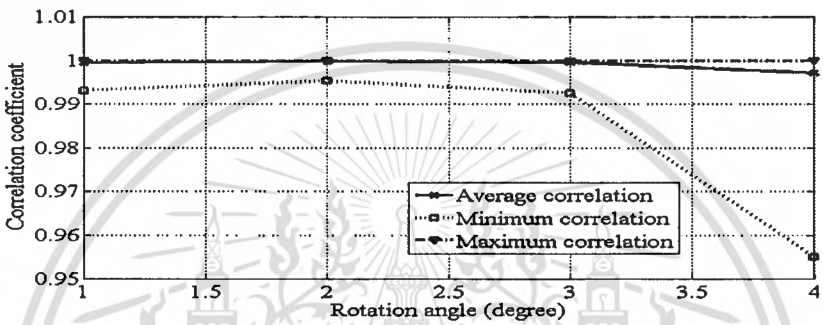


Figure 4.7 Correlation coefficient values of watermark detection after rotating attack.

4.2 Experimental Results of Method 2: Applying The Single Reference-line Intersection of Horizontal Line and The Cross Ratio Theory to Increase The Hiding-bit Capacity and Original-text Verification Performance of Text-image Watermark with Zero Watermark Embedded

Since the Method 1 is based on the actual watermark embedded between text lines, so that they have given us considerably low hiding bit capacity of watermark secret data and so easy to be destroyed by copying process. Thus, in the Method 2, we have tried to solve these weak points by applying the horizontal crossed-line intersection which virtually ran across the skeleton lines of each character.

In this experiment, the images used for testing are 35 grayscale multi-language text images. Each of grayscale text images, size of 1240x1754 pixels and 150 dpi of resolution, was used to make reference horizontal crossed-line intersections with text-character skeleton lines, so called the single reference-line intersection of horizontal line, as described in Section 3.5 of Chapter 3, in order to mark them as a position for zero watermark marking, one intersection defined as one bit of zero watermark data. The cross-ratio values used for zero watermark marking and detecting for one 50 text-line page was approximately about 33,500 values.

The results of experiment for digital marking on 35 grayscale text images of four tested languages; English, Thai, Chinese and Arabic, applied with 35 various zero watermark patterns, through measuring of watermark values from the percentage of matching zero watermark positions, one-to-one, between its original watermarked positions with the attacked one. This virtually-embedded watermark is detected if found the grayscale level of a character skeleton line, at the virtually-marked intersection position, is less than 245.

At first, tested of the controlled virtually-watermarked text image without attacking by comparing with the plotted zero watermark positioning pattern, it got 100% of matching percentage or zero percentage of error.

Under the watermark detecting process, all watermark embedded text images had been detected their edges and angles by applying the Hough transform method, as described in the Section 2.1.5 of Chapter 2. After that the type of language had been identified with the aspect ratio applying, as described in the Section 3.3 of Chapter 3. These language type identification results gave us the average reversed aspect ratio range of each language type; English, Thai, Chinese, and Arabic.

From the above results we can roughly conclude the reversed aspect ratio range (*Rr*) and the average reversed aspect ratio (*ARr*) of each tested language as shown in Table 4.2. Hence, these ranges and averages would be used as the reference reversed aspect ratio range (*Rr*) and the reference average reversed aspect ratio (*ARr*) of each tested language; English, Thai, Chinese or Arabic, for identifying the language type of the other tests as well. So that the more quickly a watermark embedded text image of one specific language could be identified its language identity, the more efficiently to find out the watermark embedding pattern.

Table 4.2 The range of reversed aspect ratio of the tested languages

Language Type	Reversed Aspect Ratio Range (Rr)	Average Reversed Aspect Ratio (ARr)
English	1.7767 - 1.8659	1.8089
Thai	1.9320 - 2.0871	2.0042
Chinese	1.0982 - 1.3421	1.2208
Arabic	1.0260 - 1.0828	1.0508

On testing, it begins with the virtually embedding process which each reference horizontal crossed line is virtually run over each text-character skeleton lines of each text character of each language to mark the intersected positions for virtually watermark marking. This embedding step revealed that this text character-line intersection watermarking technique can effectively create a novel zero watermark on any text images of any languages without any actual watermark really embedding, as per the requirement of the first objective, as specified in Section 1.2 of Chapter 1.

This horizontal-line-intersection zero watermarking method also overcomes the limitation of hiding-bit capacity, as required in the second objective. It has shown that one page of English text image, comparing with the other languages, can be contributed the biggest volume of hiding-bit capacity, up to 33,500 bits (see Fig. 4.8). This breakthrough came from the nature of the English text font itself let it has number of characters on one text line and number of text lines on one page more than the other three languages. However, these hiding-bit capacities contributed by Thai, Chinese and Arabic texts with this intersection technique are still significantly larger than other existing watermarking techniques such as inter-word shifting, line shifting, letter characteristic changing, text color changing, vowel recreating and synonym substituting which can contribute its hiding-bit capacity just only low-to-moderate level capacity, at range 25 - 2,500 bits/A4-page.

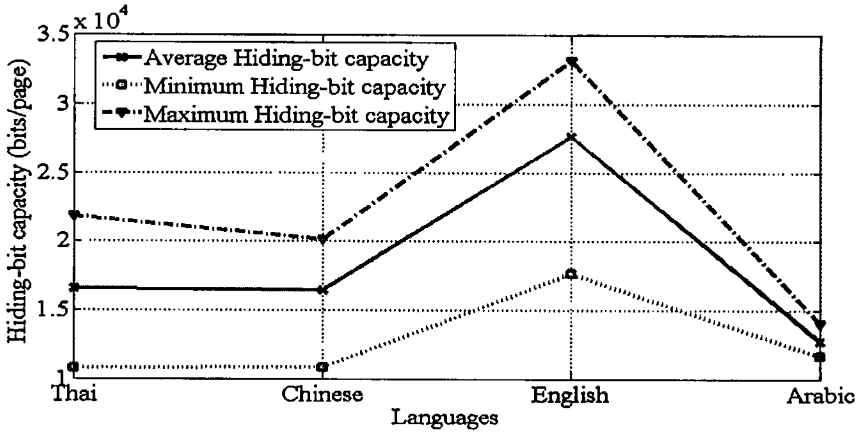


Figure 4.8 The hiding-bit capacity (bits/page) contributed by the horizontal crossed line.

The experiment revealed that the volume of watermark hiding-bit capacity varies according to the number of skeleton lines of each text character of each language, font size, number of characters/letters per text line, number of text lines per page and number of reference horizontal crossed lines which able to prominently draw across each text character. These factors would directly contribute the text hiding-bit capacity of watermark embedding of each language. This means the more text character – reference horizontal line intersection points are selected to mark for watermark, the more hiding-bit capacity of watermark would be got.

The Geometric and image manipulation attacks have shown the remarkable achievement of the third objective, as specified in Section 1.2 of Chapter 1, with the effect of the cross ratio theory to strengthen the text-image watermarking robustness, under the horizontal crossed-line intersection, against some possible attacks such as compression, sharpness, noise signal adding, shearing (see Fig. 4.9) and rotating (see Fig. 4.10). The results have shown that these attacks did not significantly affect the watermark detecting. They affected the virtually embedded watermark detecting just only 0.5 – 4.5 percent of error. This means the matching percentage of watermark detecting is mostly higher than 95.5 %.

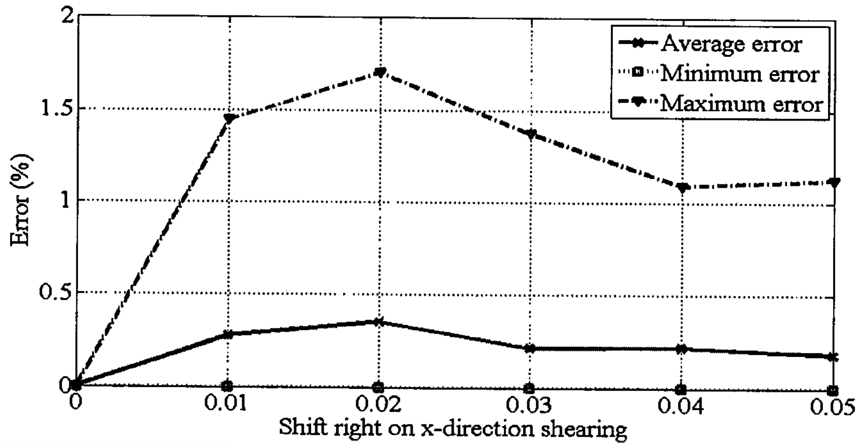


Figure 4.9 The error (%) level of horizontal crossed-line watermark detection after shearing attack.

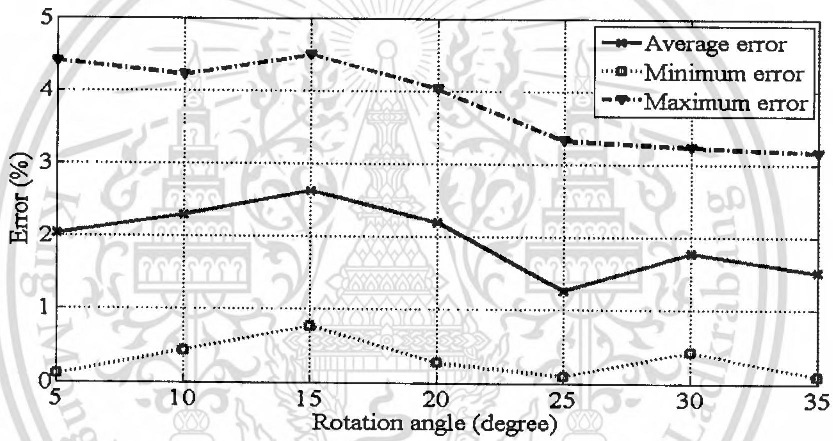


Figure 4.10 The error (%) level of horizontal crossed-line watermark detection after rotating attack.

Figs. 4.11 ~ 4.13 have shown another achievement of the fourth objective, as specified in Section 1.2 of Chapter 1, with the effect of the cross ratio theory to strengthen the original-text verification performance against some possible manipulations such as text adding, reordering and deleting. The results have shown that the horizontal line intersection method with the cross ratio theory applying can sensitively detect even a few changes such as 1 - 10 text groups adding, reordering and deleting which found the maximum level of 1.25%, 2.2 % and 3.0 % of errors respectively.

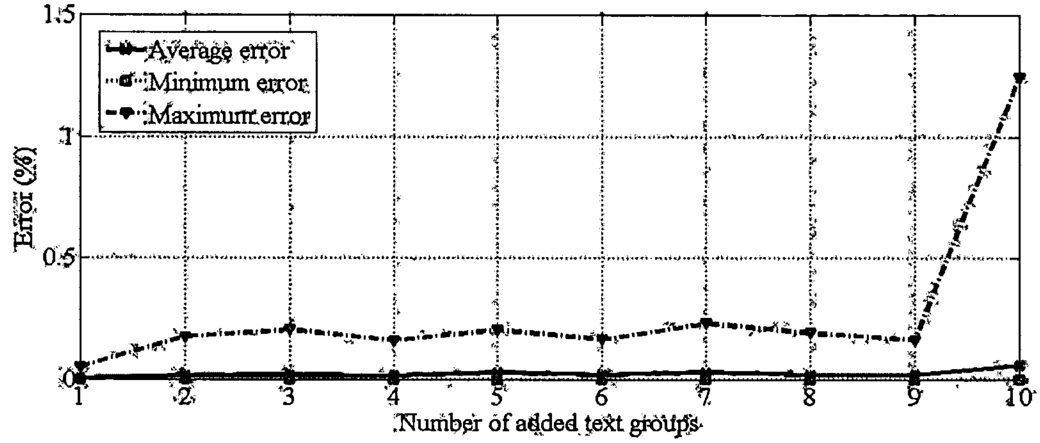


Figure 4.11 The error (%) level of horizontal crossed-line watermark detection after text adding.

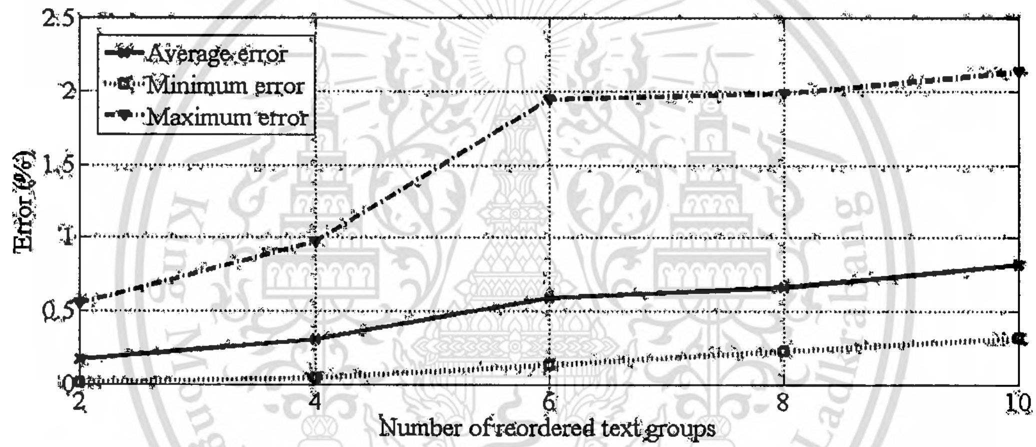


Figure 4.12 The error (%) level of horizontal crossed-line watermark detection after text reordering.

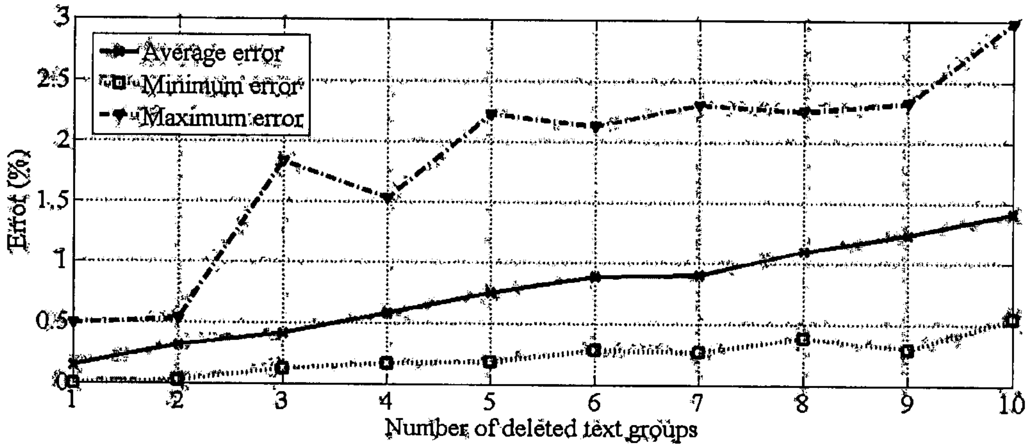


Figure 4.13 The error (%) level of horizontal crossed-line watermark detection after text deleting.

4.3 Experimental Results of Method 3: Applying The Single Reference-line Intersection of Additional Vertical Line and The Cross-Ratio Theory to Increase More Watermark-hiding Capacity and Original-text Verification Performance of Text-image Watermark with Zero Watermark Embedded

After studying the structure of character skeleton lines of English, Thai, Chinese and Arabic languages, in the Method 2, we had found that some character skeleton lines of some languages, such as English and Thai mostly have their skeleton lines in the vertical direction which the horizontal crossed line could not run and cut along them. This limitation of the horizontal line means some text manipulations, such as text adding, reordering and deleting, would not be detected. Thus, the Method 3 has been designed for exploring the advantages of the vertical crossed line.

Similarly, the experiment of Section 4.2, 35 grayscale multi-language text images were used to test. Each of grayscale text images, size of 1240x1754 pixels and 150 dpi of resolution, was used to make reference vertical crossed-line intersections with text-character skeleton lines, so called the single reference-line intersection of vertical line, as described in Section 3.6 of Chapter 3, in order to mark them as a position for zero watermark marking along the vertical lines, one intersection defined as one bit of zero watermark data. The cross-ratio values used for zero watermark marking and detecting for one 50 text-line page was approximately about 38,000 values.

The results of this experiment for digital marking on 35 grayscale text images of four tested languages; in English, Thai, Chinese and Arabic, applied with 35 various virtual watermark patterns, through measuring of watermark values from the percentage of matching virtual watermark positions, one-to-one, between its original watermarked positions with the attacked one. This virtually-embedded watermark is detected if found the grayscale level of a character skeleton line, at the virtually-marked intersection position, is less than 245.

The first step is tested of the controlled virtually-watermarked text image without attacking by comparing with the plotted virtual watermark positioning pattern, it got 100% of matching percentage or zero percentage of error.

The testing begins with the virtually embedding process which each reference vertical crossed line is virtually run over each text-character skeleton lines of each text character of each language to mark the intersected positions for virtually watermark marking. This embedding step revealed that this text character-line intersection watermarking technique, applied with the vertical lines, can also effectively create a real invisible watermark on any text images of any languages without any actual watermark really embedding, like working with the horizontal lines.

Moreover, this experiment shown that the reference vertical line intersection on the Arabic text page could contribute invisible virtual watermark hiding-bit capacity up to 40,000 bits, a little bit higher than the hiding-bit capacity of the horizontal line (see Fig. 4.14). This higher capacity came from the characteristic of the Arabic characters which mostly have many longer horizontal skeleton lines than other languages.

The geometric and image manipulation attacks have shown the achievement of the third objective, as specified in Section 1.2 of Chapter 1, with the effect of the cross ratio theory to strengthen the text-image watermarking robustness, under the vertical crossed-line intersection, against some possible attacks such as compression (see Fig. 4.15), sharpness, noise signal adding and shearing. The results have shown that these attacks affected the watermark detecting in the low level, about 0 – 14 percent of error. This means the matching percentage of watermark detecting is mostly higher than 86 %.

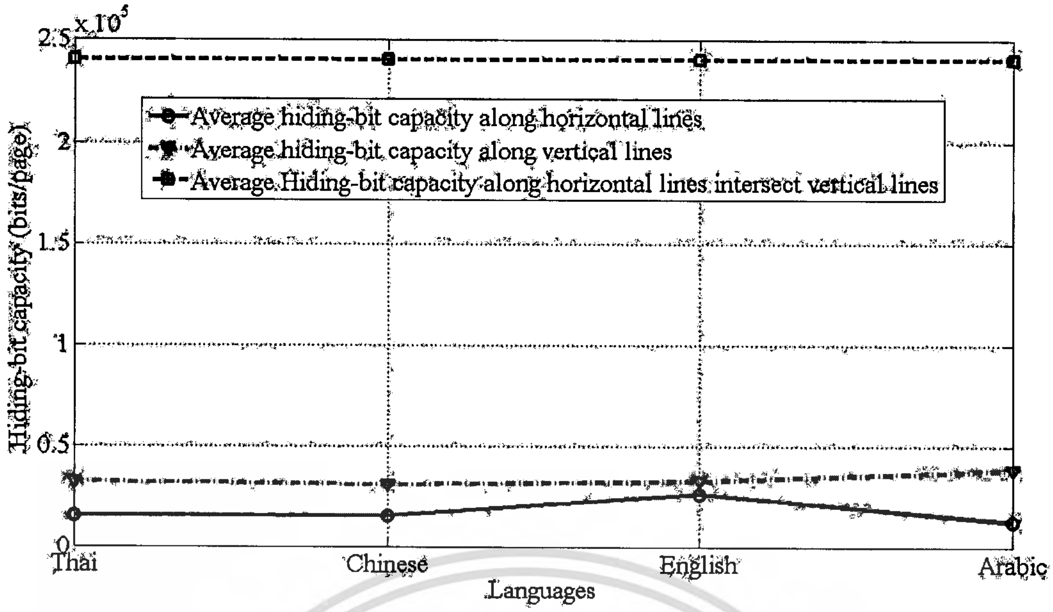


Figure 4.14 The hiding-bit capacity contributed by vertical crossed line.

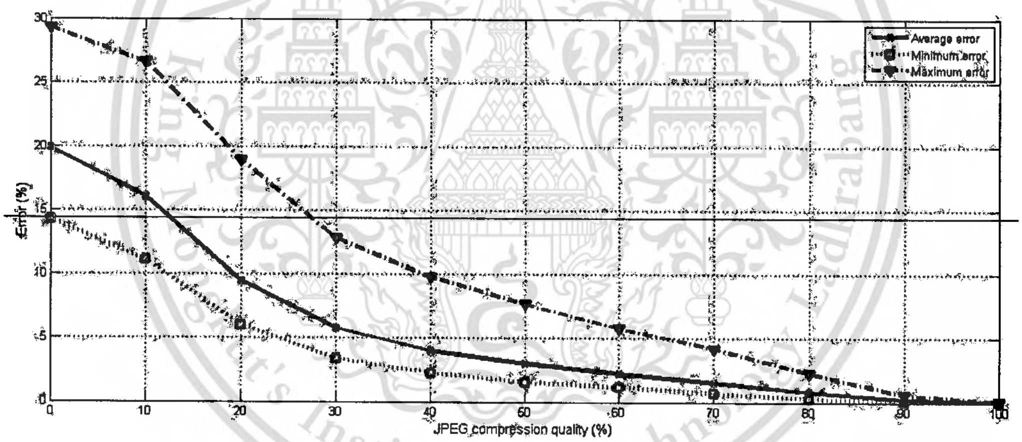


Figure 4.15 The error (%) level of vertical crossed-line watermark detection after compression attack.

Figs. 4.16 ~ 4.18 have shown the achievement of vertical line intersection with the cross ratio theory applying for strengthening the original-text verification performance against three manipulations; text adding, reordering and deleting. The results have shown that the vertical line intersection method with the cross ratio theory applying can sensitively detect even a few changes such as 1 - 10 text groups adding, reordering and deleting which found the maximum level of 0.32%, 0.60 % and 1.3 % of errors respectively.

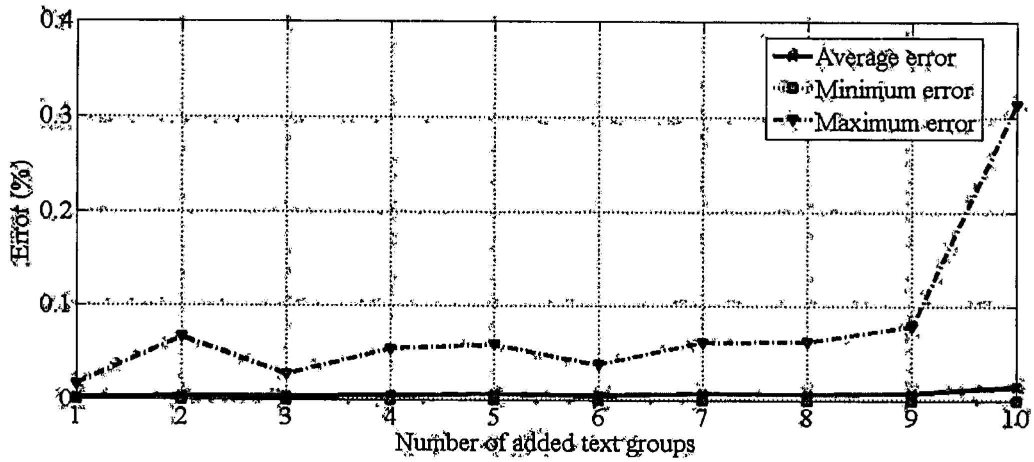


Figure 4.16 The error (%) level of vertical crossed-line watermark detection after text adding.

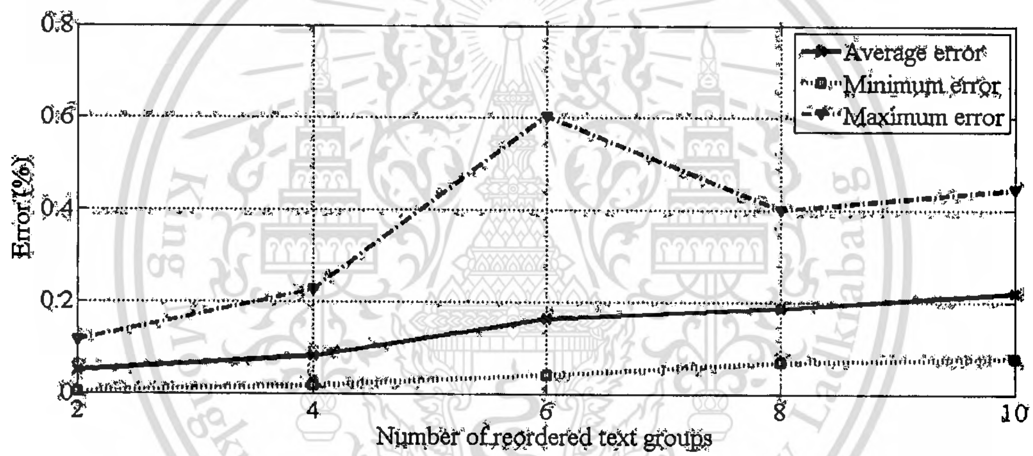


Figure 4.17 The error (%) level of vertical crossed-line watermark detection after text reordering.

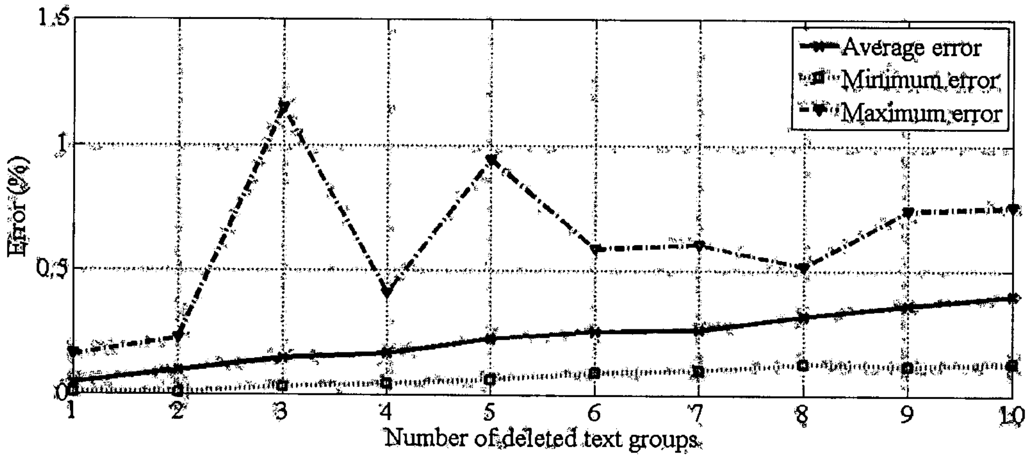


Figure 4.18 The error (%) level of vertical crossed-line watermark detection after text deleting.

4.4 Experimental Results of Method 4:

Applying The Double Reference-line Intersection of Vertical and Horizontal Lines and The Double Cross-ratio Method to Increase Maximum Watermark-hiding Capacity and Original-text Verification Precision of Text-image Watermark with Virtual Watermark Embedded

According to the Method 2 and Method 3, even each horizontal and vertical line would be fixed with the specific interval, but the intersected points run across the character skeleton lines and space could not be fixed, depending on text written on each page, so that we got the random intersection point pattern which is difficult to embed and detect watermarks. Moreover, it could not be sure that these crossed-line intersection points have covered throughout a text image page. Hence, we thought that the double or grid lines of horizontal and vertical lines would help us to solve this problem. Under this experiment, the intersection points of horizontal and vertical lines which exactly lied on the character skeleton lines or space would be counted as the marking point of the zero watermarks.

The 35 grayscale multi-language text images, size of 1240x1754 pixels and 150 dpi of resolution, still have been used in this experiment. Here, each of these grayscale text images was used to make both horizontal and vertical crossed-line intersected with text-character skeleton lines, so called the double reference-line intersection of vertical and horizontal lines, as described in Section 3.7 of Chapter 3, in order to exactly mark them at the main reference positions, where these three lines; horizontal, vertical and text-character skeleton lines crossed each other at the same points, for virtually watermark marking under directed by the cross ratio theory, here so called the double cross ratio method according to applying to both vertical and horizontal lines. One main intersection defined as one bit of virtual watermark data. The cross-ratio values used for virtual watermark marking and detecting for one 50 text-line page was approximately about 240,000 values.

The results of this experiment for digital marking on 35 grayscale text images of four tested languages; English, Thai, Chinese and Arabic, applied with 35 various virtual watermark patterns, through measuring of watermark values from the percentage of matching virtual watermark positions, one-to-one, between its original watermarked positions with the attacked one. This virtually-embedded watermark is detected if found the grayscale level of a character skeleton line, at the virtually-marked intersection position, is less than 245.

The first step is tested of the controlled virtually-watermarked text image without attacking by comparing with the plotted virtual-watermark positioning pattern, it got 100% of matching percentage or zero percentage of error.

The testing begins with the virtually embedding process which each reference horizontal and vertical crossed line is virtually run over each text-character skeleton lines of each text character of each language to mark the main intersected positions for virtually watermark marking. This embedding step revealed that this text character-two-line intersection watermarking technique, applied for both horizontal and vertical lines, can effectively create a real invisible watermark on any text images of any languages without any actual watermark really embedding, like working with individual horizontal and vertical lines.

Moreover, this experiment also shown that the reference horizontal and vertical line intersection on every text page of any language could generate invisible virtual watermark hiding-bit capacity up to 240,608 bits, including intersection points which either horizontal or vertical lines intersected character skeleton line or horizontal and vertical lines intersected each other on the blank area (see Fig. 4.14). This enormous capacity came from the watermarking program counted every intersection points, not depending on type of language, which all points could be used for virtual watermark bits embedding.

The geometric and image manipulation attacks have shown the remarkable achievement of the third objective, as specified in Section 1.2 of Chapter 1, with the effect of the cross ratio theory to strengthen the text-image watermarking robustness, under the double (H & V) crossed-line intersection, against some possible attacks such as compression, sharpness, noise signal adding and shearing (see Fig. 4.19). The results have shown that these attacks did not significantly affect the watermark detecting. They affected the virtually embedded watermark detecting just only 0 – 0.5 percent of error. This means the matching percentage of watermark detecting is mostly higher than 99.5 %.

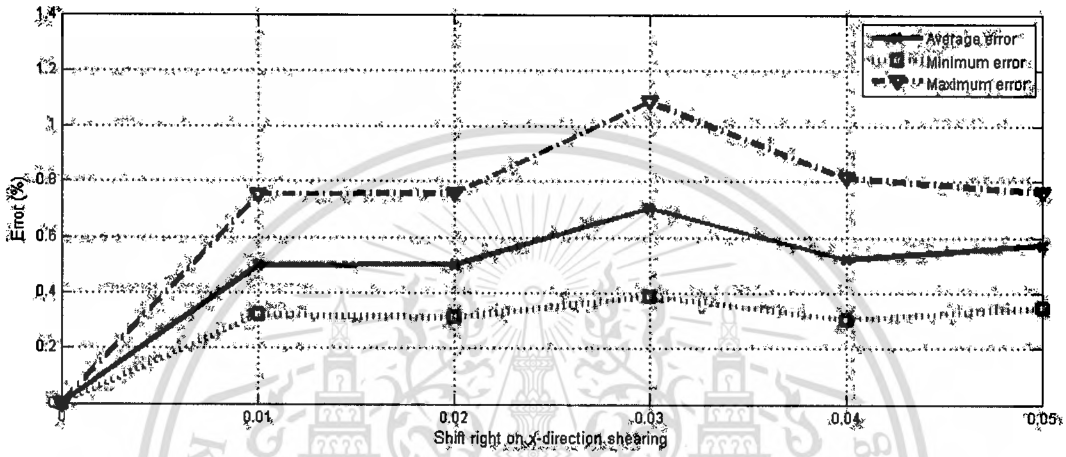


Figure 4.19 The error (%) level of double (H & V) crossed-line watermark detection after shearing attack.

Figs. 4.20 ~ 4.22 have still shown the achievement of horizontal and vertical line intersection with the cross ratio theory applying for strengthening the original-text verification performance against three manipulations; text adding, reordering and deleting. The results have shown that the horizontal and vertical line intersection method with the cross ratio theory applying can still sensitively detect even a few changes such as 1 - 10 text groups adding, reordering and deleting which found the maximum level of 0.28%, 0.38 % and 0.36 % of errors respectively.

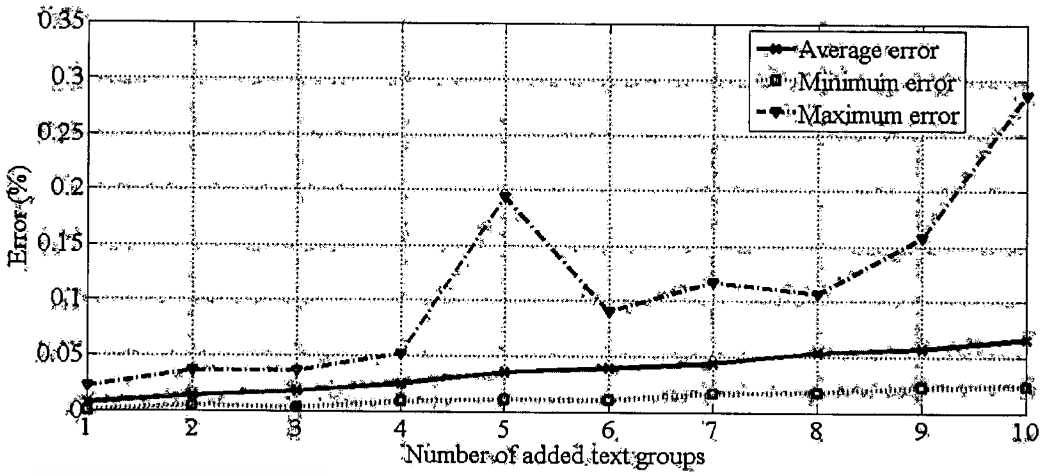


Figure 4.20 The error (%) level of double (H & V) crossed-line watermark detection after text adding.

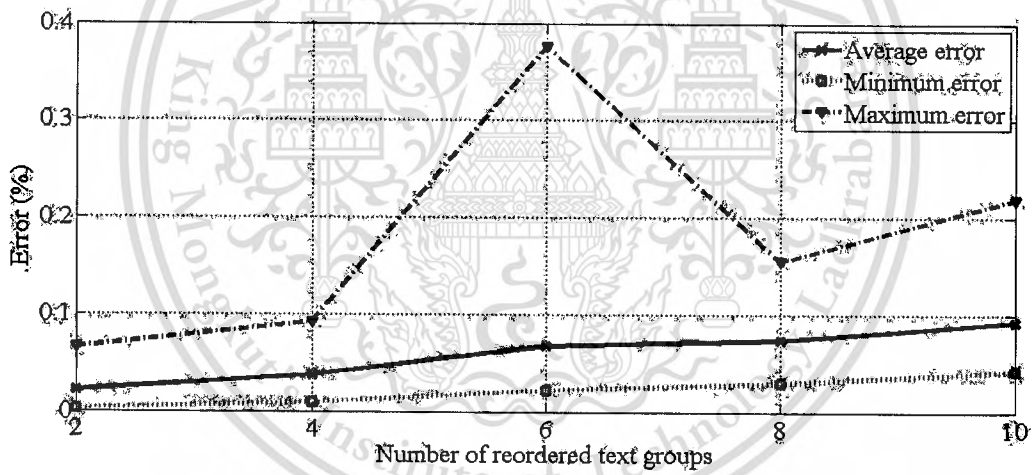


Figure 4.21 The error (%) level of double (H & V) crossed-line watermark detection after text reordering.

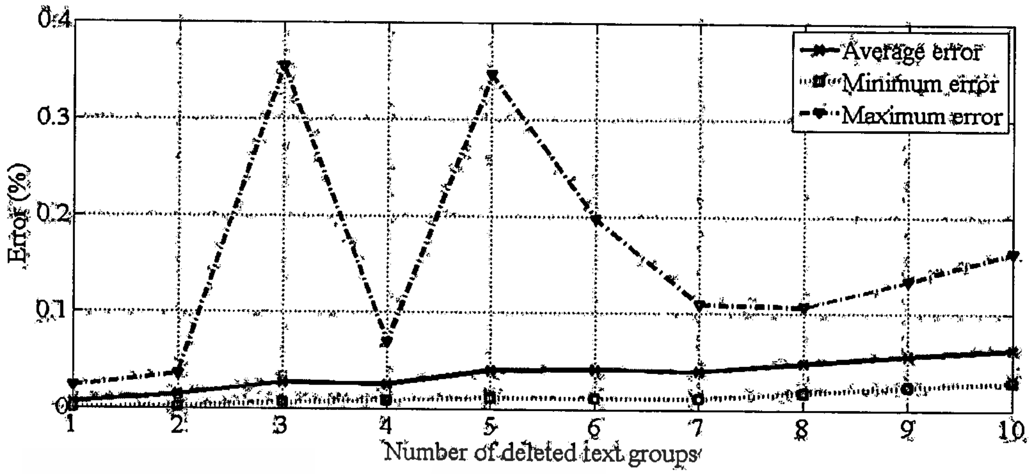


Figure 4.22 The error (%) level of double (H & V) crossed-line watermark detection after text deleting.

The above results shown that the amount of percent of errors detection were depending on the exact positions which selected for virtual watermark embedding or data collecting. For example, the intersected positions of double reference lines, vertical and horizontal lines, which run across or intersect each other on the blank area of a text image, would be used as a blank reference point for trapping any addition text from the text adding attack. That is why the double reference lines have more effective than the single reference line. However, in case of text reordering and deleting, the horizontal single reference-line intersection watermarking can do better than the others. For the hiding-bit capacity comparison, the double reference-line intersections is the best, with the enormous volume up to 240,608 bits (see Fig. 4.14 and Fig. 4.23), because all intersections of either vertical or horizontal lines, or both of them, with and without the character skeleton lines intersected, or intersected each other, were totally collected. This means it has the bit volume enough for storing the redundant virtual watermark bits and also has enough data to locate the text changing point on every area of a text image page.

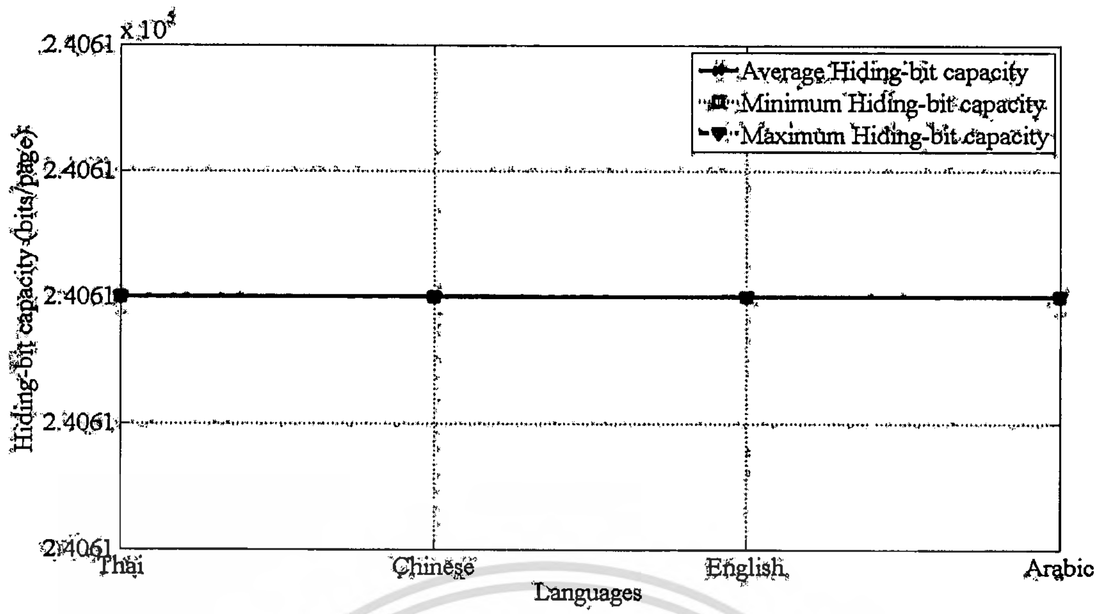


Figure 4.23 The hiding-bit capacity contributed by double (H & V) crossed line.

4.5 Experimental Results of Method 5:

Applying The Single Horizontal, Vertical and Double Reference-line Intersection and The Double Cross-ratio Method to Create The Multi-layer Watermarking for Increasing Maximum Watermark-hiding Capacity and Original-text Verification Precision of Text-image Watermark with Zero Watermark Embedded.

Lessons learned from the previous four methods, let us know a lot about strength and weak points of each crossed-line intersection watermarking method. Therefore, the Method 5 was planned to get rid of those weak points and combine their strength points of three crossed-line intersections; horizontal (H), vertical (V) and double (H & V) lines, together and then set up the pack of eleven crossed-line intersection watermarking layers which help us to overcome the main limitations of the grayscale text image watermarking, especially the hiding bit capacity and integrity verification.

4.5.1 The Tested Text Images of Multiple Languages

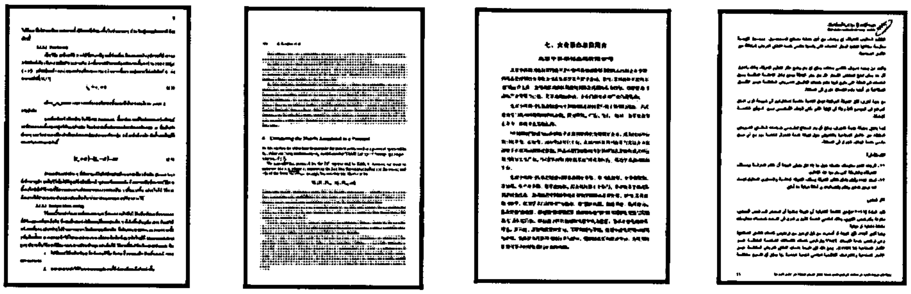
The set of 35 text images of grayscale multi-languages; Thai, English, Chinese and Arabic, size of 1240x1754 pixels and 150 dpi of resolution, had been tested in this experiment, as shown in Fig. 4.20.

4.5.2 The Number of Crossed-line Intersection Layers for Imperceptivity and Hiding-bit Capacity Testing

To get a remarkable volume of hiding-bit capacity for watermarking, at first, each page of these grayscale-text images had been virtually created their three main crossed-line intersection layers, consisting of main horizontal, vertical and double (H & V) crossed-line layers, as described in Section 3.8 of Chapter 3 and shown in Figs. 3.11 and 3.12 (a), (b) and (c). After that each layer had been derived into eight 2-pixel shifted crossed-line intersection layers, including two shifted horizontal crossed-line intersection layers (see Fig. 3.12 (a-1) ~ (a-2)), two shifted vertical crossed-line intersection layers (see Fig. 3.12 (b-1) – (b-2)) and four shifted double (H & V) crossed-line intersection layers (see Fig. 3.12 (c-1) – (c-4)). These are totally 11 crossed-line intersection layers for testing and contributing the watermark hiding-bit capacity.

4.5.3 The Tested Watermark Secret Data Embedding

Under this testing, the prominent crossed-line intersection points; the points where the main and shifted single horizontal, single vertical or double (H & V) line run across the text-character skeleton lines and the blank areas, created on each crossed-line intersection layer mentioned above, would be selected to embed the watermark secret data bits, one selected prominent intersection point for one embedded data bit with the cross ratio controlling. Here, the "COPYRIGHT" word would be used as the secret data bits for embedding on each crossed-line intersection layer, as shown in Figs. 3.15 and 3.16. One word of "COPYRIGHT" needs 72 bits; 0100001101001111010100 00001011 00101010010010010010100011101001000010101100, which could be calculated the number of redundant watermark-data blocks (*nWDB*) on each layer, given the bits spared for gap between each redundant block as 80 bits, by applying the Eq. (3.36).



(a) (b) (c) (d)

Figure 4.24 The examples of (a) Thai, (b) English, (c) Chinese and (d) Arabic text images, at resolution of 150 dpi.

4.5.4 The Geometric and Image Manipulation Attacking for Robustness Testing

Since a text image may be easily attacked with many possible ways. Thus, eleven watermarked crossed-line intersection layers have been tested with three projective-geometric distortions; shearing, scaling and rotating and five image manipulations; compression, sharpness, contrast, blur masking and noise signal adding, in order to check their robustness.

4.5.5 The Manipulating Attacks for Embedded Watermark Detection and Integrity Verification Testing

Three text-group manipulations, including 10 text groups adding, reordering and deleting manipulations, had been used to test the embedded watermark detection and integrity verification testing. The percentage of matching had been used as the measurement of this testing.

4.5.6 Results

The first testing result was came from the embedding and detection testing of the controlled watermarked text images without any manipulation attacking by comparing the plotted watermark positioning pattern and then extracting the embedded secret data; it got 100% of matching percentage or zero percentage of error. From the testing, we can briefly its result according to its objectives, as follow.

4.5.5.1 The Language Independent Testing Result

From testing, it has clearly shown that this multiple crossed-line intersection layer technique could be applied to all Thai, English, Chinese and Arabic text images. According to, under this technique, it needs only the crossed-line intersection points where the virtual reference lines run across text-character skeleton lines to embed watermark data bit without physically modifying its original text images and using any special technique for handling any specific language. However, each tested language has affected only the number of created intersection points due to its different character skeleton line structure.

4.5.5.2 The Imperceptivity Testing Result

According to these crossed lines are the virtual lines drawn only in the program process, not really drawn on each page of text images. Thus, nobody would absolutely observe them after these text images were completely zero watermark embedded, eventually, with a lot of secret data bits. On the other hand, they still exactly look like their original text images before watermarking, as shown in Fig. 4.24.

4.5.5.3 The Hiding-bit Capacity Testing Result

The testing revealed that the 7^{th}, 8^{th}, 9^{th}, 10^{th} and 11^{th} layers of the main and shifted double (H & V) line intersections could generate their maximum volume of hiding-bit capacity up to 240,608 bits/layer/A4 page of text image (see Fig. 4.25 and Table. 4.3) while the 1^{st}, 2^{nd}, 3^{rd}, 4^{th}, 5^{th} and 6^{th} layers of the main and shifted single horizontal and vertical line intersections could generate their maximum volume of hiding-bit capacity just only 77,334 bits/layer/A4 page of text image. This means the double (H & V) crossed-line layer, for one page of text image, could embed watermark up to 1,582 redundant data blocks/layer/A4 page of text image and the single horizontal and vertical crossed-line layers could embed redundant watermark data block just only 508 blocks/layer/A4 page of text image, according to Eq. (3.36) and one defined redundant block of 72 bits of COPYRIGHT secret data with given 80 bits of *Gbb*.

However, if considering for each language, this experiment has shown that the Arabic language has generated high hiding-bit capacity under both single horizontal and vertical crossed-line intersection layers (see Table 4.3). Anyway, these languages did not affect their hiding-bit capacities under the double (H & V) crossed-line intersection layer since all intersection points, either they run across text-character skeleton lines or blank areas, have been counted.

From the above results, they could be concluded all hiding-bit capacities generated by eleven main and shifted crossed-line intersection layers as shown in Table 4.3. To count the total of hiding-bit capacity per page of text image under this technique, we would calculate by totally counting all hiding-bit capacities of all layers as concluded in Table 4.3. Here, you can see the Arabic language could totally generate the maximum hiding-bit capacity per page up to 1,545,738 bits/A4 page of text image or can be embedded the redundant watermark secret data, COPYRIGHT, up to 10,169 blocks for one A4 page of text image by which applying Eq. (3.36), with given 80 bits for gap of each block.

4.5.5.4 The Robustness Testing Result

Figs. 4.25 ~ 4.27 and Table 4.4, revealed that the 7^{th}, 8^{th}, 9^{th}, 10^{th} and 11^{th} layers of the main and shifted double (H & V) line intersections did very well, up to 99 percent of watermark bit detection, under the geometric attacks; rotation, shearing and scaling, and some image manipulation attack such as compression, contrasting, sharpness and noise (salt & pepper) signal adding, excepted only blur attacking.

Table 4.4 has concluded the watermark matching percentage comparison of the embedded watermark based on the layers of main and shifted single horizontal (H) and vertical (V) lines and double (H &V) lines intersections after geometric and image manipulation attacks, such as rotation, shearing, scaling, contrasting, compression, sharpness, blur and noise signal adding. Here, we can conclude that the double (H & V) lines are strongly contributing to the zero watermark robustness.

Table 4.3 The hiding-bit capacity of each layer of each language.

Layer No.	**Type of Crossed-line Intersection**	Hiding-bit Capacity (bits)			
		Thai	English	Chinese	Arabic
1st	Main Horizontal (H)	64,278	65,125	61,614	75,329
2nd	Lower-shifted H.	67,244	64,327	61,541	75,802
3rd	Upper-shifted H.	61,829	64,457	61,893	77,334
4th	Main Vertical (V)	32,234	32,449	30,829	38,053
5th	Left-shifted V.	32,188	32,204	30,978	38,048
6th	Right-shifted V.	32,253	32,302	30,717	38,132
7th	Main Double (H & V)	240,608	240,608	240,608	240,608
8th	Upper-left Shifted Double (H & V)	240,608	240,608	240,608	240,608
9th	Upper-right Shifted Double (H & V)	240,608	240,608	240,608	240,608
10th	Lower-left Shifted Double (H & V)	240,608	240,608	240,608	240,608
11th	Lower-right Shifted Double (H & V)	240,608	240,608	240,608	240,608
Total		**1,493,066**	**1,493,904**	**1,480,612**	**1,545,738**

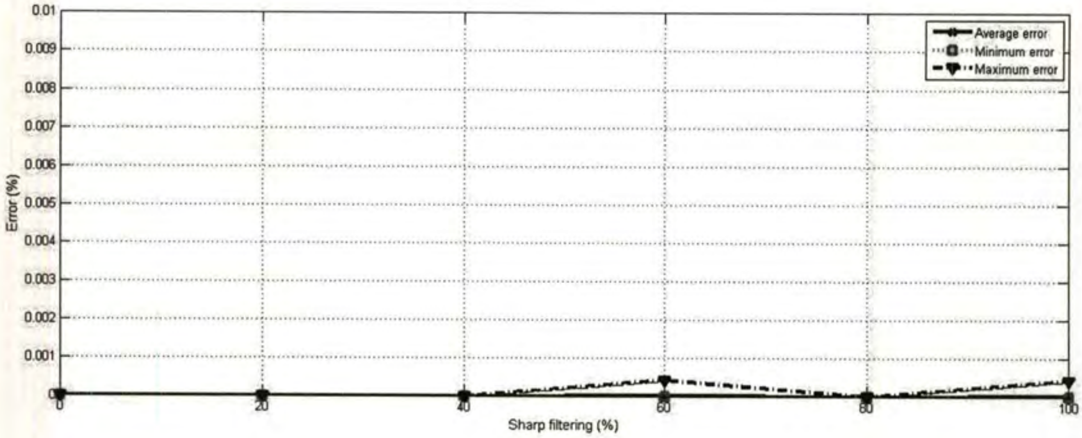


Figure 4.25 The error (%) level of double (H & V) crossed-line watermark detection after sharpness attack.

Figure 4.26 The error (%) level of double (H & V) crossed-line watermark detection after compression attack.

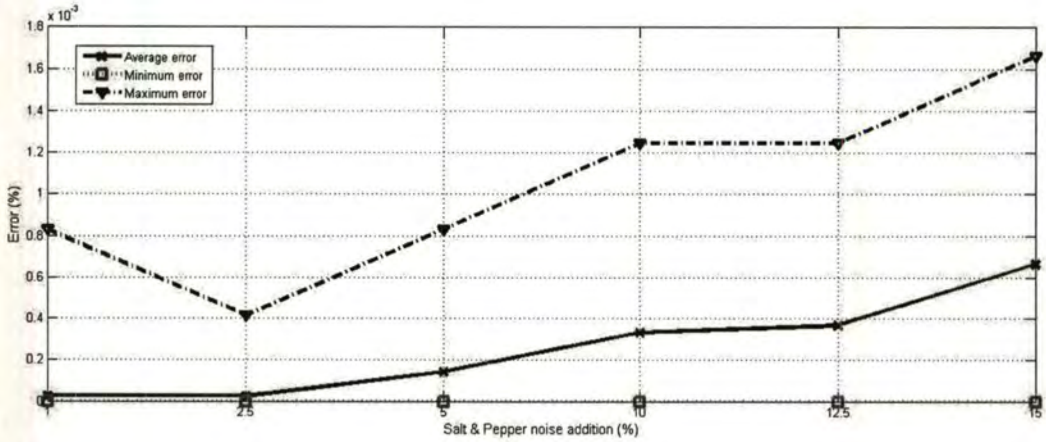


Figure 4.27 The error (%) level of double (H & V) crossed-line watermark detection after adding the salt & pepper noises.

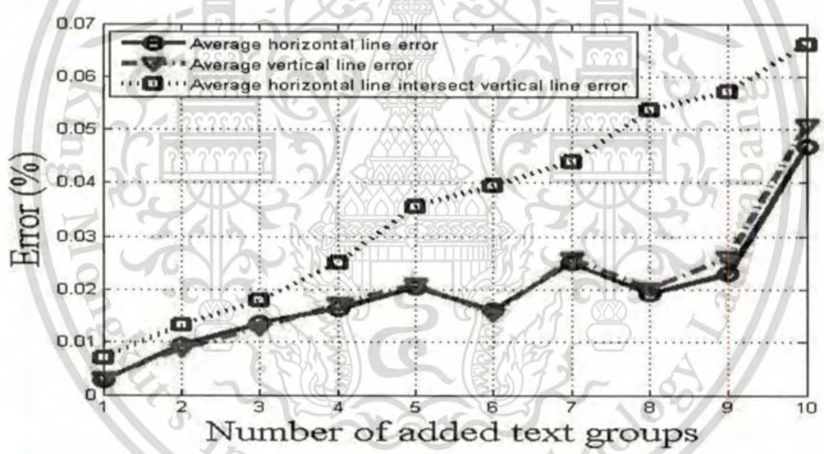


Figure 4.28 The error (%) level of main double (H & V) crossed-line watermark detection after text adding.

Table 4.4 Robustness performance of eleven crossed-line layers

Layer No.	Type of Crossed-line Intersection	Embedded bits	% Bit detected after geometric attacking			% Bit detected after image manipulation attacking					Average % detected/layer
			Rotation (5 – 35°)	Shearing (Shift 0 – 0.05)	Scaling (10 – 120%)	Contrasting (10 – 50%)	Compression (0 – 100%)	Sharpness (0 – 100%)	Blur (3x3 – 15x15)	Noise adding (S&P 0 – 15%)	
1	Main Horizontal (H)	100.00	85.13	92.92	84.89	99.62	94.05	100.00	14.14	99.85	83.83
2	Lower-shifted H.	100.00	85.17	92.93	84.92	99.65	94.10	100.00	13.98	99.84	83.82
3	Upper-shifted H.	100.00	85.14	92.91	84.89	99.63	94.09	100.00	14.10	99.84	83.83
4	Main Vertical (V)	100.00	85.68	89.49	85.20	99.61	94.27	100.00	18.86	99.85	84.12
5	Left-shifted V.	100.00	85.69	89.46	85.20	99.58	94.18	100.00	18.83	99.85	84.10
6	Right-shifted V.	100.00	85.65	89.48	85.23	99.61	94.24	100.00	18.77	99.85	84.10
7	Main Double (H & V)	100.00	99.50	99.53	99.31	99.95	99.95	100.00	93.02	99.74	98.88
8	Upper-left Shifted (H&V)	100.00	99.99	99.98	99.99	100.00	99.95	100.00	92.91	99.99	99.10
9	Upper-right Shifted (H&V)	100.00	99.99	99.98	99.99	100.00	99.95	99.99	92.91	99.99	99.10
10	Lower-left Shifted (H & V)	100.00	99.99	99.98	99.99	99.99	99.95	99.99	92.96	99.99	99.11
11	Lower-right Shifted (H&V)	100.00	99.99	99.98	99.99	100.00	99.95	100.00	92.95	99.99	99.11
Total Embedded/Detected Bits of 11 Layers		1,100.00	1,011.92	1,046.64	1,009.60	1,097.64	1,064.68	1,099.98	**563.43**	**1,098.78**	
Robustness Performance Percentage of 11 Layers			**91.99**	**95.15**	**91.78**	**99.79**	**96.79**	**100.00**	**51.22**	**99.89**	

4.5.5.5 The Copyright Ownership and Integrity Verification Testing Result

After three text manipulations testing by 10 text groups adding, reordering and deleting, it found that each layer could be detected and extracted the embedded watermark and verified its integrity or modification, measured in percent of error, as shown in Table 4.5 and Figs. 4.28 ~ 4.33.

Table 4.5 The performance of crossed-line layers in term of integrity verification.

Layer No.	Type of Crossed Line	Avg. % of Detection After Attacking with 10 Text-Group Manipulations		
		Adding	Reordering	Deleting
1st	Main Horizontal (H)	99.95	99.25	98.63
2nd	Lower-shifted H.	99.95	99.25	98.65
3rd	Upper-shifted H.	99.95	99.25	98.63
4th	Main Vertical (V)	99.95	99.23	98.63
5th	Left-shifted V.	99.95	99.26	98.64
6th	Right-shifted V.	99.95	99.25	98.65
7th	Main Double (H & V)	99.93	99.91	99.94
8th	Upper-left Shifted Double (H & V)	99.93	99.91	99.94
9th	Upper-right Shifted Double (H & V)	99.93	99.91	99.94
10th	Lower-left Shifted Double (H & V)	99.93	99.91	99.94
11th	Lower-right Shifted Double (H & V)	99.93	99.91	99.94

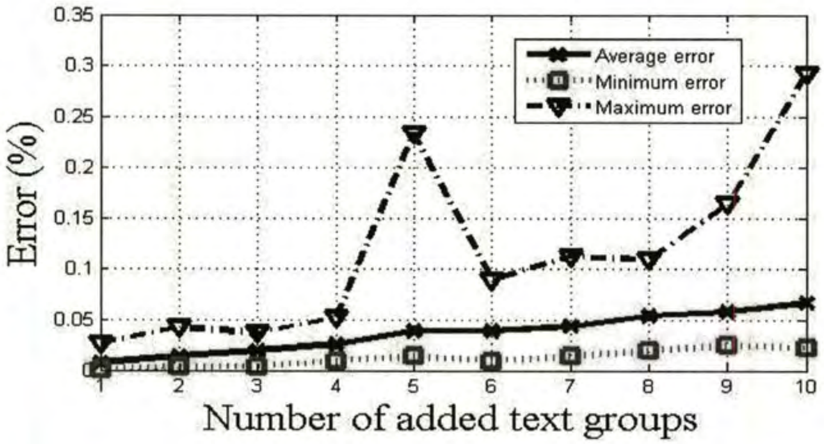


Figure 4.29 The error (%) level of lower-left shifted double (H & V) crossed-line watermark detection after text adding.

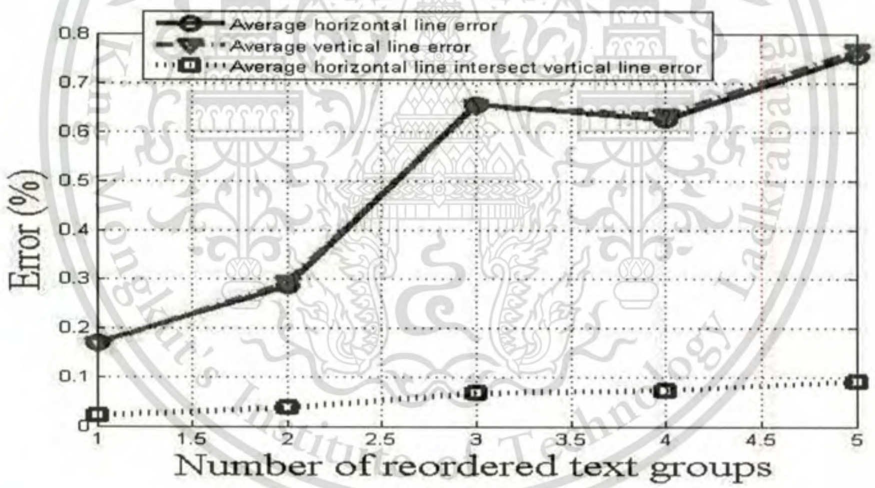


Figure 4.30 The error (%) level of main horizontal crossed-line watermark detection after text reordering.

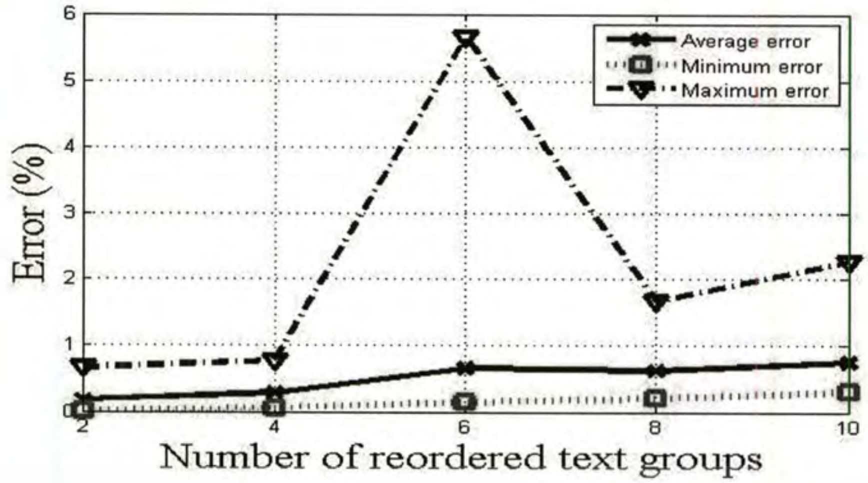


Figure 4.31 The error (%) level of lower shifted horizontal crossed-line watermark detection after text reordering.

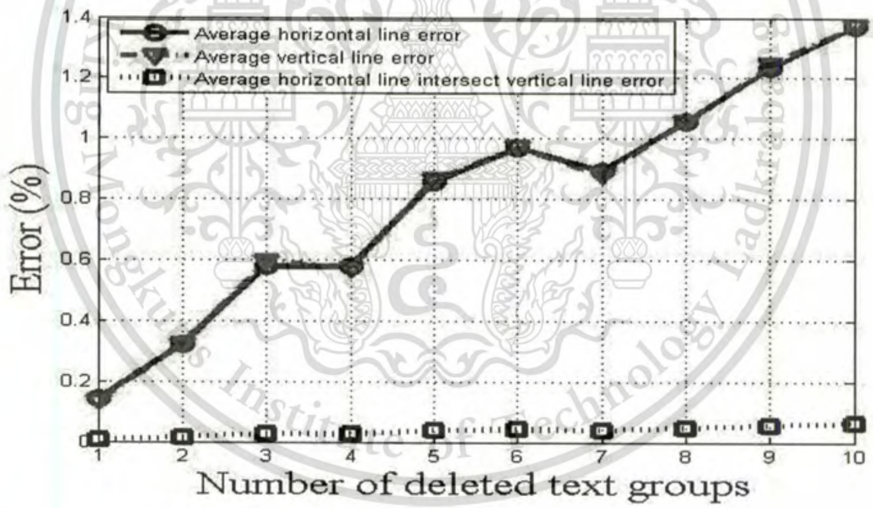


Figure 4.32 The error (%) level of main horizontal crossed-line watermark detection after text deleting.

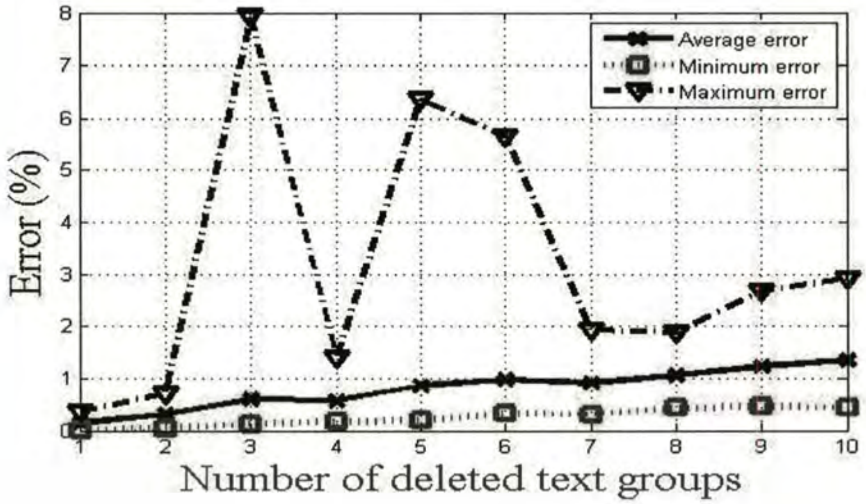


Figure 4.33 The error (%) level of lower shifted horizontal crossed-line watermark detection after text deleting.

The results shown that the main and shifted horizontal and vertical crossed-line intersection layers are suitable for using to detect the embedded watermark data; COPYRIGHT, and identify its integrity or the tested changes under the 10 text-group reordering and deleting manipulation attacks which detected the percent of errors up to 0.76% and 1.37% or 99.24% and 98.63% of matching respectively (see Figs. 4.30 ~ 4.33 and Table 4.5). While, the main and shifted double (H & V) crossed-line intersection layers are better for detecting the embedded watermark data and identifying its integrity or the tested changes under the 10 text-group adding manipulation attack at 0.07% of errors or 99.93% of matching (see Figs. 4.28 ~ 4.29 and Table 4.5). According to the results concluded in Table 4.5, it revealed that each shifted line of main horizontal, vertical and double (H & V) crossed lines could contribute the integrity verification in the same range of error percentage as its main crossed line. This due to 2-pixel shifting range had mostly still run across the same character skeleton line.

CHAPTER 5

CONCLUSION AND RECOMMENDATION

5.1 Conclusion

5.1.1 Method 1: Applying The Actual Watermark Embedded Between Text Lines and The Cross Ratio Theory to Build up The Robustness of Text-image Watermark

In the first experiment, correlation coefficient measurement, acceptable values between 0.5 – 1, which has been used for detecting the invisible grayscale actual watermark existing on the English, Thai, Chinese and Arabic text image files, has shown that the cross-ratio theory applying could be effectively used to build up the reasonably watermarking robustness against the projective-geometric distortion attacks; scaling, especially at the range higher than 11% (see Table 5.1), shearing (0 – 0.05) and rotating (1 – 4 degrees) and some manipulating attacks; compressing, at the range higher than 60%, contrasting (1 – 45%), sharpness (0 – 100%) and blur filtering which mask size should not be greater than 13x13. This built-up robustness is based on four collinear points which have been used as the watermark embedding patterns and the referred points for watermark detection. It is not necessarily to be inversely transformed before detecting watermark positions, but can be directly detecting watermark position at once, and it is not necessarily compared with original text image without watermark. Confirmation of our document from watermark detecting can be proved directly through comparison of the existed watermark pattern.

The experiment has also shown that it can be applied for all multi-language text images, does not stick to one specific language attributes like some methods mentioned above which mostly focused on testing only one specific language and not thoroughly explored the possible attacks which affect the watermark robustness. This is the original step of applying the cross ratio theory for grayscale multi-language text image watermarking.

Table 5.1 Performance comparison of our proposed watermarking techniques

Technique	Applied to	Imperceptibility	Language	Approximately Hiding Bit Capacity	Robustness	Integrity Verification?
Method 1: Applying the actual watermark embedded between text lines	Text Image	Invisible	English, Thai, Chinese, Arabic	12,500 bits/A4 page	Sharpness (0-100 %), JPEG compression (60-100 %), Blur(3x3-13x13 of mask size), Contrasting(1-45 %), Shearing(0 - 0.05 %), Rotation(0 - 4 degree), Noise signal adding (Salt & Pepper 0-1.5 %), Scaling(11 - 60 %), Brightness(5 %), Shearing(0 - 0.05)	No
Method 2: Applying the single reference-line intersection of horizontal line	Text Image	Invisible	English, Thai, Chinese, Arabic	Up to 75,329 bits/A4 page	Sharpness (0 – 100%), JPEG compression (0 – 100%), Contrasting (10 – 50%), Shearing (Shift 0.00 – 0.05), Rotation (5-35 degree), Noise signal adding (Salt & Pepper: 1 – 15%), Scaling (10 - 120%), 1 – 10 text group adding , deleting and reordering	Yes
Method 3: Applying the single reference-line intersection of additional vertical line	Text Image	Invisible	English, Thai, Chinese, Arabic	Up to 38,053 bits/A4 page	Sharpness (0 – 100%), JPEG compression (40 – 100%), Contrasting (10 – 50%), Shearing (Shift 0.00 – 0.05), Noise signal adding: (Salt & Pepper: 1 – 15%), Scaling (10 - 120%), 1 – 10 text group adding , deleting and reordering	Yes
Method 4: Applying the double reference-line intersection of vertical and horizontal lines	Text Image	Invisible	English, Thai, Chinese, Arabic	Up to 240,608 bits/A4 page	Sharpness (0 – 100%), JPEG compression (0 – 100%), Blur (3x3 - 5x5 of mask size), Contrasting (10 – 50%), Shearing (Shift 0.00 – 0.05), Rotation (5-35 degree), Noise signal adding (Salt & Pepper: 1 – 15%), Scaling (10 – 120%), 1 – 10 text group adding , deleting and reordering	Yes
Method 5: Applying the single horizontal, vertical and double reference-line intersection and the double cross-ratio method to create the multi-layer watermarking.	Text Image	Invisible	English, Thai, Chinese, Arabic	Up to 1,545,738 bits/ A4 page (11 layers)	Sharpness (0 – 100%), JPEG compression (0 – 100%), Blur (3x3 - 5x5 of mask size), Contrasting (10 – 50%), Shearing (Shift 0.00 – 0.05), Rotation (5-35 degree), Noise signal adding: (Salt & Pepper 1 – 15%), Scaling (10 – 120%), 1 – 10 text group adding , deleting and reordering	Yes

5.1.2 Method 2: Applying The Single Reference-Line Intersection of Horizontal Line and The Cross Ratio Theory to Increase The Hiding-bit Capacity and Original-Text Verification Performance of Text-image Watermark with Zero Watermark Embedded

The second experiment has disclosed that the more number of horizontal crossed lines intersected each text character of each text line on each page of a text image, the more hiding-bit capacities and marking positions of zero watermarks increasing. This means more rooms for hiding redundant confidential data of watermark on each crossed line, especially for languages that have a lot of characters per line and have many lines per page such as Thai and English languages would create more line intersections, thus get more capacity for hiding bit of watermark. However, the experiment has shown that many intersection points could not be used for virtually watermarking due to their difficulties of detection. Therefore, only some prominent intersection points have been selected for marking watermarks.

This experiment has shown that its robustness can be enhanced with the cross ratio theory of four collinear points applying. This robustness came from mathematical mechanism of cross ratio theory which able to effectively refer to its original zero watermark embedded positions even it has been attacked with some geometric distortions such as rotation and scaling.

However, this experiment also shown that some error could occur in case of small font size and small gap between each text line which sometime made some of five horizontal reference intersection lines overlapped or located nearly in the same level. These results revealed that the suitable gap between each horizontal reference crossed line should not less than three pixels, otherwise its intersection points, for zero watermark embedding, would be gone after attacking with the above mentioned geometric distortions (see Table 4.4 and Table 5.1). This bad effect is caused by small interval between each line which difficult to detect these zero watermarks after distortion attacking.

The small interval affects to the detection of its integrity or modification of its original text image as well, except only drawing the horizontal reference lines under suitable interval, such as more than three pixels, and directing watermarking plotting with the cross ratio theory. This optimum horizontal line interval would reasonably strengthen the original-text verification performance against some possible manipulations such as text adding, reordering and deleting. The results have shown that the horizontal-line intersection method can sensitively detect even a few changes of text reordering and deleting. This detection can eventually pinpoint to the exact position of text changing on each page of each text image.

5.1.3 Method 3: Applying The Single Reference-line Intersection of Additional Vertical Line and The Cross-Ratio Theory to Increase More Watermark-hiding Capacity and Original-text Verification Performance of Text-image Watermark with Zero Watermark Embedded

The experiment clearly shown that a real invisible watermark on any text image of any language without any actual watermark really embedding can be effectively created with the text character-vertical line intersection watermarking technique. It also recovered that the reference vertical-line intersection on the Arabic text page could generate a big volume of invisible zero watermark hiding-bit capacity up to 38,000 bits (see Table 4.3 and Table 5.1), lower than the hiding-bit capacity of the horizontal. This capacity came from the Arabic characters skeleton itself which mostly have many longer horizontal skeleton lines than other languages, so that has more chance to frequently intersect with the vertical reference line.

However, the vertical line intersection with the cross ratio theory applying does not outstandingly build up the original-text verification performance against three manipulations; text adding, reordering and deleting, comparing with the horizontal line which do better. This subordinate is caused by the interval space between each character and the character skeleton line structure, from vertically view angle, which almost the same so that left or right character reordering may be looked like it stands still in the same intersected position if it is replaced by the character which has similar character skeleton line.

Thus, it should be crosschecked with other single horizontal reference lines and/or double, vertical and horizontal intersected, reference lines, in order to pinpoint to specific position of text has been tampered with. The geometric and other attacking test results (see Table 4.4 and Table 5.1) revealed that vertical crossed-line robustness has the same strength and weakness as the horizontal crossed line.

5.1.4 Method 4: Applying The Double Reference-line Intersection of Vertical and Horizontal Lines and The Double Cross-ratio Method to Increase Maximum Watermark-Hiding Capacity and Original-text Verification Precision of Text-image Watermark with Zero Watermark Embedded

This experiment, with double (H & V) reference-line-intersection watermarking technique, can effectively create a real invisible watermark on any text image of any language without any actual watermark really embedding. Moreover, this experiment also shown that the reference horizontal and vertical line intersection on every text page of any language could generate invisible zero watermark hiding-bit capacity up to 240,000 bits (see Table 4.3 and Table 5.1). This bit count is including all intersection points, both with and without the character skeleton line intersected the intersection point of the horizontal and vertical lines.

It found that the horizontal and vertical line intersection with the cross ratio theory applying was very effectively robust for surviving from geometric and other image attacking. In addition, it worked very well for verifying the new text adding (see Table 4.4 and Table 5.1). Because the intersected positions of this double (H & V) reference lines have run across or intersect each other on the blank area of a text image which would be used as a reference point of blank area and for trapping any addition text from the text adding attack and also precisely pinpointing to the exact position of text adding.

5.1.5 Method 5: Applying The Single Horizontal, Vertical and Double Reference-Line Intersection and The Double Cross-Ratio Method to Create The Multi-layer Watermarking for Increasing Maximum Watermark-hiding Capacity and Original-text Verification Precision of Text-image Watermark with Zero Watermark Embedded.

The above results have shown that the multiple layers of crossed-line intersections with the cross ratio theory applying could effectively increase the hiding-bit capacity and simultaneously improve other watermarking performances as well (see Table 5.1). Belows are the technical discussions of these achievements.

5.1.5.1 Language Independent Achievement due to The Crossed-line Intersections

This achievement came from the applying of crossed-line intersection which used the horizontal, vertical and double (H & V) lines run across the text-character skeleton lines and blank areas on each page of text image and then created the intersection points for embedding watermarks. However, even though this experiment had been tested with only four languages; Thai, English, Chinese and Arabic, but it is able to absolutely presume that it can apply to other languages as well since this new technique does not need to modify any text alignment, syntactic or semantic structure. According to every language has its own character skeleton lines so that these skeleton lines could be created the intersection points whereever the reference lines virtually run across them, even the blank areas. Thus, it's not depending on any type of language. However, the number of text-character skeleton lines of each language would affect the number of intersection points. That means the more text-character skeleton lines, per one character, the more intersection points would be created.

5.1.5.2 Watermark Imperceptivity Achievement due to The Virtually Crossed Lines

As mentioned above, since this watermarking technique is based on the intersection layers of the virtual horizontal, vertical and double (H & V) reference lines vitrually run across text-character skeleton lines and blank areas and then create the intersection points for embedding watermark secret-data bits, one point for one bit, without actually modifying any text alignment, syntactic or semantic structure of any character and text image, so called zero watermark, therefore it would not leave any physical clue to be seen or be easily traced them back thereafter. Thus, all embedded watermark could not totally be seen or observed by anyone.

5.1.5.3 Remarkably Hiding-bit Capacity Creating Achievement due to The Multiple Layers of Crossed-line Intersections

Under this new technique, the hiding-bit capacity would depend on the number of intersection points which horizontal, vertical and double (H & V) reference lines, text-character skeleton lines and blank areas on each layer virtually created. One created intersection point means one bit capacity to embed one secret data bit of watermark. However, under the practical way, we do not use every intersection points but select only the main prominent points by screening them with the grayscale level which must lower than 245, for bit "one", or equal and higher than 245, for bit "zero". This selection would help us to still easily detect them after facing some possible attacks such as word adding, reordering and deleting. From this experiment, you would see that, for one given language, the hiding-bit capacity would be varied according to number of virtual crossed lines created on each layer and, of course, also be varied according to the number of crossed-line intersection layers that you could create on each page of text images. However, we have found that number of crossed lines would be limited if they have gaps or interval space between each line less than three pixels. Actually, the cause of this limitation is the accuracy problem of grayscale measurement which you have to give them some space or tolerance for clearly identifying the adjacent two points. Moreover, the number of intersection points would be also directly varied according to the drawing pattern of each crossed line such as the upper-shifted horizontal crossed lines could generate the hiding bits up to 77,334 bits/layer/A4 page of text image, while the right-shifted vertical crossed lines could generate only 38,132 bits/layer/A4 page of text image, as shown in Table 4.3. However, these both single lines have still generated just small level scale of hiding-bit capacity since one single line run across a blank area would be counted just only one bit, while comparing with the double lines, generated up to 240,608 bits/layer/A4 page of text image, would be counted all H & V intersection points which run across all blank areas. Totally, eleven layers can generate hiding-bit capacity up to 1,545,738 bits/A4 page of text image. Under this technique, you can also create more intersection points, for more hiding-bit capacity, by shifting these main crossed lines, for example plus more two pixels from each main crossed-line origin to new positions in the left and right or up and down directions, in the condition that, on each layer, these new created intersection points must not be located on the same position of the intersection points of main crossed lines. In addition, the hiding-bit capacity would be varied according to the type of language because it could affect the number of text characters contained on one page. This means the more complication of text-character skeleton lines, the more intersection points would be created on each layer. That is why, from this testing, the Arabic language could generate the highest hiding-bit capacity, as shown in Table 4.3.

5.1.5.4 The Embedded Watermark Surviving from Geometric and Other Attacks due to The Cross Ratio Bonded All Crossed-line Intersection Points Together.

Table 4.4 and Table 5.2 have concluded that the main and shifted double (H & V) crossed-line layers did very well about the robustness against the geometric attacking; rotation, shearing and scaling, and the other image manipulations such as compression, contrasting, sharpness, blur and noise signal adding. This is because the intersection points of the main and shifted double (H & V) crossed lines had been fixed with the same interval and distance throughout their layers and these intersection points had been mapped to the specific points of the blank area and character skeleton lines. Thus, these lines act like the grid lines which easily predefined with the fixed cross ratio values for embedding zero watermarks at any specific intersection points and vice versa, it's also easy to trace and detect these values back after they had been attacked.

5.1.5.5 Integrity/Change Verification Capability Achievement due to All Crossed-line Intersection Points of All Layers Collected

Another significant achievement of crossed-line intersection technique is the capability of integrity verification or change detection. As mentioned above, even though you have already identified your ownership copyright watermark data from text images, it does not mean they are the real or original ones. Therefore, it is nesscessary for us to verify their integrities as well. From this experiment, as results shown in Table 4.5, you would see that the finely and totally collected crossed-line intersection points could simultanneously detect some changes or modifications of their original text images as well. Here, the key success factors are creating many intersection points throughout each layer of each text-image page as much as you can and then collect all identified bits of these intersection points. Thus, any change or modification such as some words added, reordered or deleted, they would be easily detected. However, this capability would depend on the type of crossed-line intersection which the experiment showed that the double (H & V) line intersection was very effective for verifying the new text adding (see Fig. 4.29). Because the intersected positions of this double reference lines, vertical and horizontal lines, have run across or intersect one another on the blank area of a text image which would be used as a reference point of blank area and for trapping any additional text from the text adding attack and also precisely pinpointing to the exact position of the new text adding. While, for the text reordering and deleting, the single horizontal reference-line would do better than the single vertical reference line and double crossed lines. This advantage performance is very usefull for approving the originality and integrity of sensitive paper images such as cheque, land license certificate and significant trade contract documents.

Thus, we can conclude that the multiple crossed-line intersection layer technique is the new breakthrough of text-image watermarking. It can overcome many limitations of other text-image watermarking techniques such as the limitations of hiding-bit capacity, imperceptivity and language dependency which are the main weak points of many existing watermarking techniques (see Table 5.3). In addition, it also give us more performance of integrity verification which other techniques never done before. Actually, with the cross ratio theory of four collinear points applied for controlling their embedded positions, this crossed-line intersection technique can also presume that it will enhance the robustness of the embedded watermarks for being well detected after they have been attacked with some geometric distortions such as shearing or rotating as well. This is due to this theory is based on the projective geometric invariance so that it can effectively to recalculate back to all original watermark embedded positions.

Even though our proposed zero watermaking technique has overcome the main difficult limitations of text image watermarking, but it still has some weak points and limitaions to effectively use it in the real world. Table 5.4 shows some disadvantages and limitations of our zero watermarking technique for whom want to apply it should be aware, such as a lot of computer resources and time consuming when apply it to many pages of text image, such more than 25 pages.

However, for effectively using, it might not be necessary to apply all eleven layers, especially whenever apply it with many pages of general document images. Because, from the Table 4.4, it has shown us that the five layers of the main and shifted double (H & V) crossed lines, including 7th, 8th, 9th, 10th and 11th layer, are the most effective layers and reliable up to 99 percent under some specific geometric attacks.

Table 5.2 Performance conclusion of our watermarking technique

Principle	Applied to	Imperceptibility	Language	Approximately Hiding Bit Capacity	Watermark Matching Percentage/ Robustness	Integrity Verification?
Applying the single horizontal, vertical and double reference-line intersection and the double cross-ratio method to create the multi-layer watermarking for increasing maximum watermark-hiding capacity and original-text verification precision of text-image watermark with zero watermark embedded.	Text image	Invisible	English, Thai, Chinese, Arabic	Up to 1,545,738 bits/ A4 page (11 layers)	-Sharpness (0 – 100%): 100% (H or V) -JPEG compression (0 – 100%): 99.53 – 100% (HV) -Blur (3x3 - 5x5 of mask size): 97.0 – 99.5% (HV) -Contrasting (10 – 50%): 99.59 – 100% (H, V or HV) -Shearing (Shift 0.00 – 0.05): 99.3 – 100% (HV) -Rotation (5-35 degree): 99.14 – 99.85% (HV) -Noise signal adding (Salt & Pepper 1 – 15%): 99.73 – 99.97% (H or V) -Scaling (10 – 120%): 97.95 – 100% (HV) -Text adding: 99.33% -Text deleting: 98.63% -Text reordering: 99.24% -Hiding-bit capacity: (bits /A4 pages) - Upper-shifted horizontal(H) line-Arabic: 77,334 - Right-shifted vertical(V) line- Arabic: 38,132 - Main double(H & V) line- All languages: 240,608	Yes

Table 5.3 Performance comparison of our approach with existing watermarking techniques

Technique	Applied to	Imperceptibility	Language	Approximately Hiding Bit Capacity	Robustness	Integrity Verification?
Our approach : Applying the single horizontal, vertical and double reference-line intersection and the double cross-ratio method to create the multi-layer watermarking.	Text Image	Invisible	English, Thai, Chinese, Arabic	Up to 1,545,738 bits/ A4 page (11 layers)	Sharpness (0 – 100%), JPEG compression (0 – 100%), Blur (3x3 - 5x5 of mask size), Contrasting (10 – 50%), Shearing (Shift 0.00 – 0.05), Rotation (5-35 degree), Noise signal adding: (Salt & Pepper 1 – 15%), Scaling (10 – 120%), 1 – 10 text group adding , deleting and reordering	Yes
Changing the low-4 bits of RGB color components of characters [20]	Digital plain text	Invisible	Chinese	40,000 bits/A4 page	Robustness to resist deletion, modification attack etc.	No
Reducing or increasing length of letter for feature coding[15]	Text image	Visible	English	4,000 bits/A4 page	Low robust to document passing through document processing or OCR processing	No
Character peak point distinction [18]	Text image	Visible	Persian and Arabic	900 bits/A4 page	Low robust to brightness and noise signal adding	No
Inter-word space shifting (to left or right) [15], [16]	Digital Text document, Text image	Visible	Multi-language	720 bits/ A4 page	Easy to observe and re-manipulate or low robust to document passing through document processing or OCR	No
Adjusting character size [17]	Digital Text Document, Text image	Visible	Chinese	650 bits/A4 page	Low robust to document passing through document processing or OCR processing	No
Point shifting of letter "i" and "j" [18], [19]	Text image	Visible	English	600 bits/A4 page	Low robust to brightness and noise signal adding	No
Text line shifting [12] ,-[13], [14]	Digital Text document, Text image	Visible	Multi-language	30 bits/A4 page	Easy to observe and re-manipulate low robust to document passing through document processing or OCR	No
Particular vowel creating sequent changing [21]	Digital plain text	Invisible	Thai	1.93 bits/text line or 57.9 bits/30 text-line/A4 page	Very low robust to document passing through document processing	No
Existence of double letter (aa-zz) in the text [4]	Digital plain text	Invisible (Zero Watermark)	English	50 bits/ A4 page	Could not be applied to all types of text documents	No

Table 5.4 Advantages, disadvantages and limitations of five crossed-line watermarking methods

Criteria	Method 1		Method 2		Method 3		Method 4		Method 5	
	Advantages	Disadvantages/ Limitations	Advantages	Disadvantages/ Limitations	Advantages	Disadvantages/ Limitations	Advantages	Disadvantages/ Limitations	Advantages	Disadvantages/ Limitations
Watermark Type		Actual watermark (Easy to fade out after making many copies)	Zero watermark		Zero watermark		Zero watermark		Zero watermark	
Watermark covering area		Poor (Able to embed only space between text lines)		Fair (Able to embed every intersection point along horizontal (H) crossed lines)		Fair (Able to embed every intersection point along vertical (V) crossed lines)	Good (Able to embed every intersection point along double (H&V) crossed lines)		Excellent (Able to embed every intersection point along main and shifted H, V and H&V crossed lines of 11 layers)	
Number of languages	Multiple		Multiple		Multiple		Multiple		Multiple	
Best language applicable (based on hiding bit capacity)	Any languages (depending on number of space between lines)		Arabic (ARr = 1.0)	English, Thai, Chinese (ARr = 1.2 – 2.0)	Arabic (ARr = 1.0)	English, Thai, Chinese (ARr = 1.2 – 2.0)	English, Thai, Chinese, Arabic (ARr = 1.0 – 2.0)		English, Thai, Chinese, Arabic (ARr = 1.0 – 2.0)	

Table 5.4 Advantages, disadvantages and limitations of five crossed-line watermarking methods *(cont.)*

Criteria	Method 1		Method 2		Method 3		Method 4		Method 5	
	Advantages	Disadvantages/ Limitations	Advantages	Disadvantages/ Limitations	Advantages	Disadvantages/ Limitations	Advantages	Disadvantages/ Limitations	Advantages	Disadvantages/ Limitations
Average Hiding Capacity (based on Arabic language)		Very low (12,500 bits of space between lines)		Low (75,329 bits of horizontal (H) crossed-lines)		Very low (38,053 bits of vertical (V) crossed-lines)	Medium (240,608 bits of H&V crossed-lines)		Very high (1,545,738 bits for 11 crossed-line layers)	
Watermark detectable after geometric attacking		Fair		Fair		Fair	Good (based on main H&V crossed-lines)		Excellent (based on main and shifted H&V crossed-lines)	
Watermark detectable after text manipulation attacking		Fair	Excellent		Excellent		Excellent		Excellent	
Integrity verification capability		Unsatisfied (No reference crossed-line intersection points)	Good (Verify with horizontal crossed lines)		Good (Verify with vertical crossed lines)		Good (Verify with double (H&V) crossed lines)		Excellent (Verify with 11 layers of crossed lines)	
Considerable Text image quantity (based on three minutes of waiting)	High (for 25 pages)		High (for 25 pages)		High (for 25 pages)		High (for 25 pages)			Very low (for 2 pages)
Text image resolution limitation	150 dpi and above		150 dpi and above		150 dpi and above		150 dpi and above		150 dpi and above	

Table 5.4 Advantages, disadvantages and limitations of five crossed-line watermarking methods *(cont.)*

Criteria	Method 1		Method 2		Method 3		Method 4		Method 5	
	Advantages	Disadvantages/ Limitations	Advantages	Disadvantages/ Limitations	Advantages	Disadvantages/ Limitations	Advantages	Disadvantages/ Limitations	Advantages	Disadvantages/ Limitations
Text image color limitation		Only grayscale text image		Only grayscale text image		Only grayscale text image		Only grayscale text image		Only grayscale text image
Memory loading while processing (Exclude memory used by MATLAB)	Medium (~205.2 MB/ one redundant block/page)		Medium (~205.2 MB/ one redundant block/page)		Medium (~205.2 MB/ one redundant block/page)		Medium (~205.2 MB/ one redundant block/page)			Very high (~2,257.2 MB/ one redundant block/page/11 layers)
Time consumed for embedding and detecting (based on one redundant block/page/layer)	Low (~7 sec./ page)		Low (~7 sec./ page)		Low (~7 sec./ page)		Low (~7 sec./ page)			High (~77 sec./11 layers/page)

5.2 Recommendation

Even this research has tried to do many experiments, it still has many ways to apply and get benefits from these multi-layers of crossed-line intersections and four-collinear-point cross ratio, especially to get more hiding-bit capacity by applying more pattern crossed-line intersections such as using the diagonal lines, sine-curve lines and some unique barcode lines. In addtion, the real identity of the original text image is also proved by directing them with the relative vector among the crossed-line intersection points in order to get more efficiency of verifying some original text modifications and enhance its robustness. For example, two double (H & V) crossed-line intersection points may be selected and virtually drawn the lines connect one another, with storing their relative vector data, to form the unique web or mesh network of virtually watermarking points, by locking with the cross ratio format, of each text image. This unique mesh network will be acted as the fingerprint of watermarking pattern which can effectively use for both its integrity verification and watermark retrieval after it has been attacked. Thus, our next researches, we have planned to focus on these advanced techniques.

REFERENCES

[1] Z. Jalil, and A. M. Mirza, "A Review of Digital Watermarking Techniques for Text Documents", **International Conference on Information and Multimedia Technology**, 2009.

[2] K.U. Jaseena, and A. John, "An Invisible Zero Watermarking Algorithm using Combined Image and Text for Protecting Text Documents", **International Journal on Computer Science and Engineering (IJCSE)**, vol. 3 (6), June 2011, pp. 2265-2272.

[3] Z. Jalil, M. Farooq, H. Zafar, M. Sabir, and E. Ashraf," Improved Zero Text watermarking Algorithm against Meaning Preservation Attacks", **World Academy of Science, Engineering and Technology,** 2010, pp. 593-597.

[4] Z. Jalil, A. M. Mirza, and T. Iqbal, "A Zero-Watermarking Algorithm for Text Documents using Structural Components", **International Conference on Information and Emerging Technologies (ICIET 2010)**, June 14-16, 2010.

[5] A. Li, B. Lin, Y. Chen, and G. Lu, "Study on copyright authentication of GIS vector data based on Zero-watermarking", **The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences**. vol. VII. Part B4, 2008, pp.1783-1786.

[6] H. S. M. Coxeter, and S. L. Greitzer, "Collinearity and Concurrence.", **Geometry Revisited**, Ch. 3, Math. Assoc. Amer, 1967, pp. 51-79.

[7] R. Mohr, and L. Morin, "Relative Positioning from Geometric Invariants," **Proceedings of the Conference on Computer Vision and Pattern Recognition**, 1991, pp. 139-144.

[8] R.O. Duda, and P.E. Hart, "Use of the Hough Transform to detect lines and curves in pictures", **Comm. ACM 15**, 1972, pp. 11-15.

[9] C.R. Jung, R. Schramm, "Rectangle Detection based on a Windowed Hough Transform", **Computer Graphics and Image Processing, 2004**, October 17-20, 2004.

[10] J. Cox, M. L. Miller, and J. A. Bloom, **Digital Watermarking,** Morgan Kaufmann Publishers, 2002.

[11] J. L. Rodgers, and W. A. Nichewander, "Thirteen Ways to Look at the Correlation Coefficient", The American Statistician, Vol. 42, No. 1, pp. 59-66, 1988.

[12] CinemaSource, Inc. "Understanding Aspect Ratios", **CinemaSource Technical Bulletins,** Ch. IV.6, The CinemaSource Press, 2001, pp. 3-4.

[13] W.D. Smith, and N.C. Wormald, "Geometric separator theorems and applications", **Foundations of Computer Science, 1998**, November 8-11, 1998.

[14] P. Murrell, " Raster Images in R Graphics", **The R Journal,** Vol.3/1, June 2011, pp. 48-54.

[15] A.M. Alattar, and O.M. Alattar, "Watermarking electronic text documents containing justified paragraphs and irregular line spacing", **Proceedings of SPIE – Volume 5306,** Security, Steganography, and Wateramaking of Multimedia Contents VI, 2004, pp. 685-695.

[16] S.H. Low, N.F. Maxemchuk, J.T. Brassil, and L. O'Gorman, "Document marking and identification using both line and word shifting", **Proceedings of the Fourteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'95)**, vol.2, 1995, pp. 853-860.

[17] J. T. Brassil, S. Low, N. F. Maxemchuk, and L. O'Gorman, "Electronic Marking and Identification Techniques to Discourage Document Copying", **IEEE Journal on Selected Areas in Communications,** Vol.13, No.8, Oct 1995, pp.1495-1504.

[18] D. Huang, and H. Yan, "Interword Distance Changes Represented by Sine Waves for Watermarking Text Images", **IEEE Trans. Circuits and Systems for Video Technology,** Vol. 11, No. 12, pp. 1237-1245, 2001.

[19] Y. Kim, K. Moon, and I. Oh, "A Text Watermarking Algorithm based on Word Classification and Inter-word space Statistics", **Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR'03)**, 2003, pp. 775-779.

[20] W. Zhang, Z. Zeng, G. Pu, and H. Zhu, " Chinese Text Watermarking Based on Occlusive Components", **IEEE**, pp. 1850-1854, 2006.

[21] M.H. Shirali-Shahreza, and M. Shirali-Shahreza, "A New Approach to Persian/ Arabic Text Steganography", **IEEE International Conference on Computer and Information Science**, 2006.

[22] S. Ranganathan, A. J. Ali, K. Kathirvel, and M. Kumar, "Combined Text Watermarking", **International Journal of Computer Science and Information Technologies**, Vol. 1 (5) , pp. 414-416, 2010.

[23] M. Du, and Q. Zhao, "Text Watermarking Algorithm based on Human Visual Redundancy", **AISS Journal, Advanced in Information Sciences and Service Sciences**, Vol. 3, No. 5, pp. 229-235, 2011.

[24] N. Samphaiboon, and M. N. Dailey, " Steganography in Thai text ", In Proc. of 5th International Conference on Electrical EngineeringElectronics Computer Telecommunications and Information Technology , **IEEE ECTI-CON 2008**, pp. 133-136, 2008.

[25] U. Topkara, M. Topkara, and M. J. Atallah, "The hiding Virtues of Ambiguity: Quantifiably Resilient Watermarking of Natural language Text through Synonym Substitutiions", **In Proc. Of ACM Multimedia andSecurity Conference**, 2006.

[26] F. Antonio, "Faster Line Segment Intersection", **Graphics Gems III**, Ch. IV.6, Academic Press, 1999, pp. 199-202 and 500-501.

APPENDIX

APPENDIX A

Publications

1. Wiyada Yawai and Nualsawat Hiransakolwong: "Increase the Hiding-bit Capacity and Strength for Text Watermarking with the Line Intersection on Text Image", **The 8th International Conference on Computing Technology and Information Management (8th NCM and 3rd ICNIT)**, 2012, pp. 427-433.
2. Wiyada Yawai and Nualsawat Hiransakolwong: "Cross-Ratio Analysis for Building up The Robustness of Document Image Watermark", **The 8th International Conference on Computing and Information Technology**, 2012, pp. 81-86.
3. Wiyada Yawai and Nualsawat Hiransakolwong: "Applying of the Cross Ratio to Significant Increase the Watermark Redundancy of a Text Image", **Applied Mathematical Sciences**, vol. 6, 2012, no. 113, pp. 5617-5637.
4. Wiyada Yawai and Nualsawat Hiransakolwong: "Grid-line watermarking: A novel method for creating a high-performance text-image watermark", **ScienceAsia**, Volume 39 Number 4 (August 2013), pp. 423–435.

2012 IC²IT

THE 8TH INTERNATIONAL CONFERENCE ON COMPUTING AND INFORMATION TECHNOLOGY

PROCEEDINGS OF IC²IT 2012
THE 8TH INTERNATIONAL CONFERENCE ON COMPUTING AND INFORMATION TECHNOLOGY
9-10 MAY 2012

DUSIT THANI HOTEL, PATTAYA CITY
THAILAND

WWW. IC2IT.ORG

King Mongkut's Institute of Technology Ladkrabang

IEEE
THAILAND SECTION

FACULTY OF INFORMATION TECHNOLOGY

KING MONGKUT'S UNIVERSITY OF TECHNOLOGY NORTH BANGKOK

WWW.IT.KMUTNB.AC.TH

Cross-Ratio Analysis for Building up The Robustness of Document Image Watermark

Wiyada Yawai
Department of Computer Science, Faculty of Science
King Mongkut's Institute of Technology Ladkrabang
Bangkok, Thailand
E-mail: yawai.wiyada@gmail.com

Nualsawat Hiransakolwong
Department of Computer Science, Faculty of Science
King Mongkut's Institute of Technology Ladkrabang
Bangkok, Thailand
E-mail: khnualsa@kmitl.ac.th

***Abstract*—This research presents the applied cross-ratio theory effectively used for building up the robustness of invisible watermarks which embedded in multi-language; English, Thai, Chinese, and Arabic, grayscale document image against the geometric distortion attacks; scaling, rotating, shearing and other manipulating, like noise signal adding, compression, sharpness, blur, brightness and contrast adjusting, that occurs while scanning the embedded watermarks for verifying them. These attacks are simulated to test the effectiveness of cross-ratio theory initiatively used to enhance this mentioned watermark robustness for any document image of any language which normally is the main limitation of other watermarking methods. This theory is using 4 corners and two diagonals of a document image as the reference for watermark embedding lines located between text lines crossing against two diagonals and vertical lines of both sides according to specified cross-ratio values. For watermark embedding positions on each line will be calculated from another set of cross-ratio. Cross-ratio values of each line will be different in accordance with preset patterns. Detection of watermarks in document images is not necessary to be converted image or compared with original image. Our approach can be detected through calculation of referred 4 corners of the image and applied correlation coefficient equation to directly compare against original watermarks. Testing revealed that it could build up reasonably robustness against scaling, at range 11% up, shearing (0 - 0.05), rotating (1 – 4 degrees), compressing, range 60% up, contrasting (1 – 45%), sharpness (0 – 100%) and blur filtering at mask size less than 13x13.**

Keywords-Digital watermarking; Document image; Robustness; Geometric distortion; Cross ratio; Collinear points

I. Introduction

Digital watermarking is one of the processes of hiding data for protecting copyright of digital media either in forms of audio, video, text, etc. There are two categories of watermarking; visible watermarking and invisible watermarking. The major purpose of watermarking is to protect copyright of media through creating various forms of obstructions to violators.

This research is particularly emphasized on applying the cross ratio theory to create the robust watermark data embedded in the grayscale document image which must be survived and easily detected even it has been attacked in many possible ways, especially geometric distortion attacks which mostly not been explored in other document image watermarking researches. Most existing researches are focused on watermarking an electronic text or document file, instead of document image, of one specified language, instead of multi-language, and emphasized the watermark embedding technique instead of watermark robustness. These existing document watermarking researches can be categorized, by their watermarking technique, into 3 techniques as follows.

Technique I: Watermark embedding with text document physical layout/pattern/structure rearrangement, such as shifting of lines [1] and words, particularly binding of word spacing, word shift coding or word classification [2][3][4][5]. This technique can be applied to both watermark electronic document file and document image. However, it has some disadvantages, for instance, line shifting technique of Brassil et will be low robust to document passing through document processing, page skewing/rotation; between -3° - +3°, noise signal adding attack and a short text line. Another limitation of this process is that it can only apply to document with spaces between words, spacing of letters, shifting of baselines or line shift coding. Word shifting algorithm has also developed by Huang et al. [5], it's based on adjusting inter-word spaces in a text document so that mean spaces across different lines show characteristics of a sine wave where information or watermark can be encoded or embedded in the sine wave(s) for both horizontal and vertical directions.

Min Du et al. [6] proposed a text digital watermarking algorithm based on human visual redundancy. According to that the human eye is not sensitive to the slight change for text color, watermarks were embedded by changing the low-4 bits of RGB color components of characters. This proposed method has good invisibility and robustness which depending on its redundancy. However, this research tested its robustness against word deleting and modifying only.

Technique II: Embedding text watermark by text character/letter feature modifying, for example, Brassil, et a l. [2] have used the letter adjusting by reducing or increasing length of letter; such as, increasing length of letters b, d, or h. For principle applying to extract hidden data out of document can be done by comparing hidden data document against original document. The limitation of this process is that the hidden data will be so little robust to document passing through document processing.

Applying arithmetic expression to replace characteristic of letter with close component has developed by W. Zhang et al.[7] which has applied arithmetic expression to replace characteristic of letter with close component (in square form) which the hiding is done by adjusting sizes of those characters in document file. This process is robust against attacks or destruction and unable to observe. The test referred that the mentioned hiding is durable and more difficult to observe than those of the line-shift coding, word-shift coding, and character coding but has not presented the robustness testing information in the research document. However, this research is only be applied to Chinese characters and subject to be further researched since the process has tested only the character replacement attack, not yet against durability to various forms of watermark attacks.

Shirali-Shahreza et al. [8] has applied changing of characters in Persian that a number most characters have their distinction in their peaks (Persian letter NOON) of these characters for hiding. Due to defects in using OCR in reading Persian and Arabic document image; therefore, reading of printed text from these characters for extracting hidden data is considered complicate, especially after attacking that has not yet tested.

Suganya et al. [9] proposed to modify perceptually significant portions of an image to make the algorithm that watermark is hidden in the point's location of the English letter i and j. first few bits are used to indicate the length of the hidden bits to be stored. Then the cover medium text is scanned to store a one, the point is slightly shifted up else it remains unchanged. However, this research did not refer to any robustness testing result.

Technique III: Watermarking with semantic schemes or word/vowel substitution; Topkara et al. [10] has developed a technique for embedding secret data without changing the meaning of the text by replacing words in the text by synonyms. This method deteriorates the quality of the document and a large synonym dictionary is needed.

Samphaiboon et al. [11] proposed a steganographic scheme for electronic Thai plain text documents that exploits redundancies in the way particular vowel, diacritical, and tonal symbols are composed in TIS-620, the standard Thai character set. The scheme is blind in that the original carrier text is not required for decoding. However, it can be used with only Thai text document and its watermark data is so easy to be destroyed by reediting with a word processing program.

The following presenting research has been studied on text document image (not electronic document file), scanned or copied from an original document paper, watermarking by applying the cross-ratio theory in collinear point type, in order to build up its watermark robustness against various forms of attacks, particularly geometric distortions; scaling, shearing and rotation and other manipulations; data compression, noise signal adding and brightness, contrast, scale, sharpness and blur adjusting.

II. The Cross Ratio of Four Collinear Points

The cross-ratio is a basic invariance in projective geometry (i.e., all other projective invariance can be derived from it). Here brief introduction to the cross-ratio invariance property is given.

Let A, B, C, D be four collinear points (Three or more points A, B, C,... are said to be collinear if they lie on a single straight line [12]) as shown in Fig. 1. The cross-ratio is defined as the "double ratio" in Eq. (1)

$$(A,B;C,D) = \frac{AC \cdot BD}{BC \cdot AD} \qquad (1)$$

where all the segments are thought to be signed. The cross-ratio does not depend on the selected direction of the line ABCD, but does depend on the relative position of the points and the order in which they are listed. Based on a fundamental theory, any homography preserves the cross-ratio. Thus central projection, linear scaling, skewing, rotation, and translation preserve the cross-ratio [13].

Figure 1. Collinear points A, B, C, and D

III. Analysis of Digital Watermarking for Document Image

To apply the cross ratio to digital image watermarking, three reference points are required. In this section, a method for deriving such reference points is detailed.

A. Definition

$C_a C_b$ is line from an origin point C_a to a destination point C_b where $a = 1, 2, 3, 4$ and $b = 1, 2, 3, 4$

$Cr = (CA/CD) : BA/BD = (CA/CD)(BD/BA)$

$R = (AC/CD)/Cr$

$BA = (AD * R)/(1 + R)$

$D\&B$ is distance of BA/AD is equal to the value of BA/DA

B. Embedding Scheme

Let's start by considering the embedding part. The method is described algorithmically below.

1) Predefine the set of cross-ratio values, to be used in subsequent steps.

2) Find the image center, as denoted by D_c, by using the line intersection formula [14] (two diagonal lines of the image) as described by Eqs.(2) ~ (3) below.

$$x_c = x_t / x_b \quad (2)$$

$$y_c = y_t / y_b \quad (3)$$

$$x_t = \begin{vmatrix} a & x_1 - x_4 \\ b & x_3 - x_2 \end{vmatrix} \quad x_b = \begin{vmatrix} x_1 - x_4 & y_1 - y_4 \\ x_3 - x_2 & y_3 - y_2 \end{vmatrix}$$

$$y_t = \begin{vmatrix} a & y_1 - y_4 \\ b & y_3 - y_2 \end{vmatrix} \quad y_b = \begin{vmatrix} x_1 - x_4 & y_1 - y_4 \\ x_3 - x_2 & y_3 - y_2 \end{vmatrix}$$

where a= $\begin{vmatrix} x_1 & y_1 \\ x_4 & y_4 \end{vmatrix}$, b = $\begin{vmatrix} x_3 & y_3 \\ x_2 & y_2 \end{vmatrix}$. In addition, (x_i, y_i) is the coordinate of the point $C_i, i = 1,...,4$ (see Fig. 2).

From the above equations, x_c is the x-axis value of the point D_c of two-line intersection; C_1C_4 intersect C_2C_3, and y_c is the y-axis value of the same point and $|\,|$ denotes a determinant operator as shown in Fig. 2.

3) Find each of the primary-level watermark embedding points ($D_{LU,i}$ and $D_{LD,i}$) on the left diagonal line (see Fig. 2) as described by Eqs.(4) ~ (7) below. Those points can be identified by using two corner points of the left diagonal line (C_1 and C_4), in combination with the image center point D_c as shown in Fig. 2(a) and the predefined cross-ratio values (C_r)

$$x_{LU,i} = x_1 + DsB \times (x_4 - x_1) \quad (4)$$

$$y_{LU,i} = y_1 + DsB \times (y_4 - y_1) \quad (5)$$

$$x_{LD,i} = x_1 + DsB \times (x_4 - x_1) \quad (6)$$

$$y_{LD,i} = y_1 + DsB \times (y_4 - y_1) \quad (7)$$

where $(x_{LU,i}, y_{LU,i})$, $i = 1, \ldots, M_{LU}$, is the coordinate of the point $D_{LU,i}$, A = C_1, B = $D_{LU,i}$, $C = D_c$ and $D = C_4$. In addition, $(x_{LD,i}, y_{LD,i})$, $i = 1, \ldots, M_{LD}$, is the coordinate of the point $D_{LD,i}$, $A = C_1$, $B = D_{LD,i}$, $C = D_c$ and $D = C_4$.

4) Find each of the watermark embedding points ($D_{RU,i}$ and $D_{RD,i}$) on the right diagonal line (see Fig. 2(b)) by following the steps and equation similar to those detailed in Step 3. However, now the point A in Eqs. (8) ~ (11) represents the point C_2 while the point B now represents the point C_3. By using these substitutions, those embedding points are given by

$$x_{RU,i} = x_2 + DsB \times (x_3 - x_2) \quad (8)$$

$$y_{RU,i} = y_2 + DsB \times (y_3 - y_2) \quad (9)$$

$$x_{RD,i} = x_2 + DsB \times (x_3 - x_2) \quad (10)$$

$$y_{RD,i} = y_2 + DsB \times (y_3 - y_2) \quad (11)$$

where $(x_{RU,i}, y_{RU,i})$, $i = 1, \ldots, M_{RU}$, is the coordinate of the point $D_{RU,i}$, $A = C_2$, $B = D_{RU,i}$, $C = D_c$ and $D = C_3$. In addition, $(x_{RD,i}, y_{RD,i})$, $i = 1, \ldots, M_{RD}$, is the coordinate of the point $D_{RD,i}$, $A = C_2$, $B = D_{RD,i}$, $C = D_c$ and $D = C_3$.

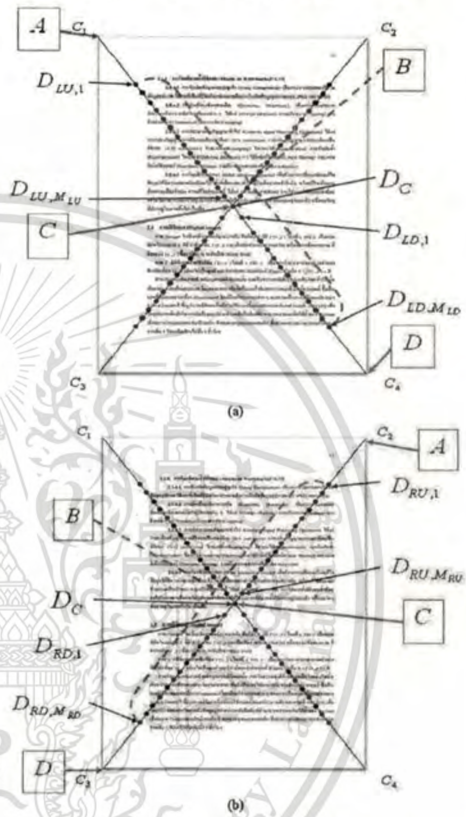


Figure 2. Notations of collinear points A, B, C, and D, defined in cross-ratio equation, on the left (a) and right (b) diagonal line of the document image.

5) For each pair of $D_{LU,i}$, $D_{RU,i}$, and $D_{LD,i}$, $D_{RD,i}$ levels, find an intersection, x_i, y_i, of crossed line of each level drawn across left side; $L_{LU,1}..L_{LD,1}$ and right side; $L_{RU,1}..L_{RD,1}$ of document image borders (see Fig. 3(a)); C_1C_3 and C_2C_4 by applying Eqs. (12) ~ (13)

$$x_i = x_t / x_b \quad (12)$$

$$y_i = y_t / y_b \quad (13)$$

$$x_t = \begin{vmatrix} a & x_1 - x_2 \\ b & x_3 - x_4 \end{vmatrix} \; x_b = \begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix}$$

$$y_t = \begin{vmatrix} a & y_1 - y_2 \\ b & y_3 - y_4 \end{vmatrix} \; y_b = \begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix}$$

where $a = \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix}, b = \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix}$

6) Find each of the watermark embedding points ($E_{HU,i,k}$ and $E_{HD,i,k}$) on the watermarked embedding lines (see Fig. 3(b)) Eqs. (14) ~ (17) represents the embedding points ($E_{HU,i,k}$ and $E_{HD,i,k}$) are given by

$$x_{HU,i,k} = x_{LU,i} + DsB \times (x_{RU,i} - x_{LU,i}) \quad (14)$$

$$y_{HU,i,k} = y_{LU,i} + DsB \times (y_{RU,i} - y_{LU,i}) \quad (15)$$

$$x_{HD,i,k} = x_{LD,i} + DsB \times (x_{RD,i} - x_{RD,i}) \quad (16)$$

$$y_{HD,i,k} = y_{LD,i} + DsB \times (y_{RD,i} - y_{RD,i}) \quad (17)$$

where ($x_{LU,i}$, $y_{LU,i}$), i = 1, ... M_{LU} , $A = L_{LU,i}$, $B = E_{HU,i,k}$, $C = D_{RU,i}$ and $D = L_{RU,i}$. In addition, ($x_{LD,i}$, $y_{LD,i}$), i = 1, ...M_{LD} , $A = L_{LD,i}$, $B = E_{HD,i,k}$, $C = D_{RD,i}$ and $D = L_{RD,i}$.

7) From all watermark embedding points, embed the watermark patterns by means of a spread-spectrum principle [15] using the following equations

Given the set of watermark embedding points $E_k = (x_k, y_k)$, k = 1, ...M, and each of the watermarking pattern bits w_k, $w_k \in$ {1,-1}, k = 1.... M, each watermarking pattern bit is embedded to the original image by using the following Eq. (18)

$$I_e(x_m^k, y_n^k) = I(x_m^k, y_n^k) + \alpha w_k \quad (18)$$

where $x_m^k = x_k + m$, m = -P, . . . , P, $y_n^k = y_k + n$, n = -Q, . . . , Q and α = strength of watermark.

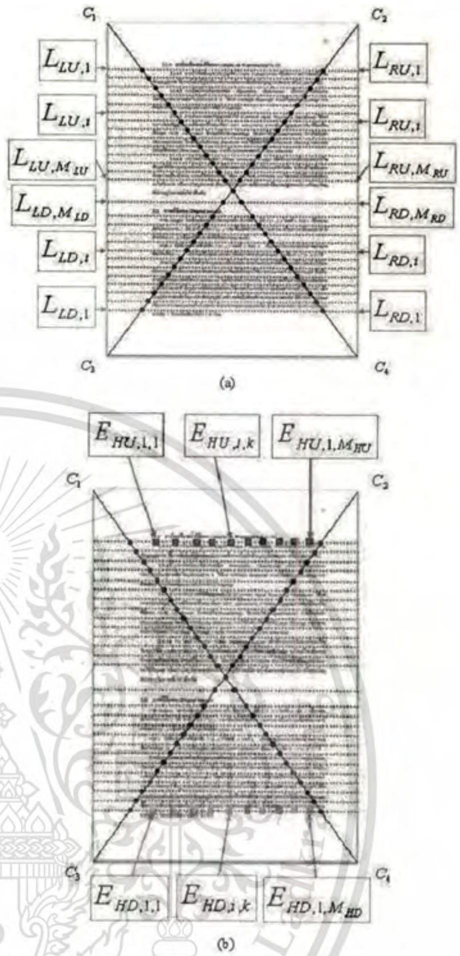


Figure 3. (a) Notations of horizontal lines intersect with 2 diagonal lines and left and right border lines in text document image. (b) Notations of collinear points used for embedding 20 invisible actual watermark pattern bits in the document image of English, Thai, Chinese, and Arabic languages.

C. Detection Scheme

To detect a watermark from the document image I_e', the four image corner points must first be detected. This can be achieved, for example, by using any of the existing corner detection algorithms. Once the four corner points are detected watermark embedding points must be identified. Each point can be calculated by using the method similar to that of the embedding stage (see Section A. for details). By extracting the values of the pixels corresponding to those watermark embedding points, denoted by $I_e'(x_k, y_k)$, a watermark can be detected by using any of the existing watermark detectors. Here, we adopt the correlation coefficient detector [15]. The correlation coefficient value is computed by the following equation.

$$Z_{cc}(I_e', w_k) = \frac{(\widetilde{I}_e \times \widetilde{W}_k)}{\sqrt{(\widetilde{I}_e \times \widetilde{I}_e)(\widetilde{W}_k \times \widetilde{W}_k)}} \qquad (19)$$

where $\widetilde{I}_e = I_e' - \bar{I}_e, \widetilde{W}_k = W_k - \overline{W}_k$

Watermark is detected if the correlation coefficient value is greater than a detection threshold. For example, in the experiment that follows, the detection threshold is 0.5.

IV. Experiment

Under this computer simulation experiment, 35 grayscale multi-language document images, size of 1240x1754 pixels, were used to add 20 different invisible actual watermark patterns of length 100 bits, α is 3, block size of watermark 5x5 pixels/watermark pattern bit. The cross-ratio values used for watermark embedding and detecting were 120 values.

The results of experiment for digital embedding in 35 grayscale document images comprising of the images with text in English, Thai, Chinese, and Arabic applied with 20 various watermark patterns (see Fig. 4), through measuring of watermark values from correlation coefficient by fixing threshold value equal to 0.5 (if there are watermarks in text document images with threshold value from 0.5 onward, if there is no watermarks its value must be less than 0.5) revealed the reasonable watermark robustness enhancement of the cross-ratio applying.

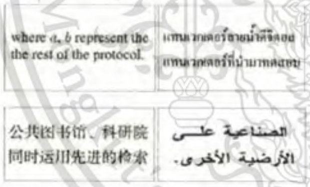


Figure 4. Show some examples of 20 random invisible actual watermark pattern bits which were created and embedded between each text line of English, Thai, Chinese, and Arabic text document images.

Firstly, tested the controlled document image without watermark by comparing with watermark pattern could obtain value of correlation coefficient = 0, while the image with watermark pattern obtained value of correlation coefficient = 1.

After that testing the cross-ratio watermarking robustness with 9 attacks, including 3 geometric distortions; shearing, scaling and rotating and 6 manipulations; compression, sharpness, brightness, contrast, blur masking and noise signal adding. The actual watermark detecting results can be classified their effects into three groups as follows:

Group I: No effect on actual watermark robustness has been found under attacking of sharpness which shown the correlation coefficient = 1, for all percentage of sharpness filtering variation, range 0 – 100%.

Group II: Very low effect on actual watermark robustness has been found under attacking of compression, at the range 60 – 100 % of JPEG compression quality (see Fig. 5), scaling, at the range 11 – 60% of scaling factor (see Fig. 6), blur, at the range 3x3 – 13x13 of blur filtering mask size (see Fig. 7), contrasting, at the range 1 – 45%, shearing, at the range 0 – 0.05 and rotating, at the range of angle between 1- 4 degrees (see Fig. 8) which shown the acceptable correlation coefficient values between 0.5 - 1, for all kinds of attacking values specified above.

Group III: High effect on actual watermark robustness has been found under attacking of brightness, at the range higher than 5%, Salt & Pepper noise signal adding, at the range higher than 1.5%, and Gaussians noise signal adding, at all ranges shown unacceptable correlation coefficient values which mostly near "0". It has shown that noise disturbing

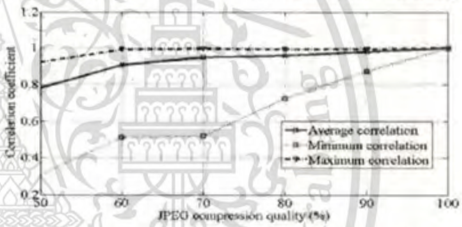


Figure 5. Correlation coefficient of watermarked multi-language document images which shown that all invisible actual watermarks still be reasonably detected after making the JPEG compression quality (%) down to 60% level.

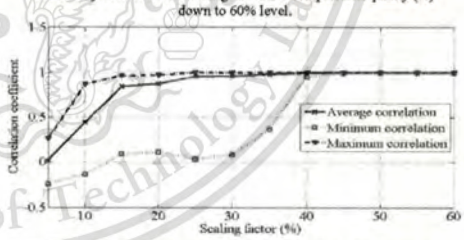


Figure 6. Correlation coefficient of watermarked multi-language document images in scaling factors which shown its robustness if varied scaling factors between 11% and 120 %.

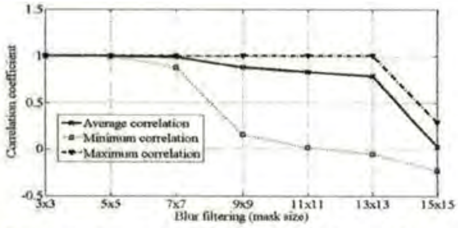


Figure 7. Correlation coefficient of watermarked multi-language document images after attacked with blur filtering mask size between 3x3 and 15x15. It shown that its robustness could be kept at the blur filtering mask size 13x13.

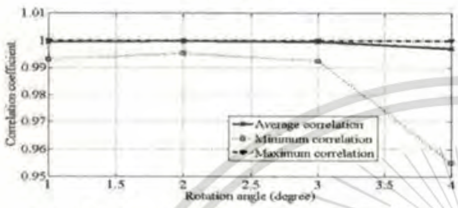


Figure 8. Correlation coefficient values of watermarked multi-language document images which rotation angle varied from 1 to 4 degrees which has still shown its robustness.

signals is the most complicated factor to affect watermark detecting, if there are more disturbing signals, detecting for watermarks can be difficult to be done.

V. Conclusions

The correlation coefficient measurement, acceptable values between 0.5 – 1, which has been used for detecting the invisible grayscale watermark existing on the multi-language document image file, has shown that the cross-ratio theory applying could be effectively used to build up the reasonably watermarking robustness against the geometric distortion attacks; scaling, especially at the range higher than 11%, shearing (0 – 0.05) and rotating (1 – 4 degrees) and some manipulating attacks; compressing, at the range higher than 60%, contrasting (1 – 45%), sharpness (0 – 100%) and blur filtering which mask size should not be greater than 13x13. This built-up robustness is based on four collinear points which have been used as the watermark embedding patterns and the referred points for watermark detection. It is not necessarily to be inversely transformed before detecting watermark positions, but can be directly detecting watermark position at once, and it is not necessarily compared with original document image without watermark. Confirmation of our document from watermark detecting can be proved directly through comparison of the existed watermark pattern.

The experiment has also shown that it can be applied for all multi-language document images, not depending on specific language attributes like some methods mentioned above which mostly focused on testing only one specific language and not thoroughly explored the possible attacks which affect the watermark robustness. This is the original step of applying the cross ratio theory for grayscale multi-language document image watermarking. For the next step we hope to improvise it to build up robustness significantly higher, especially resist the noise signal adding, rotating and brightness attacks.

References

[1] J. T. Brassil, et al., "Electronic Marking and Identification Techniques to Discourage Document Copying", IEEE Journal on Selected Areas in Communications, Vol.13, No.8, Oct 1995, pp.1495-1504.

[2] S.H. Low, N.F. Maxemchuk, J.T. Brassil, and L. O'Gorman, "Document marking and identification using both line and word shifting", Proceedings of the Fourteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'95), vol.2, 1995, pp. 853-860

[3] Y. Kim, K. Moon, and I. Oh, "A Text Watermarking Algorithm based on Word Classification and Inter-word space Statistics", Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR'03), 2003, pp. 775-779.

[4] A.M. Alattar and O.M. Alattar, "Watermarking electronic text documents containing justified paragraphs and irregular line spacing", Proceedings of SPIE – Volume 5306, Security, Steganography, and Wateramaking of Multimedia Contents VI, 2004, pp. 685-695.

[5] D. Huang, and H. Yan, "Interword Distance Changes Represented by Sine Waves for Watermarking Text Images", IEEE Trans. Circuits and Systems for Video Technology, Vol. 11, No. 12, pp. 1237-1245, 2001.

[6] Du Min and Zhao Quanyou, "Text Watermarking Algorithm based on Human Visual Redundancy", AISS Journal, Advanced in Information Sciences and Service Sciences. Vol. 3, No. 5, pp. 229-235, 2011.

[7] W. Zhang, Z. Zeng, G. Pu, and H. Zhu, " Chinese Text Watermarking Based on Occlusive Components", IEEE, pp. 1850-1854, 2006.

[8] M.H. Shirali-Shahreza, and M. Shirali-Shahreza, "A New Approach to Persian/ Arabic Text Steganography", IEEE International Conference on Computer and Information Science, 2006.

[9] Ranganathan Suganya, Johnsha Ahamed, Ali, Kathirvel.K & Kumar, Mohan, "Combined Text Watermarking", International Journal of Computer Science and Information Technologies, Vol. 1 (5) , pp. 414-416, 2010.

[10] U. Topkara, M. Topkara, M. J. Atallah, "The hiding Virtues of Ambiguity: Quantifiably Resilient Watermarking of Natural language Text through Synonym Substitutiions", In Proc. Of ACM Multimedia andSecurity Conference, 2006

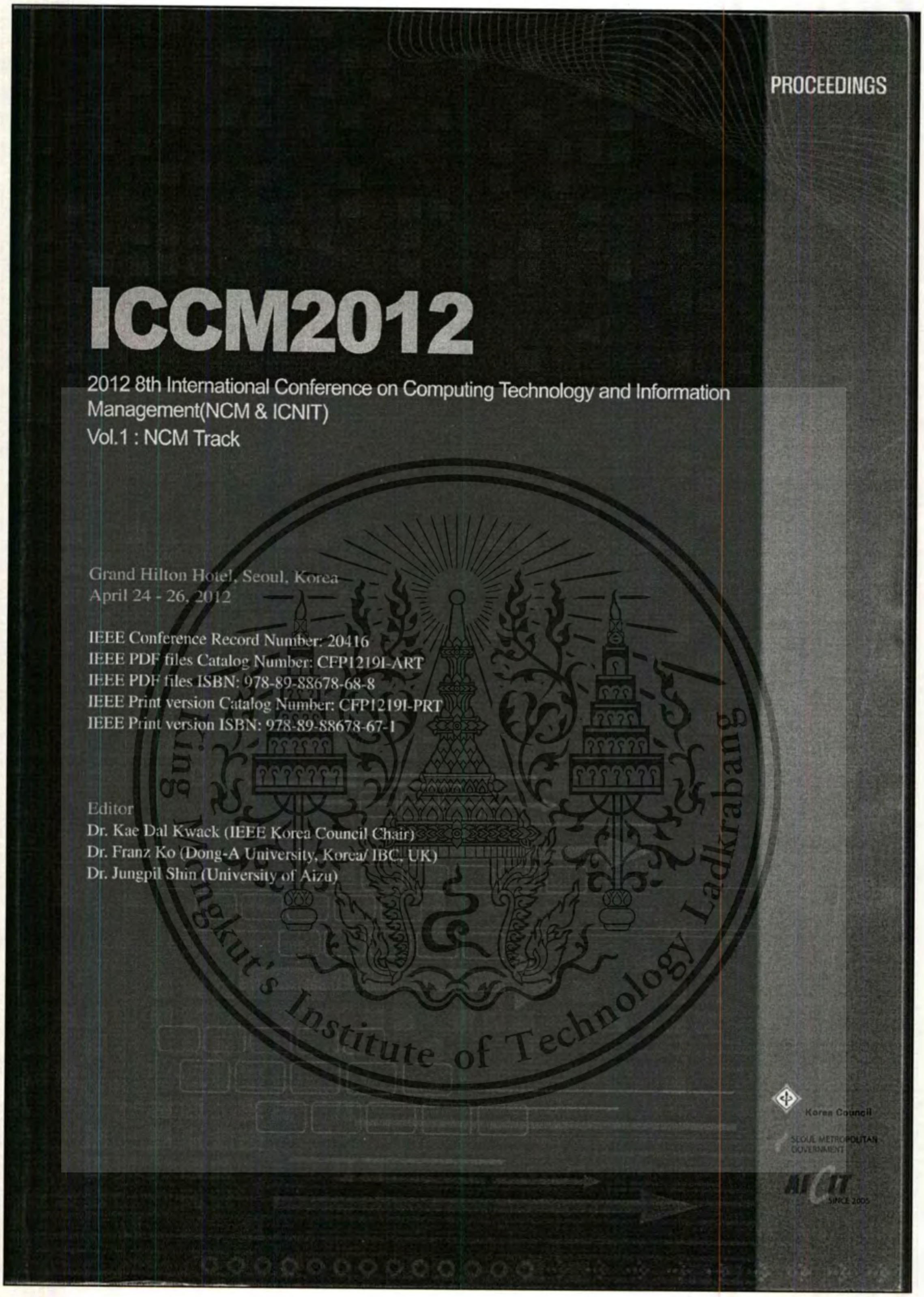
[11] Samphaiboon Natthawut, and Dailey Matthew N, "Steganography in Thai text ", In Proc. of 5th International Conference on Electrical EngineeringElectronics Computer Telecommunications and Information Technology , IEEE ECTI-CON 2008, pp. 133-136, 2008.

[12] Coxeter, H. S. M. and Greitzer, S. L. "Collinearity and Concurrence.", Geometry Revisited, Ch. 3, Math. Assoc. Amer, 1967, pp. 51-79.

[13] R. Mohr and L. Morin, "Relative Positioning from Geometric Invariants," Proceedings of the Conference on Computer Vision and Pattern Recognition, 1991, pp. 139-144.

[14] Antonio, F. "Faster Line Segment Intersection", Graphics Gems III, Ch. IV 6, Academic Press, 1999, pp. 199-202 and 500-501.

[15] J. Cox, M. L. Miller, and J. A. Bloom, [Digital Watermarking], Morgan Kaufmann Publishers, 2002


PROCEEDINGS
ICCM2012
2012 8th International Conference on Computing Technology and Information Management(NCM & ICNIT)
Vol.1 : NCM Track
Grand Hilton Hotel, Seoul, Korea
April 24 - 26, 2012
IEEE Conference Record Number: 20416
IEEE PDF files Catalog Number: CFP1219I-ART
IEEE PDF files ISBN: 978-89-88678-68-8
IEEE Print version Catalog Number: CFP1219I-PRT
IEEE Print version ISBN: 978-89-88678-67-1
Editor
Dr. Kae Dal Kwack (IEEE Korea Council Chair)
Dr. Franz Ko (Dong-A University, Korea/ IBC, UK)
Dr. Jungpil Shin (University of Aizu)
Korea Council
SEOUL METROPOLITAN GOVERNMENT

Increase the Hiding-bit Capacity and Strength for Text Watermarking with the Line Intersection on Text Image

Wiyada Yawai
Department of Computer Science, Faculty of Science
King Mongkut's Institute of Technology Ladkrabang
Bangkok, Thailand
E-mail: yawai.wiyada@gmail.com

Nualsawat Hiransakolwong
Department of Computer Science, Faculty of Science
King Mongkut's Institute of Technology Ladkrabang
Bangkok, Thailand
E-mail: khnualsa@kmitl.ac.th

***Abstract*- This research presents how to use the intersection position of horizontal line, virtually run across text character skeleton line on a text image under the cross ratio applying, to be the marking point of zero watermark; means watermarking without any actual watermark, which consequently increase the remarkable watermark hiding-bit capacity and robustness. One point of intersection across one text character skeleton line is one specific pixel which defined as one hiding-bit of zero watermark; just only marking its pattern, so that the more reference horizontal lines plotting, the more points of intersection would be got and increasing the hiding-bit capacity for watermarking which generally is the most limitation of other text watermarking methods. The high capacity for hiding bits means more chance to make many sets of redundant encrypted watermark data, in case some parts of text document image have been modified or tampered with. In addition, intersection points for virtually embedding watermarks still have a great strength for identifying the point of text which has been modified if all positions of intersection points are collected. Moreover, these intersection positions also has been applied with the cross ratio of four collinear points, to build up its robustness in order to survive from possibly attacking. The testing proved that this watermarking method can be effectively used to hide more redundant data, verify its integrity, be robust and easily apply it to multi-language text images such as English, Thai, Chinese and Arabic languages.**

Keywords-Digital watermarking; Hiding-bit capacity; Text image; Robustness; Geometric distortion; Cross ratio; Collinear points; Zero watermark

I. INTRODUCTION

Hiding watermark data is the main process of digital watermarking for protecting copyright of digital media either in forms of audio, video, text, etc. However, it still has some limitations of the hiding data process, especially for a black and white text document image file. Main limitations are making digital watermark invisible on text and increasing a hiding-bit capacity to have enough volume for embedding all watermark data they want.

Mostly, text image is embedded with the visible watermarking for indicating the ownership of such media. In case of making invisible watermark, a document image must be embedded with slightly hiding data in order to difficultly observe. The ideal solutions for watermarking should be easily processed, robust, and imperceptible. It should also been convertible into various forms of texts with high data capacity as well as be efficient in printing/digital proofing.

This research is emphasized on applying the virtual character-line intersection for virtually embedding and detecting zero/invisible watermark on a text image which eventually increasing its hiding-bit capacity and robustness of watermark against some possible attacks.

At present, most existing researches are depending on shifting of lines and words, particularly binding of word spacing or word shift coding [1-4]. Brassil et al. [2] proposed watermarking techniques for images containing text. In this paper propose techniques that discourage unauthorized distribution by embedding each document with a unique codeword, based on text-line shifting. Low et al. [1] and Alattar et al. [4] proposed hiding data by horizontal relocation of words in each line (to left or right). The principle is that if any word be shifted to right will be replaced by a "1" binary, but if any word be shifted to left will be replaced by an "0" binary, and the shifted word must be between un-shifted words. The principle in identifying for extracting data out of the document will be considered from the size of space between the shifted words from the original locations for specifying of the hiding bit value. However, the hiding data limitation depending on number of inter-space between words and it will be low robust to document passing through document processing; and another limitation of this process is that it can only apply to document with spaces between words, spacing of letters, shifting of baselines or line shift coding, but the limitation is ability to hide low data since it must use up to two lines for hiding data of one bit. Moreover, hiding of data through this process will be easy to observe the change of document since hiding data of this process will change location of the total line causing spaces of each line in document page are not equal.

Another word shifting algorithm has been developed by Huang et al. [5], by adjusting inter-word spaces that replace one sine wave. Signals have been encoded in phrases, waves, and frequencies of sine waves. The algorithm can do both non-blind modes for inserting signals, spaces between letter should also been changed, feature coding. The limitation of this process is that the hiding data will be so little robust to document passing through document processing. W. Zhang et al. [6] has applied arithmetic expression to replace characteristic of letter with close component (in square form) which the hiding is done by adjusting sizes of those characters. This process is robust against attacks or destruction and unable to observe. The test referred that the mentioned hiding is durable and more difficult to observe than those of the line-shift coding, word-shift coding, and character coding (but has not presented the testing information in the research document). However, this research is only applied to Chinese characters and subject to be further researched since the process has not yet tested against durability to various forms of watermark attacks.

Shirali-Shahreza et al. [7] has applied changing of characters in Persian that a number most characters have their distinction in their peaks (Persian letter NOON) of these characters for hiding. Due to defects in using OCR in reading Persian and Arabic; therefore, reading of printed text from these characters for extracting hiding data is considered complicate. M.A. Qadir et al. [8] have developed the process for hiding data in digital plain text document (ASCII) for the purpose of retyping durability without changing the plain text. This research has been presented hiding watermark in information through applying with plain text information (ASCII) for unobservable of changes that create high security, difficult to destroy, and this process can be further developed for applying with Unicode characters as well as other document formats.

Suganya et al. [9] proposed to modify perceptually significant portions of an image to make the algorithm more robust against attacks. The watermark is hidden in the point's location of the letter i and j. First few bits are used to indicate the length of the hidden bits to be stored. Then the cover medium text is scanned to store a one, the point is slightly shifted up else it remains unchanged. Thus, the hiding capacity would be limited upon number of letter i and j on a document.

Topkara et al. [10] has developed a technique for embedding secret data without changing the meaning of the text by replacing words in the text by synonyms. This method deteriorates the quality of the document and a large synonym dictionary is needed and got low hiding-bit capacity. Samphaiboon et al. [11] proposed a steganographic scheme for Thai plain text documents that exploits redundancies in the way particular vowel, diacritical, and tonal symbols are composed in TIS-620, the standard Thai character set. The scheme is blind in that the original carrier text is not required for decoding. However, the hiding-bit capacity of this method is still limited because it's depending on a number of specific encrypted vowels and it could not be applied to watermark a text image. Min Du et al.[12] proposed a text digital watermarking algorithm based on human visual redundancy. According to that the human eye is not sensitive to the slight change for text color, watermarks were embedded by changing the low-4 bits of RGB color components of characters. This proposed method has good invisibility and robustness to resist deletion, modification attack etc., but still has lower hiding-bit capacity because a whole text must be dedicated for one bit of embedded watermark data.

The following presenting research has been focused on proving that the virtual character-line intersection, with the cross ration theory in collinear point type applying, is able to overcome the above mentioned limitations of text image watermarking, especially the hiding-bit capacity, robustness and language dependency. Thus, there are three main objectives of this study, the first is to study the new way to make invisible watermark on text image with not depending on some specific languages, the second is to increase remarkable hiding-bit capacity for virtually embedding digital watermark and the third is to test its robustness against various types of attack, particularly geometric distortion such as scaling, shearing and rotating.

II. Applied Mathematical Methods For Text Image Watermarking

There are two mathematical methods applied to this research, the first one is the cross-ratio theory for defining zero watermarking pattern and detecting a location of zero watermark after it has been attacked with some geometric distortion and the second one is the matching percentage method for verifying watermarking point and its integrity.

A. The Cross Ratio of Four Collinear Points

The cross-ratio is a basic invariance in projective geometry (i.e., all other projective invariance can be derived from it). Here brief introduction to the cross-ratio invariance property is given.

Let A, B, C, D be four collinear points (Three or more points A, B, C, …, are said to be collinear if they lie on a single straight line[13]) as shown in Fig. 1. The cross-ratio is defined as the "double ratio" in Eq. (1).

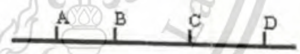


Figure 1 Collinear points A, B, C, and D

$$(A,B;C,D)=\frac{AC\cdot BD}{BC\cdot AD} \quad (1)$$

where all the segments are thought to be signed. The cross ratio does not depend on the selected direction of the line ABCD, but does depend on the relative position of the points and the order in which they are listed. Based on a fundamental theory, any homography preserves the cross-ratio. Thus central projection, linear scaling, skewing, rotation, and translation preserve the cross-ratio [14].

B. The Matching Percentage

The matching percentage is applied for comparing the watermark embedded position with the watermark detected position, one-to-one, and then count all the matched positions to calculate the percentage of matching by using Eq. (2).

$$MPW = \frac{\sum D_{wm}(x_i, y_i)}{\sum P_{ip}(x_i, y_i)} \times 100 \qquad (2)$$

where MPW = Matching percentage of watermark embedded position detection which corresponding to the selected intersection point for embedding watermark,

D_{wm} = Detected watermark at position x_i, y_i,

P_{ip} = Prominent Intersection points at position x_i, y_i which were selected to embed watermark.

III. Text Watermark Marking And Detecting Analysis

Here, the text watermarking process is based on three sub-processes; the first one is to define watermark marking; not really embedding, positions and its hiding-bit capacity with line intersection, the second one is to match the cross-ratio reference points with zero watermark marking positions for easily tracking the zero watermark marking points after a text document image has been attacked and the last one is to detect the zero watermarks and verify its integrity with the matching percentage method.

A. Hiding-bit Capacity for Watermark Marking

Able to calculate and increase the watermark hiding-bit capacity by following the procedure as described below.

1) Identify and draw the lines for the top, $L_{Top-Border}$, and bottom, $L_{Bottom-Border}$, borders of each text line (see Fig. 2 and Fig. 3).

2) Find and draw the main-middle line, $L_{Main-Middle}$, between the top and bottom border lines of each text line.

3) Find and draw the top-middle line, $L_{Top-Middle}$, between the top-border line and the main-middle line.

4) Find and draw the bottom-middle line, $L_{Bottom-Middle}$, between the bottom-border line and the main-middle line. More lines can be drawn across text characters if more hiding-bit capacity is required.

5) Detect each prominent intersection position of each line which runs across each text-character line where its tone less than 253grayscale level. These detected prominent positions would be randomly specifying as the invisible zero watermarks.

6) Count all intersection positions on each page of each language.

7) Define each intersection position as 1 bit so that total of intersection position of each page means the capacity of hiding bit for hiding watermark data.

B. Invisible Zero Watermark Marking Scheme

To apply the cross ratio to digital image watermarking, three reference points are required. In this section, a method for deriving such reference points is detailed.

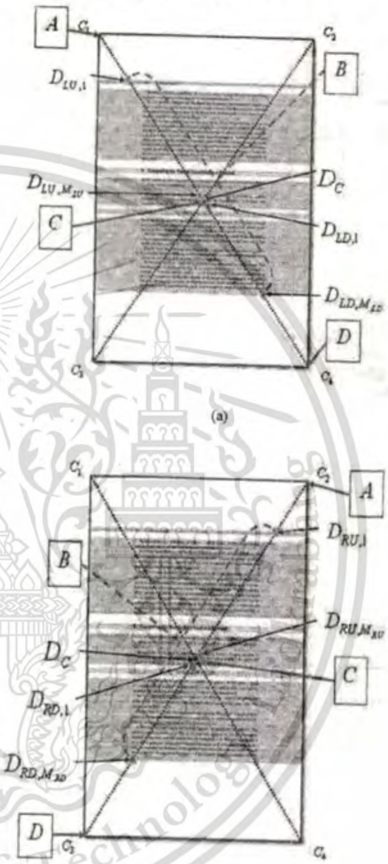
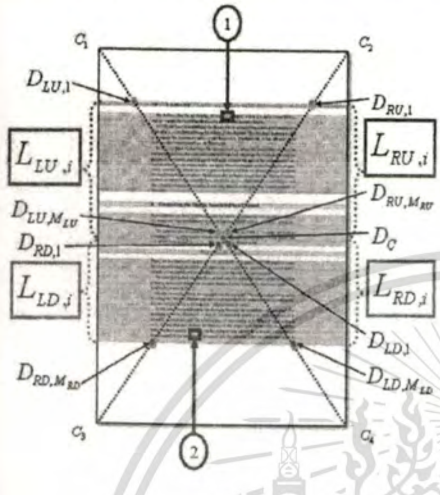


Figure 2 Notations of collinear points *A, B, C,* and *D*, defined in cross-ratio equation, on the left (a) and right (b) diagonal lines of document images.

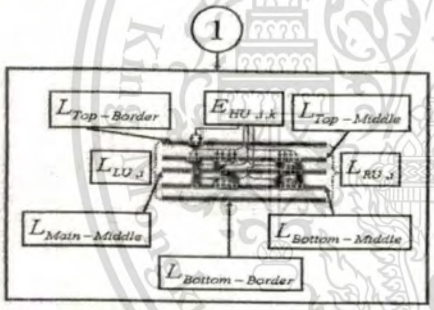
Let's start by considering the marking part. The method is described algorithmically below.

1) Predefine the set of cross-ratio values, to be used in subsequent steps.

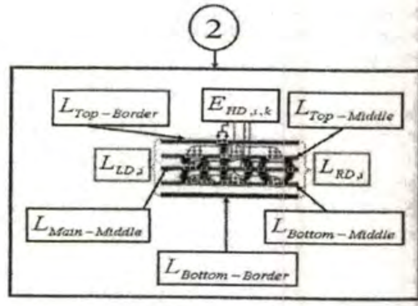
2) Find the image center, as denoted by, by using the line intersection formula [15] (two diagonal lines of the image) as described by Eqs. (3) ~ (4) below.



(a)



(b)



(c)

Figure 3 (a) Notations of horizontal lines intersect 2diagonal lines on a document image. (b) and (c) Notations of virtual 5 lines intersect text character lines to form collinear points used for embedding or marking zero watermark pattern bits in the text image.

$$x_c = x_t / x_b \quad (3)$$

$$y_c = y_t / y_b \quad (4)$$

$$x_t = \begin{vmatrix} a & x_1 - x_4 \\ b & x_3 - x_2 \end{vmatrix} \quad x_b = \begin{vmatrix} x_1 - x_4 & y_1 - y_4 \\ x_3 - x_2 & y_3 - y_2 \end{vmatrix}$$

$$y_t = \begin{vmatrix} a & y_1 - y_4 \\ b & y_3 - y_2 \end{vmatrix} \quad y_b = \begin{vmatrix} x_1 - x_4 & y_1 - y_4 \\ x_3 - x_2 & y_3 - y_2 \end{vmatrix}$$

where $a = \begin{vmatrix} x_1 & y_1 \\ x_4 & y_4 \end{vmatrix}$, $b = \begin{vmatrix} x_3 & y_3 \\ x_2 & y_2 \end{vmatrix}$.

In addition, (x_i, y_i) is the coordinate of the point $C_i, i = 1,\ldots,4$ (see Fig. 2).

From the above equations, x_c is the x-axis value of the point D_c of two-line intersection; $C_1 - C_4$ intersect $C_2 - C_3$, and y_c is the y-axis value of the same point and $|\,|$ denotes a determinant operator as shown in Fig. 2.

3) Find each of the primary-level watermark marking points ($D_{LU,i}$ and $D_{LD,i}$) on the left diagonal line (see Fig. 2) as described by Eqs. (5) ~ (8) below. Those points can be identified by using two corner points of the left diagonal line (C_1 and C_4), in combination with the image center point D_c as shown in Fig. 2(a) and the predefined cross-ratio values (C_r),

$$x_{LU,i} = x_1 + DsB \times (x_4 - x_1) \quad (5)$$

$$y_{LU,i} = y_1 + DsB \times (y_4 - y_1) \quad (6)$$

$$x_{LD,i} = x_1 + DsB \times (x_4 - x_1) \quad (7)$$

$$y_{LD,i} = y_1 + DsB \times (y_4 - y_1) \quad (8)$$

where ($x_{LU,i}, y_{LU,i}$), i = 1, ... M_{LU}, is the coordinate of the point $D_{LU,i}$, $A = C_1$, $B = D_{LU,i}$, $C = D_c$ and $D = C_4$. In addition, ($x_{LD,i}, y_{LD,i}$), i = 1, ... M_{LD}, is the coordinate of the point $D_{LD,i}$, $A = C_1$, $B = D_{LD,i}$, $C = D_c$ and $D = C_4$. DsB (Distance of BA/AD) = BA / AD, BA= (AD * R) / (1 + R) , R= (AC/CD)/ Cr , and Cr = (CA/CD) : BA/B D = (CA/CD) (BD/BA)

4) Find each of the watermark marking points ($D_{RU,i}$ and $D_{RD,i}$) on the right diagonal line (see Fig. 2(b)) by following the steps and equation similar to those detailed in Step 3. However, now the point *A* in Eqs. (9) ~ (12) represents the point C_2 while the point *B* now represents the point C_3. By using these substitutions, those marking points are given by

$$x_{RU,i} = x_2 + DsB \times (x_3 - x_2) \quad (9)$$

$$y_{RU,i} = y_2 + DsB \times (y_3 - y_2) \quad (10)$$

$$x_{RD,i} = x_2 + DsB \times (x_3 - x_2) \quad (11)$$

$$y_{RD,i} = y_2 + DsB \times (y_3 - y_2) \quad (12)$$

where ($x_{RU,i}, y_{RU,i}$), i = 1, ... M_{RU}, is the coordinate of the point $D_{RU,i}$, $A = C_2$, $B = D_{RU,i}$, $C = D_c$ and $D = C_3$. In addition, ($x_{RD,i}, y_{RD,i}$), i = 1, ... M_{RD}, is the coordinate of the point $D_{RD,i}$, $A = C_2$, $B = D_{RD,i}$, $C = D_c$ and $D = C_3$.

5) For each pair of $D_{LU,i}$, $D_{RU,i}$, and $D_{LD,i}$, $D_{RD,i}$ levels, find an intersection, ,of crossed line of each level drawn across left side; $L_{LU,1} \ldots L_{LD,1}$ and right side; $L_{RU,1} \ldots L_{RD,1}$ of document image borders (see Fig. 3(a)); $C_1 - C_3$ and $C_2 - C_4$ by applying Eqs. (13) ~ (14)

$$x_i = x_t / x_b \quad (13)$$

$$y_i = y_t / y_b \quad (14)$$

$$x_t = \begin{vmatrix} a & x_1 - x_2 \\ b & x_3 - x_4 \end{vmatrix} \quad x_b = \begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix}$$

$$y_t = \begin{vmatrix} a & y_1 - y_2 \\ b & y_3 - y_4 \end{vmatrix} \quad y_b = \begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix}$$

where $a = \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix}$, $b = \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix}$

6) Find each of the watermark marking points ($E_{HU,i,k}$ and $E_{HD,i,k}$) on the watermarked marking lines (see Fig. 3(b)) Eqs. (15) ~ (18) represents the marking points ($E_{HU,i,k}$ and $E_{HD,i,k}$) are given by

$$x_{HU,i,k} = x_{LU,i} + DsB \times (x_{RU,i} - x_{LU,i}) \quad (15)$$

$$y_{HU,i,k} = y_{LU,i} + DsB \times (y_{RU,i} - y_{LU,i}) \quad (16)$$

$$x_{HD,i,k} = x_{LD,i} + DsB \times (x_{RD,i} - x_{RD,i}) \quad (17)$$

$$y_{HD,i,k} = y_{LD,i} + DsB \times (y_{RD,i} - y_{RD,i}) \quad (18)$$

where ($x_{LU,i}$, $y_{LU,i}$), i = 1, ... M_{LU} , $A = L_{LU,i}$, $B = E_{HU,i,k}$, $C = D_{RU,i}$ and $D = L_{RU,i}$. In addition, ($x_{LD,i}$, $y_{LD,i}$), i = 1, ... M_{LD} , $A = L_{LD,i}$, $B = E_{HD,i,k}$, $C = D_{RD,i}$ and $D = L_{RD,i}$.

C. *Invisible Zero Watermark and Integrity Detection Scheme*

To detect a watermark from the text document image, the four image corner points must firstly be detected. This can be achieved, for example, by using any of the existing corner detection algorithms. Once the four corner points are detected watermark marking points must be identified. Each point can be calculated by using the method similar to that of the marking stage (see Section *B of III.* for detail). By extracting the values of the pixels corresponding to those watermark marking points, a watermark can be detected by using any of the existing watermark detectors. Here, we adopt the basic matching percentage method as described in Section *B of II.* to compare each detected watermark positions with its originally marked positions, one-by-one, till cover all specified positions. Predefine the set of cross-ratio values, to be used in subsequent steps.

IV. Experiment

In this section, results from the computer simulation experiment are reported. The images used for this experiment are 35 grayscale multi-language text document images. Each of grayscale document images of size 1240x1754 pixels was used to make crossed-line intersections with text-character skeleton lines, as described in Section *A of III.*, in order to mark them as a position for zero watermark marking, one intersection defined as one bit of zero watermark data. The cross-ratio values used for zero watermark marking and detecting for one 50 text-line page was about 33,500 values.

The results of experiment for digital marking on 35 grayscale multi-language text images comprising of the images with text in English, Thai, Chinese, and Arabic applied with 35 various zero watermark patterns, through measuring of watermark values from the percentage of matching zero watermark positions, one-to-one, between its original watermarked positions with the attacked one. Watermark is detected if found the grayscale level is less than 253.

At first, tested of the controlled watermarked document image without attacking by comparing with the plotted zero watermark positioning pattern, it got 100% of matching percentage or zero percentage of error.

On testing, it begins with the embedding process which each crossed line is virtually run over each text-character skeleton lines of each text character of each language to mark the intersected positions for virtually zero-watermark marking. This embedding step revealed that this text character-line intersection watermarking technique can effectively create a real invisible watermark on any text image of any language without any actual watermark really embedding, as per the requirement of the first objective.

This step also overcomes the limitation of hiding-bit capacity, as required in the second objective. It has shown that one page of English text image, comparing with the other languages, can be contributed the biggest volume of hiding-bit capacity, up to 33,500 bits (see Fig. 4). This breakthrough came from the nature of the English text font itself let it has number of characters on one text line and number of text lines on one page more than the other three languages. However, these hiding-bit capacities contributed by Thai, Chinese and Arabic texts with this intersection technique are still significantly larger than

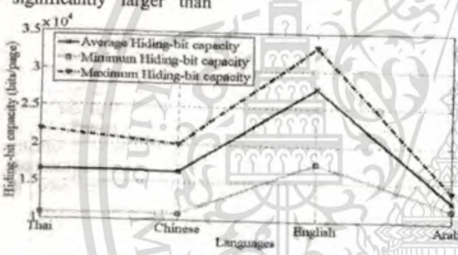


Figure 4 Comparison of the hiding-bit capacity (bits/page) among Thai, Chinese, English and Arabic languages.

other existing watermarking techniques such as inter-word shifting, line shifting, letter characteristic changing, text color changing, vowel recreating and synonym substituting which can contributed its hiding-bit capacity just only low-to-moderate level capacity, at range 25 - 2,500 bits/A4-page.

The experiment revealed that the volume of watermark hiding-bit capacity varies according to the number of skeleton lines of each text character of each language, font size, number of characters/letters per text line, number of text lines per page and number of reference crossed lines which able to prominently draw across each text character. These factors would directly contribute the text hiding-bit capacity of watermark embedding. This means the more text character – reference line intersection points are selected to mark for watermark, the more hiding-bit capacity of watermark would be got as well.

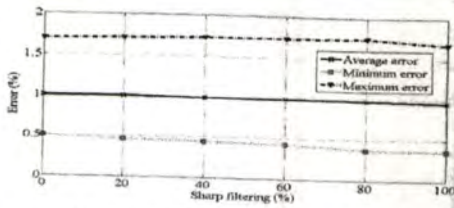


Figure 5 Low error (%) level of zero watermark detection after sharpness attacking.

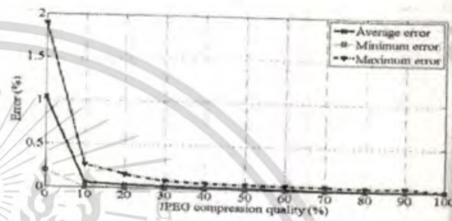


Figure 6 Low error (%) level of zero watermark detection after compression attacking.

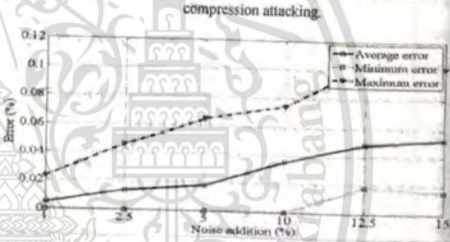


Figure 7 Low error (%) level of watermark detection after adding the salt & pepper noises.

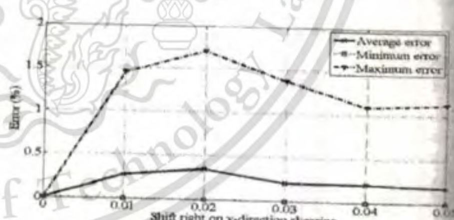


Figure 8 Low error (%) level of watermark detection after making a right shift, on x-direction, shearing.

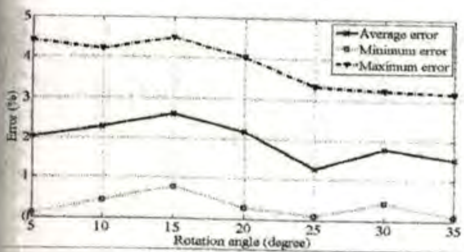


Figure 9 Error level 4.5 % of watermark detection after rotating between 5 – 35 degree of angle.

Fig. 5 – 9 has shown the remarkable achievement of the third objective with the effect of the cross ratio theory to strengthen the text-image watermarking robustness against some possible attacks such as compression, sharpness, noise signal adding, shearing and rotating. The results have shown that these attacks did not significantly affect the watermark detecting. They affected the virtually embedded watermark detecting just only 0.5 – 4.5 percent of error. This means the matching percentage of watermark detecting is mostly higher than 95.5 %.

V. Conclusions

This research has disclosed the more number of crossed lines intersected each text character of each text line on each page of a text document image, the more hiding-bit capacities and marking positions of watermarks increasing. This means more rooms for hiding redundant confidential data of watermark on each crossed line, especially for languages that have a lot of characters per line and have many lines per page such as Thai and English languages would create more line intersections, thus get more capacity for hiding bit of watermark. However, the experiment has shown that many intersection points could not be used for watermarking due to their difficulties of detection. Therefore, only some prominent intersection points have been selected for marking watermarks.

This experiment has shown that its robustness can be enhanced with the cross ratio theory of four collinear points applying. This robustness came from mathematical mechanism of cross ratio theory which able to effectively refer to its original watermark embedded positions even it has been attacked with some geometric distortions such as rotation and scaling.

However, this experiment also shown that some error could occur in case of small font size and small gap between each text line which sometime made some of 5 intersection lines located in the same level. These results revealed that the suitable gap between each reference crossed line should not less than 3 pixels, otherwise its intersection points, for zero watermark embedding, would be gone after attacking with the above mentioned geometric distortions. This bad effect is caused by small interval between each line which difficult to detect these zero watermarks after distortion attacking. This affect to the detection of its integrity or modification of its original text image as well.

References

[1] S.H. Low, N.F. Maxemchuk, J.T. Brassil, and L. O'Gorman, "Document marking and identification using both line and word shifting", Proceedings of the Fourteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM'95), vol.2, 1995, pp. 853-860.

[2] J. T. Brassil, et al., "Electronic Marking and Identification Techniques to Discourage Document Copying", IEEE Journal on Selected Areas in Communications, Vol.13, No.8, Oct 1995, pp.1495-1504.

[3] Y. Kim, K. Moon, and I. Oh, "A Text Watermarking Algorithm based on Word Classification and Inter-word space Statistics", Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR'03), 2003, pp. 775-779.

[4] A.M. Alattar and O.M. Alattar, "Watermarking electronic text documents containing justified paragraphs and irregular line spacing", Proceedings of SPIE – Volume 5306, Security, Steganography, and Wateramaking of Multimedia Contents VI, 2004, pp. 685-695.

[5] D. Huang, and H. Yan, "Interword Distance Changes Represented by Sine Waves for Watermarking Text Images", IEEE Trans. Circuits and Systems for Video Technology, Vol. 11, No. 12, pp. 1237-1245, 2001.

[6] W. Zhang, Z. Zeng, G. Pu, and H. Zhu, " Chinese Text Watermarking Based on Occlusive Components", IEEE, pp. 1850-1854, 2006.

[7] M.H. Shirali-Shahreza, and M. Shirali-Shahreza, "A New Approach to Persian/ Arabic Text Steganography", IEEE International Conference on Computer and Information Science, 2006.

[8] M.A. Qadir, and I. Ahmad, "Digital Text Watermarking: Secure Content Delivery and Data Hiding in Digital Document", IEEE, 2005.

[9] Ranganathan Suganya, Johnsha Ahamed, Ali, Kathirvel.K & Kumar, Mohan, "Combined Text Watermarking", International Journal of Computer Science and Information Technologies, Vol. 1 (5) , pp. 414-416, 2010.

[10] U. Topkara, M. Topkara, M. J. Atallah, "The hiding Virtues of Ambiguity: Quantifiably Resilient Watermarking of Natural language Text through Synonym Substitutiions", In Proc. Of ACM Multimedia andSecurity Conference, 2006

[11] Samphaiboon Natthawut, and Dailey Matthew N, " Steganography in Thai text ", In Proc. of 5th International Conference on Electrical EngineeringElectronics Computer Telecommunications and Information Technology , IEEE ECTI-CON 2008, pp. 133-136, 2008.

[12] Du Min and Zhao Quanyou, "Text Watermarking Algorithm based on Human Visual Redundancy", AISS Journal, Advanced in Information Sciences and Service Sciences: Vol. 3, No. 5, pp. 229-235, 2011.

[13] Coxeter, H. S. M. and Greitzer, S. L. "Collinearity and Concurrence.", Geometry Revisited, Ch. 3, Math. Assoc. Amer, 1967, pp. 51-79.

[14] R. Mohr and L. Morin, "Relative Positioning from Geometric Invariants," Proceedings of the Conference on Computer Vision and Pattern Recognition, 1991, pp. 139-144.

[15] Antonio, F. "Faster Line Segment Intersection", Graphics Gems III, Ch. IV.6, Academic Press, 1999, pp. 199-202 and 500-501.

HIKARI Ltd
International Publishers of Science, Technology and Medicine

| Home | Journals | Books | Paper submission | About us and our mission | News | Contact us |

HIKARI

Site Search: Go

APPLIED MATHEMATICAL SCIENCES

ISSN 1312-885X

- AIMS AND SCOPES
- INDEXING AND ABSTRACTING
- AUTHOR GUIDELINES
- FORTHCOMING PAPERS
- EDITORIAL BOARD
- PROCEEDINGS OF ICNPAA 2008
- CURRENT ISSUE

ISSN 1312-885X Applied Mathematical Sciences Hikari Ltd

Print and Online Editions
Subject: Applied Mathematics

AMS is publishing refereed, high quality original research papers in a broad range of applied mathematics and related applied sciences.

ONLINE EDITION: 2007, 2008, 2009, 2010, 2011, 2012

Applied Mathematical Sciences, Vol. 6, 2012, no. 113, 5617 - 5637

Applying of the Cross Ratio to Significant Increase the Watermark Redundancy of a Text Image

Wiyada Yawai

Department of Computer Science, Faculty of Science
King Mongkuts Institute of Technology Ladkrabang
Bangkok, Thailand
wyd.yawai@gmail.com

Nualsawat Hiransakolwong

Department of Computer Science, Faculty of Science
King Mongkuts Institute of Technology Ladkrabang
Bangkok, Thailand

Abstract

The objective of this experiment was to apply the cross ratio theory to significantly increase the watermark data redundancy and integrity verification performance of a multi-language text image. Here, the cross ratio of four collinear points was applied to control the watermark embedding and detection processes by using the intersection positions of the horizontal (H) and vertical (V) and double (H & V) grid lines which virtually run across the text-character skeleton lines as the guidelines. Three layers of grid-line intersections were tested using multi-language text images including Thai, English, Chinese and Arabic. Our experiments revealed that three layers of three patterns of grid-line intersections on each multi-language text image page, in which these grid lines were drawn and used to create multiple layers of intersection points for embedding secret data bits for watermarking effectively yielded a huge watermark hiding-bit redundancy. Nevertheless, when all points are embedded, it would be helpful for owners to be able verify their integrity.

Mathematics Subject Classification: 53B05, 53B10

Keywords: cross ratio, collinear points, text image, watermark hiding-bit redundancy, integrity verification

1 Introduction

1.1 The Redundancy Limitations of Text-Image Watermarks

Increasing watermark hiding-bit redundancy is a challenge for all text-image watermarking researchers. The difficulty of this task varies with the nature of the content, the language and the text itself, which is mostly written in monochrome and provides relatively little variation space for hiding invisible watermark data. The data are thus easy to observe. Many researchers have proposed algorithms and applied many mathematical models for hiding watermark data in a text image, typically applying only certain specific language to yield a small redundant hiding-bit data , as their algorithms are mostly based on physical watermarking, i.e., inter-word or line-space shifting. Increasing watermark hiding-bit redundancy is usually the main purposes of watermarking applications that store copyright and other secret data for all digital media and text images. Unfortunately, these data are the most difficult types to embed in a plain black-and-white text image, especially if they should be absolutely invisible.

1.2 Text-Image Watermark Redundancy in Existing Research

Text watermarking is difficult to implement because, first, text contains little redundancy compared to other media and, second, humans are sensitive to abnormal-looking text [1]. Any text transformation should preserve the meaning, fluency, grammaticality, writing style and other properties of the text. Short documents have low watermarking redundancies and are relatively difficult to protect. Text watermarking algorithms also depend on text size, language, rules, grammar, conventions and writing styles. Previous text watermarking techniques have been proposed including text watermarking using text images and synonym-, pre-supposition-, syntactic tree-, noun-verb-, word and sentence-, acronym- and typographical-error-based methods [2].

The majority of methods are physical watermark embedding techniques, applied the correlation coefficient mathematical method for embedding and detecting an embedded watermarks, that are easily observed. Text watermarking systems aim to embed information by discretely modifying the original data such that the modifications are imperceptible. In image text watermarking, this goal is achieved by exploiting the redundancy in images and the limitations of the human visual system. Conversely, language has a discrete and syntactical nature that makes such techniques more difficult to apply. Specifically, language and its consequent text representations have two important properties that differ from image representations. Sentences have a combinatorial syntax and semantics [3].

These existing researches emphasize the use of a physical watermark embedding technique, which rarely increases the watermark hiding-bit redundancy, rather than the use of a zero watermark method, without physically modifying any text character, which provides more room to increase this redundancy. The existing text watermarking studies can be classified into three groups based on their performance with respect to watermark redundancy creation (Table 1).

Table 1: Existing text watermarking methods grouped by performance regarding watermark redundancy creation.

Group	Methodology	Language	Approx. Redundancy
1st Low Redundancy	Text line shifting [2],[4],[5]	Multi-language	30 bits/A4 page
	Particular vowels changed to create sequences [1]	Thai	58 bits/A4 page
	Point shifting of letters "i" and "j" [3]	English	600 bits/A4 page
	Adjusting character size [6]	Chinese	650 bits/A4 page
	Inter-word space shifting (left-right) [7],[8]	Multi-language	720 bits/A4 page
	Substitution of words in the text with synonyms [9]	English	870 bits/A4 page
	Character peak point distinction [10]	Persian and Arabic	900 bits/A4 page
2nd Moderate Redundancy	Reducing or increasing the length of letters for feature coding [11]	English	4,000 bits/A4 page
3rd High Redundancy	Changing the Low-4 bits of RGB color of characters [12]	Chinese	40,000 bits/A4 page

1.3 Eperimental Objectives

Here, we focused on applying the cross ratio theory to significantly increase the watermark hiding-bit redundancy of any text image in any language without the possibility of the watermark being observed by anyone. This applying proposed and tested here uses multiple layers of grid-line intersections, wherein the virtual intersection points on each layer of each text image created by horizontal (H), vertical (V) and double (H & V) grid lines run across the text-character skeleton lines and blank areas and are designated as watermark embedding positions. This technique aims to create as many layers and grid-line intersections points as the user requires to generate a sufficiently large hiding-bit quota, with one point for each bit embedded, to embed redundant secret watermark data bits and simultaneously verify their integrity or trace any changes in the text images. This objective depends on the cross-ratio theory of four collinear points, which has been effectively applied to control watermark embedding and detecting schemes on each grid-line intersection layer.

2 Mathematical Methods for Text-Image Watermarking

The two mathematical methods applied in this work are the cross-ratio theory, for marking and detecting the location of an embedded zero watermark or secret data, and the matching percentage method, used to verify an embedded zero watermark and its integrity.

2.1 Cross-Ratio of Four Collinear Points

The cross-ratio is a basic invariant in projective geometry (i.e., all other projective invariants can be derived from it). The cross-ratio invariance property is briefly introduced here.

Let A, B, C and D be four collinear points (three or more points A, B, C, are said to be collinear if they lie on a single straight line [13]), as in Fig. 1. The cross-ratio is the double ratio in Eq. (1).

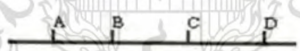


Figure 1: Collinear points; A, B, C and D

$$A,B;C,D = \frac{AC.BD}{BC.AD}, \quad (1)$$

where all the segments are considered signed. The cross-ratio does not depend on the selected direction of line ABCD but does depend on the relative positions of the points and the order in which they are listed. Based on a fundamental theory, any homography preserves the cross-ratio; central projection, linear scaling, skewing, rotation, and translation all thus preserve the cross-ratio [14].

2.2 The Matching Percentage

The matching percentage is the percentage of the number of watermark embedded positions that exactly matches the watermark detected position, one-to-one, based on the total number of watermark embedded positions, as in Eq. (2).

$$MPW = \frac{\sum D_{wm}(x_i, y_i)}{\sum P_{ip}(x_i, y_i)} \times 100, \quad (2)$$

where MPW is the matching percentage of the watermark embedded position detection, which corresponds to the selected intersection point for embedding the watermark. D_{wm} is the detected watermark at position $(x_i\ ,\ y_i)$ and P_{ip} is the prominent intersection points at position $(x_i\ ,y_i)$ which were selected to embed the watermark.

3 A New Grid-line and Cross-Ratio Applying for Digitally Watermarking a Text Image

There are three steps for arranging the zero watermarks on a text image. The first step is to define the zero watermark embedding positions for hiding secret data and their hiding-bit redundancy using certain intersection points on the horizontal (H), vertical (V) or both (H & V) grid lines that run across text-character skeleton lines and blank areas. The second step is to match the cross-ratio reference points on specific line intersection points with zero watermarking positions to easily track the zero watermarking points and extract the embedded secret data after a text image has been compromised. The last step is to detect the specific zero watermarked points and verify their integrity using the matching percentage method.

Definition 3.1 *C_aC_b is a line from the origin (C_a) to a destination point (C_b), where a = 1, 2, 3, 4, b = 1, 2, 3, 4 and*

Cr = (CA/CD) : BA/BD = (CA/CD) (BD/BA),

R = (AC/CD)/Cr and

*BA = (AD * R)/(1 + R).*

DsB is the distance of BA/AD, which is equal to the value of BA/DA.

3.1 Multi-Layer Watermark Embedding Technique

3.1.1 Create the Watermark Redundancy with Multiple Layers of Grid-Line Intersections

A large watermark redundancy volume in a text image can be effectively created using three layers of grid-line intersection of the horizontal (H), vertical (V) and double (H & V) grid lines that cross one another under a specific pattern, as in Fig. 2.

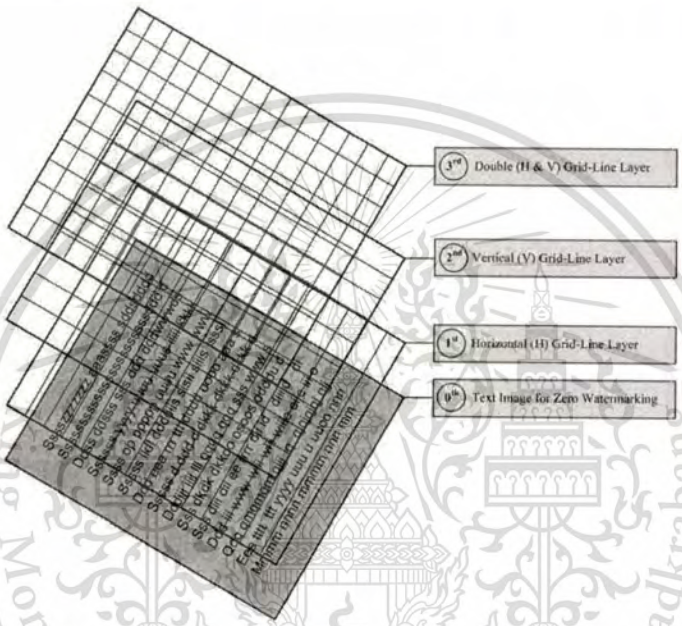


Figure 2: Three grid-line intersection layers of horizontal (H), vertical (V) and double (H & V) grid lines for embedding a large volume of watermark secret data in each layer.

These three grid-line intersection layers are designed to generate relatively large watermark hiding-bit redundancies for each text-image page. In our experiments, we tested 35 pages of Thai, English, Chinese and Arabic language text images by counting all grid-line intersection points in all layers, where one intersection point encodes one bit of watermark hiding data.

3.1.2 Transform the Watermarked Secret Data into Binary and Embedding Positions

After the redundant watermark data bits are completely specified, the next steps are followed as described below.

1) Define the secret data to be embedded as the watermark on each redundant bit of each layer.

2) Transform the watermarked secret data, i.e., the word COPYRIGHT, into binary, i.e., the bits 0 and 1.

3) Select the major intersection points or their locations, $(x_i\ ,\ y_i)$, corresponding to specifics bits (1 or 0) of the transformed secret data, as shown in Fig. 3 and listed in Table 2, to assign the watermark embedding positions.

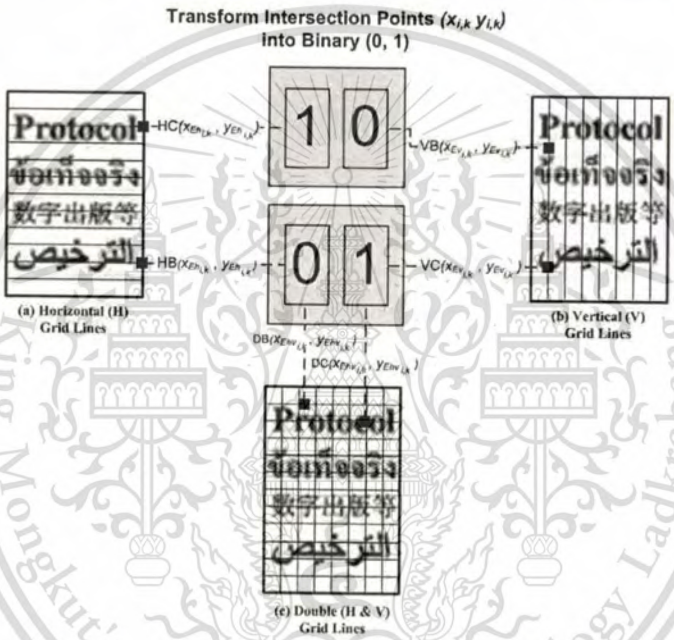


Figure 3: The technique for transforming intersection points $(x_i\ ,\ y_i)$, running across text-character skeleton lines and blank areas with (a) horizontal (H), (b) vertical (V) and (c) double (H & V), grid lines, into binary or bit, 0, 1, data to create the watermark-hiding bit redundancy for each layer.

Table 2: The mapping between watermark data bit values (0 or 1) and the intersection points of the horizontal (H), vertical (V) and double (H & V) grid lines that run across character skeleton lines and blank areas on each text-image page.

Layer No.	Type of Grid Line	Variable Name and Intersection Point (Position): Across Character (C) Skeleton Line	Variable Name and Intersection Point (Position): Across Blank (B) Area	Watermark Data Bit Value
1st	Horizontal (H)	HC (x_i, y_i)		1
			HB (x_i, y_i)	0
2nd	Vertical (V)	VC (x_i, y_i)		1
			VB (x_i, y_i)	0
3rd	Double (H & V)	DC (x_i, y_i)		1
			DB (x_i, y_i)	0

3.1.3 Embed Watermarks under Cross-Ratio Directing

In this experiment, the watermark embedding technique is actually based on two sub-processes. The first sub-process matches the cross-ratio reference points with zero watermark marking positions to easily track the zero watermark marking points after a text image has been attacked. The second sub-process detects the zero watermark and verifies the integrity of the text image using the matching percentage method. Applying the cross-ratio to digital image watermarking requires three reference points. This section details a method for deriving such reference points.

We begin by considering the marking component. The method is described algorithmically, step-by-step, below.

1. Predefine the set of cross-ratio values to be used in subsequent steps.
2. Find the image center using the line intersection formula [15] (two diagonal lines of the image) described by Eqs. (3) - (4) below.

$$x_c = x_t/x_b, \tag{3}$$

$$y_c = y_t/y_b, \tag{4}$$

$$x_t = \begin{vmatrix} a & x_1 - x_4 \\ b & x_3 - x_2 \end{vmatrix}, x_b = \begin{vmatrix} x_1 - x_4 & y_1 - y_4 \\ x_3 - x_2 & y_3 - y_2 \end{vmatrix},$$

$$y_t = \begin{vmatrix} a & y_1 - y_4 \\ b & y_3 - y_2 \end{vmatrix}, \; y_b = \begin{vmatrix} x_1 - x_4 & y_1 - y_4 \\ x_3 - x_2 & y_3 - y_2 \end{vmatrix},$$

where

$$a = \begin{vmatrix} x_1 & y_1 \\ x_4 & y_4 \end{vmatrix} and\; b = \begin{vmatrix} x_3 & y_3 \\ x_2 & y_2 \end{vmatrix}.$$

In addition, $(x_i\ , y_i)$ is the coordinate of the point C_i, $i = 1, \ldots, 4$ (Fig. 4 (a)). From the above equations, x_c is the x-axis value of the point D_c of a two-line intersection; C_1C_4 intersects C_2C_3, and y_c is the y-axis value of the same point. | | denotes a determinant operator, as shown in Fig. 4 (a).

3. Find each of the primary-level watermark marking points (TCr_1 and TCr_2) on the right diagonal line (Fig. 4 (a)), as described by Eqs. (5) - (6) below. These points can be identified after using two corner points of the left diagonal line (C_2 and C_3) in combination with the image center point D_c and the predefined cross-ratio values (C_r).

$$x_{TCr_i} = x_2 + DsB \times (x_3 - x_2), \quad (5)$$

$$y_{TCr_i} = y_2 + DsB \times (y_3 - y_2), \quad (6)$$

where $(x_{TCr_i}\ , y_{TCr_i})$, $i = 1, 2$, $A = C_2$, $B = TCr_i$, $C = D_c$ *and* $D = C_3$.

4. For each pair of coordinates $(x_1\ , y_1)$ $(x_2\ , y_2)$, find an intersection, $(x_i\ , y_i)$, of the grid line of each level drawn across $(x_3\ , y_3)$ $(x_4\ , y_4)$ by applying Eqs. (7) - (8).

$$x_i = x_t/x_b, \quad (7)$$

$$y_i = y_t/y_b, \quad (8)$$

$$x_t = \begin{vmatrix} a & x_1 - x_2 \\ b & x_3 - x_4 \end{vmatrix}, \; x_b = \begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix},$$

$$y_t = \begin{vmatrix} a & y_1 - y_2 \\ b & y_3 - y_4 \end{vmatrix}, \; y_b = \begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix},$$

where

$$a = \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} and\; b = \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix}.$$

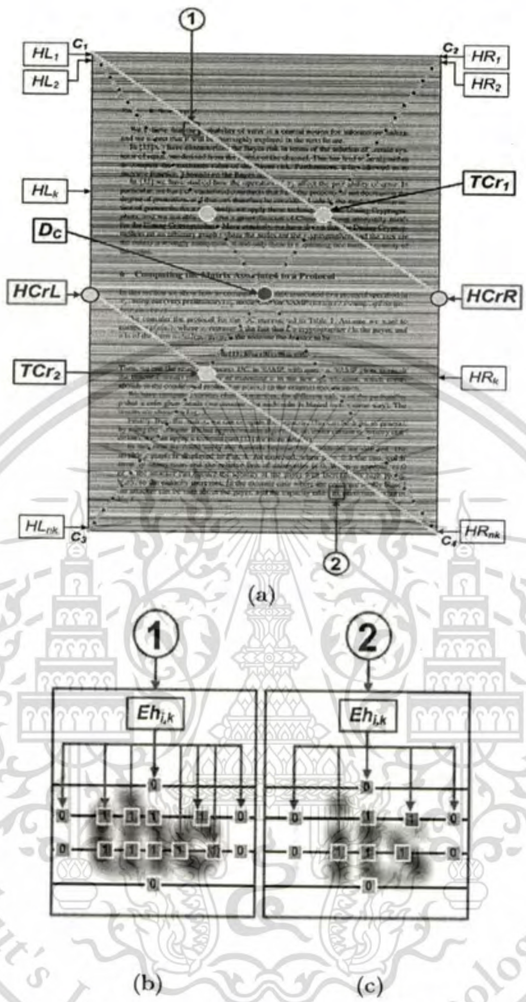


Figure 4: (a) Notation for the intersection of the horizontal grid lines and two diagonal lines in a text image; (b) and (c) notation for the intersection of the virtual horizontal grid lines and text character lines or blank areas to form collinear points used to embed or mark zero watermark pattern bits onto the text image.

For the line C_4TCr_2, find a left-hand horizontal grid-line intersection, (x_{HCrL}, y_{HCrL}), of this line drawn across the line C_1C_3, where $(x_1, y_1) = (x_{TCr_2}, y_{TCr_2})$, $(x_2, y_2) = (x_{C_4}, y_{C_4})$, $(x_3, y_3) = (x_{C_3}, y_{C_3})$, and $(x_4, y_4) = (x_{C_1}, y_{C_1})$. For the line C_1TCr_1, find a right-hand horizontal grid-line intersection, (x_{HCrR}, y_{HCrR}), of this line drawn across the C_4C_2, where $(x_1, y_1) = (x_{C_1}, y_{C_1})$, $(x_2, y_2) = (x_{TCr_1}, y_{TCr_1})$, $(x_3, y_3) = (x_{C_4}, y_{C_4})$, $(x_4, y_4) = (x_{C_2}, y_{C_2})$.

5. Find each of the secondary level watermark marking points (HL_k) on the left-hand horizontal grid line, as described by Eqs. (9) - (10) below. These points can be identified after using two corner points of the line C_1C_3, in combination with the image left-hand horizontal central point ($HCrL$) and the predefined cross-ratio values (C_r), as shown in Figs. 4 (a), (b) and (c).

$$x_{HL_k} = x_1 + DsB \times (x_3 - x_1), \tag{9}$$

$$y_{HL_k} = y_1 + DsB \times (y_3 - y_1), \tag{10}$$

where $A = C_1$, $B = (x_{HL_k}, y_{HL_k})$, $C = HCrL$ and $D = C_3$.

6. Find each of the secondary level watermark marking points (HR_k) on the right-hand horizontal grid line, as described by Eqs. (11) - (12) below. These points can be identified after using two corner points of the line C_2C_4 in combination with the image right-hand horizontal central point ($HCrR$) and the predefined cross-ratio values (C_r), as shown in Figs. 4 (a), (b) and (c).

$$x_{HR_k} = x_2 + DsB \times (x_4 - x_2), \tag{11}$$

$$y_{HR_k} = y_2 + DsB \times (y_4 - y_2), \tag{12}$$

where $A = C_2$, $B = (x_{HR_k}, y_{HR_k})$, $C = HCrR$ and $D = C_4$.

7. Repeat Steps 4 through 6 to find other secondary-level watermark marking points $(Eh_{i,k})$ at the prominent horizontal grid-line intersection positions where each horizontal grid line runs across a text-character line or blank area and its tone is lower or higher than the 245 grayscale level when more effective embedding and detection is required. These prominent positions are specified as the embedded bits, one or zero, of invisible zero watermarks $(Eh_{i,k})$ when their tones are lower than 245 or higher than 245, respectively, as indicated in Figs. 4 (b) - (c).

8. Find each of the secondary level watermark marking points (VU_i) on the upper-part vertical grid line, as described by Eqs. (13) - (14) below. These points can be identified after using two corner points of the line C_1C_2 in combination with the image upper-part vertical central point ($VCrU$) and the predefined cross-ratio values (C_r), as shown in Figs. 5 (a), (b) and (c).

$$x_{VU_i} = x_1 + DsB \times (x_2 - x_1), \tag{13}$$
$$y_{VU_i} = y_1 + DsB \times (y_2 - y_1), \tag{14}$$

where $A = C_1$, $B = (x_{VU_i}, y_{VU_i})$, $C = VCrU$ and $D = C_2$.

9. Find each of the secondary level watermark marking points (VD_i) on the lower-part vertical grid line, as described by Eqs. (15) - (16) below. These points can be identified after using two corner points of the line C_3C_4 in combination with the image lower-part vertical central point ($VCrD$) and the predefined cross-ratio values (C_r), as shown in Figs. 5 (a), (b) and (c).

$$x_{VD_i} = x_3 + DsB \times (x_4 - x_3), \tag{15}$$
$$y_{VD_i} = y_3 + DsB \times (y_4 - y_3), \tag{16}$$

where $A = C_3$, $B = (x_{VD_i}, y_{VD_i})$, $C = VCrD$ and $D = C_4$.

10. For each vertical grid line, detect other prominent intersection positions of each vertical grid line that runs across each text-character skeleton line or blank area, where its tone is lower or higher than the 245 grayscale level when more effective embedding and detection is needed. These detected prominent positions are specified as the embedded bits, one or zero, of invisible zero watermarks ($Ev_{i,k}$) when their tones are lower or higher than 245, respectively, as indicated in Figs. 5 (b) - (c).

11. For each pair (x_{VU_i}, y_{VU_i}) and (x_{VD_i}, y_{VD_i}), find an intersection, $(x_{Evh_{i,k}}, y_{Evh_{i,k}})$, of the grid line of each level drawn across (x_{HL_i}, y_{HL_i}) and (x_{HR_i}, y_{HR_i}), as shown in Figs. 6 (a), (b) and (c), after applying Eqs. (17) - (18).

$$x_{Evh_{i,k}} = x_t/x_b, \tag{17}$$

$$y_{Evh_{i,k}} = y_t/y_b, \tag{18}$$

$$x_t = \begin{vmatrix} a & x_1 - x_2 \\ b & x_3 - x_4 \end{vmatrix}, \; x_b = \begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix},$$

$$y_t = \begin{vmatrix} a & y_1 - y_2 \\ b & y_3 - y_4 \end{vmatrix}, \; y_b = \begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix},$$

where

$$a = \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix} \text{ and } b = \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix}.$$

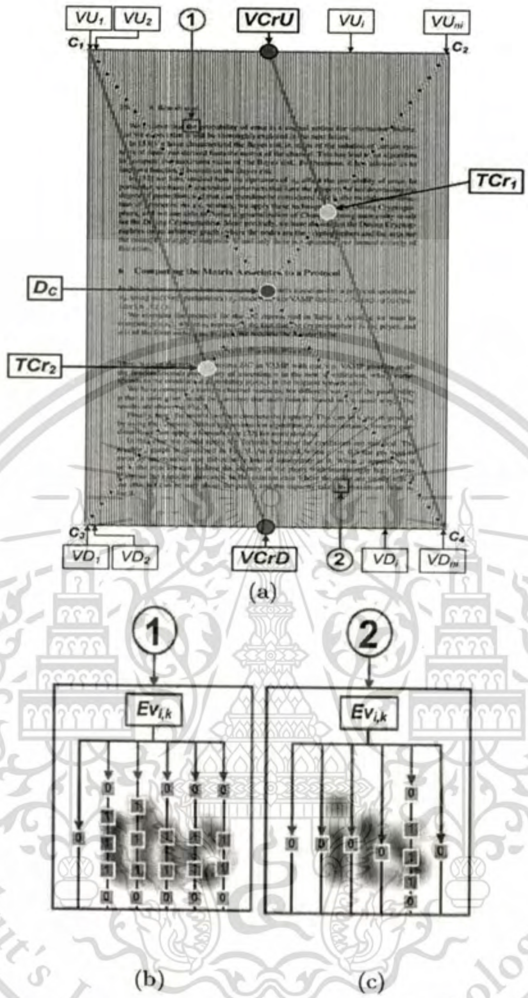


Figure 5: (a) Notation for the intersection of the vertical grid lines and text character lines in a text image; (b) and (c) notations for the intersection of the virtual vertical grid lines and text character lines or blank areas to form collinear points used to embed or mark zero watermark pattern bits in the text image.

For each pair (x_{VU_i}, y_{VU_i}) and (x_{VD_i}, y_{VD_i}), find an intersection, $(x_{Evh_{i,k}}, y_{Evh_{i,k}})$, of the grid line of each level drawn across (x_{HL_k}, y_{HL_k}) and (x_{HR_k}, y_{HR_k}), $(x_1, y_1) = (x_{VU_i}, y_{VU_i})$, $(x_2, y_2) = (x_{VD_i}, y_{VD_i})$, $(x_3, y_3) = (x_{HL_k}, y_{HL_k})$, $(x_4, y_4) = (x_{HR_k}, y_{HR_k})$.

12. Determine the positions of each prominent intersection of the double (H & V) grid lines that run across each text-character line or blank area with tones lower or higher than the 245 grayscale level when more effective embedding and detection is needed. These prominent positions are specified as the embedded bits, one or zero, of invisible zero watermarks $(x_{Evh_{i,k}}, y_{Evh_{i,k}})$ when their tones are lower or higher than 245, respectively, as indicated in Figs. 6 (b) - (c).

3.2 Watermark Detection Technique

3.2.1 Detection of Watermark Bits at the Embedded Intersection Points

To detect a watermark in a text image, four image corner points must first be detected. This detection can be achieved using any existing corner detection algorithm. After detecting the four corner points, watermark marking points must be identified. Each selected intersection point used to embed a watermark data bit can be calculated using a method similar to that in the marking stage (Eq. (2)). By extracting and transforming the pixel bit values corresponding to these watermark marking points, a watermark can be detected using any existing watermark detector. We adopt the basic matching percentage method described in The matching percentage to compare these detected watermark positions with their originally marked positions, one-by-one, until all specified positions are covered. We predefine the set of cross-ratio values to be used in subsequent steps.

3.2.2 Translation of Watermark Bits into the Embedded Secret Data

After obtaining the extracted bit values of each redundant bit of each layer, the next step is translating them into characters or embedded watermark secret data and comparing them with the original embedded secret data. These comparisons are made bit-by-bit by checking the matching percentage.

3.2.3 Verification of Text Image Integrity and Change Detection

If the extracted data do not match the original embedded secret data, the unmatched bits lead us to their positions, (x_i, y_i), where we can check for modifications, i.e., new text additions, reordering or deletions. This helps the copyright owners to trace any changes in their original text images. This

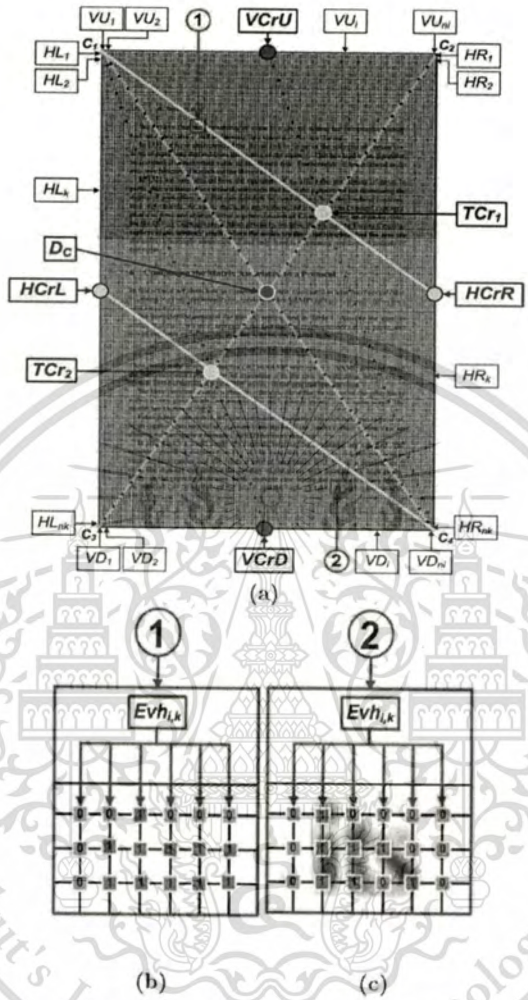


Figure 6: (a) Notation for the double (H & V) grid-line intersections in a text image; (b) and (c) notation for the double (H & V) grid line intersections with text character skeleton lines or blank areas used to embed or mark zero watermark pattern bits in the text image.

technique can also prevent copyright owners from accepting ownership of counterfeit text images that embedding their verified secret watermark data.

4 Experiments

4.1 Text Images Tested in Four Languages

We tested a set of 35 text images of grayscale multi-language text, including Thai, English, Chinese and Arabic, with a size of 1,240 x 1,754 pixels and a resolution of 150 dpi.

4.2 Testing the Number of grid-Line Intersection Layers for Hiding-Bit Redundancy

To obtain a high hiding-bit redundancy for watermarking, each page of these grayscale-text images, we first virtually created the three grid-line intersection layers, containing the horizontal (H), vertical (V) and double (H & V) grid-line layers, as in Figs. 4, 5 and 6. There are three total grid-line intersection layers for testing and contributing the watermark hiding-bit redundancy.

4.3 Text Images Tested in Four Languages

In this test, the prominent grid-line intersection points created on each grid-line intersection layer described above, i.e., the points where the horizontal, single vertical or double (H & V) grid lines run across the text-character skeleton lines and blank areas, were selected to embed the watermark secret data bits. One selected prominent intersection point represented one embedded data bit with cross-ratio controlling. Here, the word COPYRIGHT was encoded in the secret data bits to be embedded in each grid-line intersection layer. Each instance of COPYRIGHT requires 72 bits; 010000110100111101010000010110 010101001001001001010001110100100001010100.

4.4 Testing for Robustness against Manipulation Attacks on Embedded Watermark Detection

Three text-group manipulations, including 10 text groups with addition, re-ordering and deletion manipulations, were used to test the embedded watermark detection and integrity verification scheme. The percentage of matching was used as the criterion for this testing.

5 Results

The testing revealed that the double (H & V) grid line intersections generated a maximum hiding-bit redundancy volume of 240,608 bits/layer/A4 page of text image (Table 3), whereas the single horizontal and vertical grid-line intersections generated a maximum hiding-bit redundancy volume of only 75,329 bits/layer/A4 page of text image.

Table 3: The hiding-bit redundancy volume of each language and layer.

Layer No.	Type of Grid Line Intersection	Hiding-bit Redundancy (bits)			
		Thai	English	Chinese	Arabic
1st	Horizontal (H)	64,278	65,125	61,614	75,329
2nd	Vertical (V)	32,234	32,449	30,829	38,053
3rd	Double (H & V)	240,608	240,608	240,608	240,608
	Total	337,120	338,182	333,051	353,990

With respect to each language, this experiment demonstrated that Arabic text generates high hiding-bit redundancies under both single horizontal and vertical grid-line intersection layers (Table 3). The tested languages did not affect their hiding-bit redundancies under the double (H & V) grid-line intersection layer, as all intersection points, whether running across text-character skeleton lines or blank areas, were counted.

The above results describe the hiding-bit redundancies generated by three grid-line intersection layers, as in Table 3. To count the total hiding-bit redundancy per text image page using this technique, we would count all the hiding-bit redundancies in all layers, as in Table 3. Arabic script generates the maximum hiding-bit redundancy per page: up to 353,990 bits/A4 page of text image can be hidden.

After the three text manipulation tests, with 10 text groups added, reordered and deleted, each layer could be used to detect and extract the embedded watermark and verify its integrity or modification, measured as an error percentage, as summarized in Table 4.

Table 4: The performance of embedded watermark detection and integrity/change verification of each grid-line intersection layer after the text images were attacked using text-group addition, reordering and deletion.

Layer No.	Type of Grid Line	Avg. % of Error Detected After Attacking with 10 Text-Group Manipulations		
		Adding	Reordering	Deleting
1st	Horizontal (H)	0.0468	0.7549	1.3691
2nd	Vertical (V)	0.0508	0.7652	1.3725
3rd	Double (H & V)	0.0661	0.0924	0.0628

The results showed that the horizontal and vertical grid-line intersection layers are suitable for detecting the embedded watermark data, COPYRIGHT, and identifying their integrity after the tested changes under the 10 text-group reordering and deletion manipulation attacks, which detected error percentages up to 0.76% and 1.37%, respectively, equivalent to 99.24% and 98.63% matching percentages (Table 4). The double (H & V) grid-line intersection layer is better for detecting the embedded watermark data and identifying their integrity after changes under a 10 text-group addition manipulation attack, with 0.07% error and 99.93% matching percentages (Table 4).

6 Discussion

The above results demonstrate that the use of multiple grid-line intersection layers, based on the cross-ratio theory, can effectively increase text hiding-bit redundancy and simultaneously improve other watermarking performance. Applying the cross-ratio with grid-line technique, the hiding-bit redundancy depends on the number of intersection points that are virtually created in each layer by the horizontal, vertical and double (H & V) reference grid lines, text-character skeleton lines and blank areas. One virtual intersection point equates to one bit of secret watermark data embedding capacity.

In practice, every intersection point cannot be used; only the prominent points are selected by screening their grayscale levels, which must lower than 245, for a one bit, or higher than 245, for a zero bit. This selection allows the user to easily detect the points after facing manipulation attacks, e.g., word addition, reordering or deletion. Our results indicate that for a given language, the hiding-bit capacity varies depending on the number of virtual grid lines intersections created in each layer and the number of grid-line intersection layers created on each text image page. However, we found that the number of grid lines is limited when the gaps or interval spaces between each line are less than 3 pixels. This limitation occurs due to the grayscale measurement accuracy problem, which requires space or tolerance to clearly identify two adjacent points.

Another significant achievement of the grid-line intersection with cross-ratio applying technique is its integrity verification and change detection capability. As described above, although ownership copyright watermark data can be identified from text images, they are not necessarily the real or original watermarks. It is thus necessary to verify their integrity. Based on the results of this experiment, as reported in Table 4, the finely collected crossed-line intersection points can be used to simultaneously detect several changes or modifications in their original text images. The key success factors are the creation of as many intersection points throughout each layer of each text-image page as possible and then the collection of all the identified bits of these intersection points.

Any changes or modifications, i.e., added, reordered or deleted words, are easily detected under the applying of the cross-ratio theory. However, this capability depends on the type of grid-line intersection; our data show that the double (horizontal and vertical) grid-line intersection was the most effective for verifying added text. The intersections of double reference grid lines, e.g., vertical and horizontal grid lines, running across or intersecting one another on the blank area of a text image are used as reference points for blank areas to trap any additional text from the text-addition attack and precisely pinpoint the exact position of the new text. For text reordering and deletion, the single horizontal and vertical reference grid lines perform better than the double grid lines. This performance advantage is useful for approving the originality and integrity of sensitive study images including checks, land license certificates and significant trade contract documents.

7 Conclusions

It can be concluded that the cross ratio theory applying with the multiple grid-line intersection layer technique is a new breakthrough in text-image watermarking that overcomes the watermark redundancy increasing limitations of prior text-image watermarking techniques. This method also provides better integrity verification performance, a shortcoming of other techniques. With the cross-ratio theory of applying four collinear points to control their embedded positions, this grid-line intersection technique can also enhance the robustness of the embedded watermarks to allow for robust detection after an attack based on text manipulations.

ACKNOWLEDGEMENTS. This research was performed under the computer science doctoral study program of the Department of Computer Science, Faculty of Science, King Mongkuts Institute of Technology Ladkrabang, Bangkok, Thailand, where all lecturers and officers supported us in completing this research.

References

[1] Natthawut S., Matthew N. D., Steganography in Thai text, *5th International Conference on Electrical Engineering/Electronics, Computer Telecommunications and Information Technology, IEEE ECTI-CON 2008*, (2008), 133-136.

[2] Zunera J., Anwar M. M., A Review of Digital Watermarking Techniques for Text Documents, *2009 International Conference on Information and Multimedia Technology*, (2009), 230-234.

[3] Suganya R., Ahamed J. A., Kathirvel. K., Mohan K. M., Combined Text Watermarking, *International Journal of Computer Science and Information Technologies*, 1(5) (2010), 414-416.

[4] S. H. Low, N. F. Maxemchuk, J. T. Brassil, L. O'Gorman., Document marking and identification using both line and word shifting, *14th Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM95)*, 2 (1995), 853-860.

[5] Adnan M. A., Osama M. A., Watermarking electronic text documents containing justified paragraphs and irregular line spacing, *SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents VI*, 5306 (2004), 685-694.

[6] Wenyin Z., Zhenbin Z., Geguang P., Huibiao Z., Chinese Text Watermarking Based on Occlusive Components, *2nd Information and Communication Technologies*, (2006), 1850-1854.

[7] Young-Won K., Kyung-Ae M., Il-Seok O., A Text Watermarking Algorithm based on Word Classification and Inter-word space Statistics, *7th International Conference on Document Analysis and Recognition (ICDAR03)*, (2003), 775-779.

[8] Ding H., Hong Y., Interword Distance Changes Represented by Sine Waves for Watermarking Text Images, *IEEE Trans. Circuits and Systems for Video Technology*, 11(12) (2001), 1237-1245.

[9] Umut T., Mercan T., Mikhail J. A., The hiding Virtues of Ambiguity: Quantifiably Resilient Watermarking of Natural language Text through Synonym Substitutions, *8th workshop on Multimedia and Security*, (September, 2006), 164-174.

[10] M. Hassan S., Mohammad S., A New Approach to Persian/Arabic Text Steganography, *5th IEEE/ACIS International Conference on Computer and Information Science*, (2006), 310-315.

[11] Jack T. B., Steven L., Nicholas F. M., Lawrence O., Electronic Marking and Identification Techniques to Discourage Document Copying, *IEEE Journal on Selected Areas in Communications*, 13(8) (1995), 1495-1504.

[12] Min D., Quanyou Z., Text Watermarking Algorithm based on Human Visual Redundancy, *IAISS Journal, Advanced in Information Sciences and Service Sciences*, 3(5) (2011), 229-235.

[13] H. S. M. Coxeter, S. L. Greitzer, *Collinearity and Concurrence, Geometry Revisited*, Math. Assoc. Amer., 1967.

[14] R. Mohr, L. Morin, Relative Positioning from Geometric Invariants, Proceedings of the Conference on Computer Vision and Pattern Recognition, *IEEE Conference on Computer Vision and Pattern Recognition*, (1991), 139-144.

[15] Franklin A., *Faster Line Segment Intersection, Graphics Gems III*, Academic Press, 1999.

Received: June, 2012

ScienceAsia

A peer-reviewed journal jointly published by
THE SCIENCE SOCIETY OF THAILAND
AND THE NATIONAL RESEARCH COUNCIL OF THAILAND

Online access : http://www.scienceasia.org

วช.
NRCT

Volume 39 Number 4
August 2013
ISSN 1513-1874 CODEN: SCIASFZ
39(4): 331 – 447 (2013)

- The genus *Morinda* (Rubiaceae) in Thailand
- Organic acids and lauric arginate against *Listeria* and *Salmonella*
- An aubasidan-like β-glucan produced by *Aureobasidium pullulans*
- Accumulation of heavy metals in mangrove water, sediments, and crabs

RESEARCH ARTICLE
doi: 10.2306/scienceasia1513-1874.2013.39.423

ScienceAsia 39 (2013): 423–435

Grid-line watermarking: A novel method for creating a high-performance text-image watermark

Wiyada Yawai*, **Nualsawat Hiransakolwong**

Department of Computer Science, Faculty of Science, King Mongkut's Institute of Technology Ladkrabang, Chalongkrung Road, Ladkrabang, Bangkok 10520 Thailand

*Corresponding author, e-mail: wyd.yawai@gmail.com

Received 4 Jun 2012
Accepted 18 Jan 2013

ABSTRACT: This study aims to discover an effective method to watermark any text image in any language. The ultimate goal of this work is to generate invisible and more robust watermarks, increase the hiding capacity, and identify any change in the original text. Generally these are the limitations of most text-image watermarking methods. Using a grid of horizontal and vertical lines is one of the most effective methods to overcome these limitations. These grid lines run across text-image character-skeleton lines to finely mark and detect watermarks on these line-character intersection points; there is one intersection for one specific zero watermark pixel. Each intersection point is defined by one hiding bit of the watermark such that the more reference horizontal and vertical lines of grid pattern are plotted, the more points of intersection are obtained. This approach increases the hiding-bit capacity to watermark embedding, which is a disadvantage shared by many text-image watermarking methods. In addition, if all positions of all intersection points are collected, these points will act as the identity reference points of all original characters to verify their integrities after they are modified.

KEYWORDS: digital watermarking, text image, integrity, cross ratio, collinear points, grid lines, zero watermark

INTRODUCTION

State of problem

Currently, creating and embedding watermarks in a text image faces many limitations, particularly reduced robustness against a variety of attacks and a reduced capacity to hide a large volume of watermark data due to the nature of the content, language and text, which is mostly written in black. This approach provides a small variation space to hide the large volume of invisible watermark data, hence making the data easy to observe and destroy by simple attacks. Many researchers have proposed algorithms to hide large amounts of watermark data in a text image. However, these algorithms only apply to certain languages, only achieve a small hiding-bit data capacity and are not strong enough to survive attacks. Thus increasing the capacity for watermark hiding and robustness are the main goals of watermarking applications for storing the copyrighted data of all digital media and text documents. Unfortunately, these features are the most difficult aspects to address for black and white text images. In fact, if watermarks are made to be absolutely invisible and impervious to any attack or tampering, these difficulties will be doubled. For the latter, even though we may be able to detect our own copyright by extracting information using intact redundant watermarks, we have not necessarily obtained our original text image. In contrast, a copyright on a modified text image is achieved instead. This issue is the greatest challenge for text-image watermarking research.

Existing research

According to the various types of host media, digital watermarking may be classified into the following four categories: image watermarking, video watermarking, audio watermarking and text watermarking. The principles of image watermarking, video watermarking and audio watermarking are similar in that they make use of the redundant information of their host media, but text watermarking techniques are different from those of non-text watermarking. Moreover, text watermarking algorithms cannot easily satisfy the requirements of transparency (i.e., invisibility or imperceptibility) and robustness[1].

Most watermarking studies have concluded that text watermarking is the most difficult type of steganography, largely due to the relative lack of redundant information in a text file when compared with a picture or a sound file[2]. One reason that text steganography is difficult is that text contains little redundancy compared to other media. Another reason is that humans are sensitive to abnormal text[3].

Text files present various challenges for copyright protection. Any text transformation should preserve the meaning, fluency, grammar, writing style and value of the text. Short documents have a low capacity for watermark embedding and are relatively difficult to protect. Text watermarking algorithms are also dependent on the text size, language, rules, grammar, conventions and writing style. In the past, text watermarking methods based on text images, synonyms, pre-supposition, syntactic-trees, nouns and verbs, word and sentences, acronyms and type errors[4].

The watermarking of text falls under two domains: (1) text-image watermarking and (2) natural language watermarking. The aim of both of these watermarking systems is the embedding of information by modifying the original data in a discrete manner such that the modifications are imperceptible and the embedded information is impervious to possible attacks. In image text watermarking, this goal is achieved by exploiting the redundancy in images and the limitations of the human visual system. However, language has a discrete and syntactical nature that makes such techniques more difficult to apply[5].

Our survey revealed that most of the existing studies are focused on watermarking an electronic text or document file, rather than a text image, in one specified language, as opposed to multiple languages, and stress the watermark embedding technique over watermark robustness, not the integrity verification. These existing text watermarking studies can be categorized into three methods as follows (see Table 1).

First method: Watermark embedding with text document physical layout/pattern/structure rearrangement. Specific examples of this method include the shifting of lines[4,6,7] and words, particularly the binding of word spacing, word shift coding or word classification[8,9]. This method has some disadvantages. For example, the line shifting technique of Low et al[6] would be weakly robust for a document passing through document processing.

Second method: Embedding text watermark by text character/letter feature modification. For example, Du et al[11] proposed a text digital watermarking algorithm based on a slight change in text colour; watermarks were embedded after changing the low 4 bits of the RGB colour components of the characters. However, this study tested only the robustness of this algorithm against word deletion and modification. Brassil, et al[10] used letter adjusting by reducing or increasing the length of letters, such as increasing the lengths of the letters *b*, d, or h. The limitation of this process is that the hidden data will be weakly robust for a document passing through document processing.

Third method: Watermarking with semantic schemes or word/vowel substitution and text structure. Topkara et al[13] developed a technique for embedding secret data without changing the meaning of the text after replacing words in the text with synonyms. This method reduces the quality of a document, and a large synonym dictionary is necessary. Samphaiboon et al[3] proposed a steganographic scheme for electronic Thai plain text documents that exploits redundancies for a particular vowel in the standard Thai character set. However, this method can be used with only Thai text documents, and these watermarks are easily destroyed after re-editing with a word processing program. Meanwhile, structural schemes of text watermarking use text structures to embed watermarks without modifying any original text, so called the zero watermarking[14,15]. For instance, the existence of double letter (aa-zz) in the text is used to watermark the copyright data. However, the structural algorithms are not applicable to all types of text documents and the algorithm use an alphabetical watermark[12].

Research objectives

This study aims to discover a novel method to solve the previously stated problems and to eliminate the weakest points of the currently used methods. In doing so, the cross-ratio theory will be applied with intersections of the virtual reference grid lines and character-skeleton lines to build an effective watermark performance including robustness against possible attacks, hiding-bit data capacity, imperceptivity, multi-language text image applying and text integrity verification. Thus the five objectives of this work are the following:

(1) To find a new method to effectively embed watermarks in any text image in any language.

(2) To make watermarks absolutely invisible or difficult to detect.

(3) To obtain a higher hiding-bit data capacity.

(4) To make the watermark more robust to possible attacks.

(5) To identify any change in an original text image and verify its integrity.

MATHEMATICAL METHODS FOR TEXT WATERMARKING

The two mathematical methods applied in this study are the cross-ratio theory to build marking robustness and detect the embedded zero watermark location and the matching percentage method to verify embedded zero watermarks and the integrity of the text.

The cross ratio is a basic invariance in projective geometry (i.e., all other projective invariances can be

Table 1 Performance comparison of the existing watermarking method.

Method	Technique	Applied to	Imperceptivity	Language	Approx. Capacity	Robustness
first	Text line shifting[4,6,7]	Digital text document, text image (300–400 dpi)	Visible	Multi-language	30 bits/A4 page	Weakly robust to document processing or OCR
	Inter-word space shifting (to left - right)[8,9]	Digital text document, text image	Visible	Multi-language	720 bits/A4 page	Weakly robust to document processing or OCR
second	Point shifting of letter 'i' & 'j'[5]	Text image	Visible	English	600 bits/A4 page	Weakly robust to brightness and noise signal adding
	Adjusting character size[1]	Digital text document, text image	Visible	Chinese	650 bits/A4 page	Weakly robust to document processing or OCR
	Character peak point distinction[2]	Text image	Visible	Persian and Arabic	900 bits/A4 page	Weakly robust to brightness and noise signal adding
	Reduce/increase length of letter for feature coding[10]	Text image	Visible	English	4000 bits /A4 page	Weakly robust to document processing or OCR
	Changing the low 4 bits of RGB colour of characters[11]	Digital plain text	Invisible	Chinese	40 000 bits /A4 page	Robust to deletion modification attack etc.
third	Existence of double letter (aa–zz) in the text[12]	Digital plain text	Invisible	English	50 bits/A4 page	Not applicable to all types of text document
	Particular vowel creating sequent changing[3]	Digital plain text	Invisible	Thai	58 bits/A4 page	Very weakly robust to document processing
	Substitute words in the text by synonyms[13]	Digital plain text	Invisible	English	870 bits/A4 page	Very weakly robust to document processing

derived from it). Let A, B, C and D be four collinear points (three or more points; A, B, C, are said to be collinear if they lie on a single straight line[16]). The cross ratio is the 'double ratio' which is given by $A,B;C,D = AC.BD/BC.AD$. Based on a fundamental theory, any homography preserves the cross ratio. Thus central projection, linear scaling, skewing, rotation, and translation preserve the cross ratio[17].

The matching percentage MPW is the percentage of the amount of the watermark embedded position, which exactly matches the watermark detected position, one-to-one, based on the total amount of the watermark embedded positions:

$$\text{MPW} = \frac{\sum D_{\text{wm}}(x_i, y_i)}{\sum P_{\text{ip}}(x_i, y_i)} \times 100\%.$$

D_{wm} is the detected watermark at position (x_i, y_i) and P_{ip} is the prominent intersection points at position (x_i, y_i) which were selected to embed the watermark.

A NOVEL TECHNIQUE FOR DIGITALLY WATERMARKING TEXT IMAGES

Three steps for arranging the zero watermarks on a text image exist. The first step is to define the zero watermark embedding positions, for hiding secret data, and its hiding capacity, with certain intersection points where either horizontal or vertical grid lines, or both of grid lines run across the text-character skeleton lines. The second step is to match the cross-ratio reference points on specific line intersection points with zero watermarking positions to easily track the zero watermarking points after a text image has been attacked. The last step is to detect the specific zero

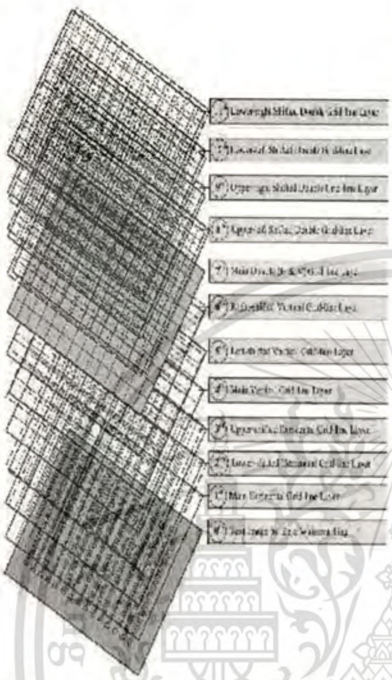


Fig. 1 Eleven grid-line intersection layers of main and 2-pixel-shifted horizontal (H), vertical (V) and double (H & V) lines for embedding a large volume of watermark secret data in each layer.

watermarked points and verify the integrity of the text with the matching percentage method.

Line C_aC_b runs from an origin point (C_a) to a destination point (C_b), where $a = 1, 2, 3, 4$ and $b = 1, 2, 3, 4$ and $C_r = ((CA/CD)/(BA/BD)) = (CA/CD)(BD/BA)$, $R = ((AC/CD)/C_r)$ and $BA = (AD \times R)/(1 + R)$. Q is the distance of $BA : AD$, which is equal to $BA : DA$.

Embedding Scheme: Create watermark embedding positions and hiding-bit capacity with multiple layers of grid-line intersections

The points for embedding zero watermarks and increasing the watermark hiding-bit capacity can be achieved by effectively created using multiple layers of grid-line intersection of the main and shifted horizontal (H), vertical (V) and double (H & V) lines that cross one another under a specific pattern, as in Fig. 1. Each layer is created by following the procedure below.

Step 1: Construct the first, second and third layers of main, lower-shifted and upper-shifted horizontal grid-line intersections by identifying and drawing the first reference horizontal lines for the top borders of the first text line on each text image page. Draw the second main, lower-shifted and upper-shifted horizontal lines next to the first main, lower-shifted and upper-shifted horizontal lines, the third and the rest until the last text line, at 3-pixel intervals, which is a suitable gap for a geometric variation buffer, as found in a previous experiment, on the first, second and third layers, respectively.

Step 2: Create the fourth, fifth and sixth layers of main, left-shifted and right-shifted vertical grid-line intersections by identifying the right and left borders of the text body on a text image, and draw the first main, left-shifted and right-shifted vertical lines along the left border of the fourth, fifth and sixth layers, respectively. Draw the second main, left-shifted and right-shifted vertical lines, the third and the rest until reaching the end of the right border, using a 3-pixel interval.

Step 3: Create the seventh, eighth, ninth, tenth and eleventh layers by drawing both the main, upper-left, upper-right, lower-left and lower-right shifted horizontal (H) and vertical (V) lines, respectively, e.g., the main double (H & V) lines, using a 3-pixel interval, on the same text-image page to create the seventh main double-line intersection layer, where both the main horizontal and vertical lines run across either text-character skeleton lines or blank areas and create intersection points for embedding watermark data bits.

Step 4: Select each prominent grid-line intersection position in each main or shifted line that runs across each text-character skeleton line and blank area, as shown in Fig. 2, for the embedding watermark data. If the tone of the grid-line intersection is less than the 245 greyscale level or is a gray/black tone, its bit value is specified as 1. If its greyscale value is higher than the 245 greyscale level or is a white tone, the bit value is specified as 0. These selected prominent white and black positions are specified as the marking points of the zero digital watermark bits for hiding secret data, and these points represent the main

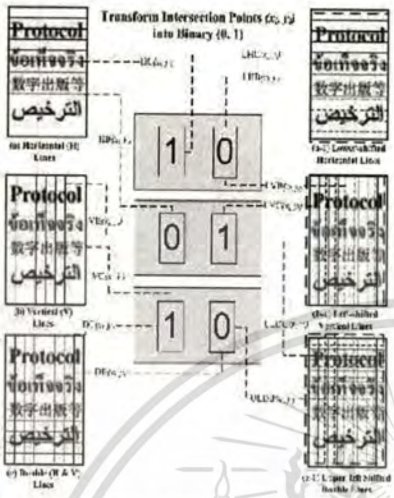


Fig. 2 The technique for transforming intersection points (x_i, y_i), running across text-character skeleton lines and blank areas with (a) horizontal (H), (b) vertical (V), (c) double (H & V), (a-1) lower-shifted horizontal, (b-1) left-shifted vertical and (c-1) upper-left shifted double H & V lines, into binary or bit, 0, 1, data to create the watermark-hiding bit capacity for each layer.

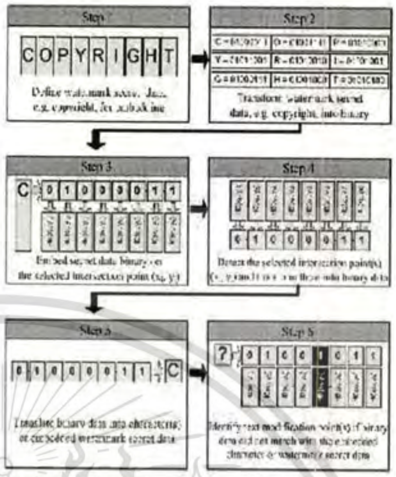


Fig. 3 The six steps for embedding and detecting watermark secret data, i.e., 'COPYRIGHT', which is transformed into binary data (0, 1) before being embedded into the selected reference line intersection points. The watermark is then again detected by checking whether the embedded intersection positions exist.

hiding-bit capacities of the first, second, third, fourth, fifth, sixth, seventh, eighth, ninth, tenth and eleventh grid-line layers.

Step 5: If more hiding-bit capacity is required in each layer, the horizontal, vertical and double (H & V) lines can be shifted to new positions, and new shifted horizontal, vertical and double (H & V) lines can be drawn to create new intersection points that run across the text-character skeleton lines and blank areas. These shifts create more intersection points at new positions to provide more hiding-bit capacity. Moreover, these new intersection points, surrounding their main original intersection points, can be used as crosscheck points to identify the exact point of origin and verify any changes to the integrity of the text image.

These 11 grid-line intersection layers are designed to generate relatively large watermark hiding-bit capacities for each text-image page. In our experiments, we tested 35 pages of Thai, English, Chinese and Arabic language scanned text images by using our MATLAB program to draw automatically grid lines and counting all grid-line intersection points in all layers, where one intersection point encodes one bit of watermark hiding data. This way, it can automatically watermark a lot of scanned text papers.

Transform the watermarked secret data into binary and embedding positions

To transform the watermarked secret data into binary and embedding positions can be done as described below.

Step 1: Define the secret data to be embedded as the watermark on each layer.

Step 2: Transform the watermarked secret data, i.e., the word COPYRIGHT, into binary, i.e., the bits 0 and 1, as illustrated in Figs. 3 and 4.

Step 3: Select the major intersection points or their locations, (x_i, y_i), corresponding to specifics bits (1 or 0) of the transformed secret data to assign the watermark embedding positions.

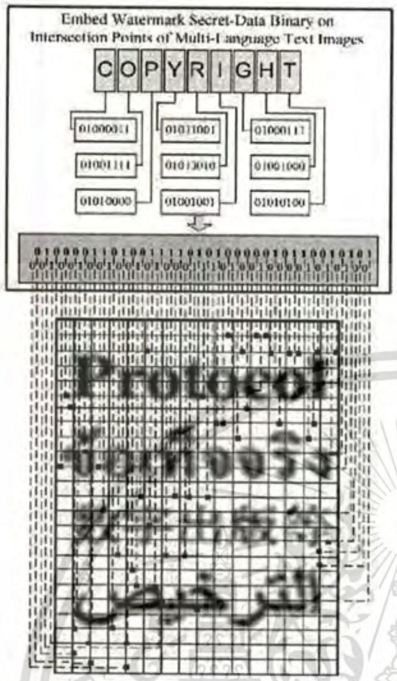


Fig. 4 The technique for embedding secret watermark data bits (0, 1) into the selected grid-line intersection positions.

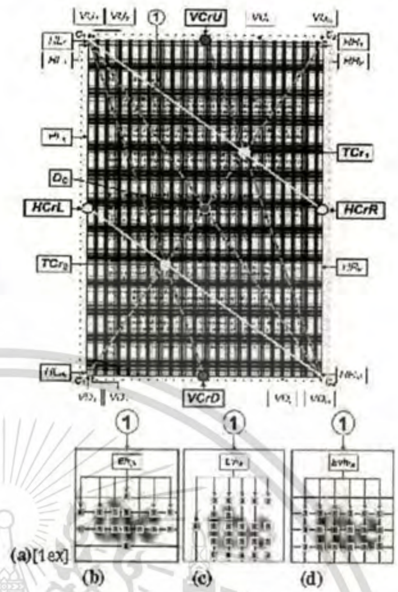


Fig. 5 (a) Notation of double reference grid-line intersection of vertical and horizontal lines on a text image. (b), (c) and (d) Notation of the horizontal, vertical and horizontal & vertical grid lines, in the upper parts, respectively, intersect text character skeleton lines and blank areas used to embed or mark zero watermark pattern bits in the text image.

Embed watermarks under cross-ratio directing

The text watermarking process shown in Figs. 3 (Step 3) and 4 is based on two subprocesses. The first one is to match the cross-ratio reference points with zero watermark marking positions to easily track the zero watermark marking points after a text image has been attacked. The second one is to detect the zero watermarks and verify the integrity of the text with the matching percentage method.

To apply the cross ratio to digital image watermarking, three reference points are required. In this section, a method for deriving such reference points is described. Let us start by considering the marking part. The method is described algorithmically, step by step, as follows.

Step 1: Predefine the set of cross-ratio values to be used in subsequent steps.

Step 2: Find the image centre using the line intersection formula[18] (two diagonal lines of the image) described by the following equations: $x_c = x_t/x_b, y_c = y_t/y_b$;

$$x_t = \begin{vmatrix} a & x_1 - x_4 \\ b & x_3 - x_2 \end{vmatrix}, \quad x_b = \begin{vmatrix} x_1 - x_4 & y_1 - y_4 \\ x_3 - x_2 & y_3 - y_2 \end{vmatrix},$$

$$y_t = \begin{vmatrix} a & y_1 - y_4 \\ b & y_3 - y_2 \end{vmatrix}, \quad y_b = \begin{vmatrix} x_1 - x_4 & y_1 - y_4 \\ x_3 - x_2 & y_3 - y_2 \end{vmatrix},$$

where

$$a = \begin{vmatrix} x_1 & y_1 \\ x_4 & y_4 \end{vmatrix}, \quad b = \begin{vmatrix} x_3 & y_3 \\ x_2 & y_2 \end{vmatrix}.$$

In addition, (x_i, y_i) is the coordinate of the point C_i, $i = 1, \dots, 4$ (Fig. 5a). From the above equations, x_c is the x-axis value of the point D_c

of a two-line intersection; C_1C_4 intersects C_2C_3, and y_c is the y-axis value of the same point. $||$ denotes a determinant operator (Fig. 5a).

Step 3: Find each of the primary-level watermark marking points (TCr_1 and TCr_2) on the right diagonal line (Fig. 5a), using: $x_{TCr_i} = x_2 + Q(x_3 - x_2)$, $y_{TCr_i} = y_2 + Q(y_3 - y_2)$, where (x_{TCr_i}, y_{TCr_i}), $i = 1, 2$, $A = C_2$, $B = T_{Cr_i}$, $C = D_c$ and $D = C_3$. These points can be identified after using two corner points of the left diagonal line (C_2 and C_3), in combination with the image centre point D_c and the predefined cross-ratio values (C_r).

Step 4: For each pair of coordinates $((x_1, y_1)(x_2, y_2))$, find an intersection, (x_i, y_i), of the grid line of each level drawn across $(x_3, y_3)(x_4, y_4)$ by applying the following equations: $x_i = x_t/x_b$, $y_i = y_t/y_b$;

$$x_t = \begin{vmatrix} a & x_1 - x_2 \\ b & x_3 - x_4 \end{vmatrix}, \quad x_b = \begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix},$$

$$y_t = \begin{vmatrix} a & y_1 - y_2 \\ b & y_3 - y_4 \end{vmatrix}, \quad y_b = \begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix},$$

where

$$a = \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix}, \quad b = \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix}.$$

For the line C_4TCr_2, find a left-hand horizontal grid-line intersection, (x_{HCrL}, y_{HCrL}), of this line drawn across the line C_1C_3, where $(x_1, y_1) = (x_{TCr_2}, y_{TCr_2})$, $(x_2, y_2) = (x_{C_4}, y_{C_4})$, $(x_3, y_3) = (x_{C_3}, y_{C_3})$, and $(x_4, y_4) = (x_{C_1}, y_{C_1})$. For the line C_1TCr_1, find a right-hand horizontal grid-line intersection, (x_{HCrR}, y_{HCrR}), of this line drawn across the C_4C_2, where $(x_1, y_1) = (x_{C_1}, y_{C_1})$, $(x_2, y_2) = (x_{TCr_1}, y_{TCr_1})$, $(x_3, y_3) = (x_{C_4}, y_{C_4})$, $(x_4, y_4) = (x_{C_2}, y_{C_2})$.

Step 5: Find each of the secondary level watermark marking points (HL_k) on the left-hand horizontal grid line, as described by $x_{HL_k} = x_1 + Q(x_3 - x_1)$, $y_{HL_k} = y_1 + Q(y_3 - y_1)$, where $A = C_1$, $B = (x_{HL_k}, y_{HL_k})$, $C = HCrL$ and $D = C_3$. These points can be identified after using two corner points of the line C_1C_3, in combination with the image left-hand horizontal central point ($HCrL$) and the predefined cross-ratio values (C_r).

Step 6: Find each of the secondary level watermark marking points (HR_k) on the right-hand horizontal grid line, as described by $x_{HR_k} = x_2 + Q(x_4 - x_2)$, $y_{HR_k} = y_2 + Q(y_4 - y_2)$, where $A = C_2$, $B = (x_{HR_k}, y_{HR_k})$, $C = HCrR$ and $D = C_4$. These points can be identified after using two corner points of the line C_2C_4 in combination with the image right-hand horizontal central point ($HCrR$) and the predefined cross-ratio values (C_r).

Step 7: Repeat Steps 4–6 to find other secondary-level watermark marking points ($Eh_{i,k}$) at the prominent horizontal grid-line intersection positions where each horizontal grid line runs across a text-character line or blank area and its tone is lower or higher than the 245 greyscale level when more effective embedding and detection is required. These prominent positions are specified as the embedded bits, one or zero, of invisible zero watermarks ($Eh_{i,k}$) when their tones are lower than 245 or higher than 245, respectively, as indicated in Fig. 5b.

Step 8: Find each of the secondary level watermark marking points (VU_i) on the upper-part vertical grid line, as described by $x_{VU_i} = x_1 + Q(x_2 - x_1)$, $y_{VU_i} = y_1 + Q(y_2 - y_1)$, where $A = C_1$, $B = (x_{VU_i}, y_{VU_i})$, $C = VCrU$ and $D = C_2$. These points can be identified after using two corner points of the line C_1C_2 in combination with the image upper-part vertical central point ($VCrU$) and the predefined cross-ratio values (C_r).

Step 9: Find each of the secondary level watermark marking points (VD_i) on the lower-part vertical grid line, as described by $x_{VD_i} = x_3 + Q(x_4 - x_3)$, $y_{VD_i} = y_3 + Q(y_4 - y_3)$, where $A = C_3$, $B = (x_{VD_i}, y_{VD_i})$, $C = VCrD$ and $D = C_4$. These points can be identified after using two corner points of the line C_3C_4 in combination with the image lower-part vertical central point ($VCrD$) and the predefined cross-ratio values (C_r).

Step 10: For each vertical line, detect other prominent intersection positions of each vertical line that runs across each text-character skeleton line or blank area, where its tone is lower or higher than the 245 greyscale level when more effective embedding and detection is needed. These detected prominent positions are specified as the embedded bits, one or zero, of invisible zero watermarks ($Ev_{i,k}$) when their tones are lower or higher than 245, respectively, as indicated in Fig. 5c.

Step 11: For each pair (x_{VU_i}, y_{VU_i}) and (x_{VD_i}, y_{VD_i}), find an intersection, $(x_{Evh_{i,k}}, y_{Evh_{i,k}})$, of the grid line of each level drawn across (x_{HL_i}, y_{HL_i}) and (x_{HR_i}, y_{HR_i})

after applying the followings equations: $x_{Evh_{i,k}} = x_t/x_b$, $y_{Evh_{i,k}} = y_t/y_b$;

$$x_t = \begin{vmatrix} a & x_1 - x_2 \\ b & x_3 - x_4 \end{vmatrix}, \quad x_b = \begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix},$$

$$y_t = \begin{vmatrix} a & y_1 - y_2 \\ b & y_3 - y_4 \end{vmatrix}, \quad y_b = \begin{vmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_4 & y_3 - y_4 \end{vmatrix},$$

where

$$a = \begin{vmatrix} x_1 & y_1 \\ x_2 & y_2 \end{vmatrix}, \quad b = \begin{vmatrix} x_3 & y_3 \\ x_4 & y_4 \end{vmatrix}.$$

For each pair (x_{VU_i}, y_{VU_i}) and (x_{VD_i}, y_{VD_i}), find an intersection, $(x_{Evh_{i,k}}, y_{Evh_{i,k}})$, of the grid line of each level drawn across (x_{HL_k}, y_{HL_k}) and (x_{HR_k}, y_{HR_k}), $(x_1, y_1) = (x_{VU_i}, y_{VU_i})$, $(x_2, y_2) = (x_{VD_i}, y_{VD_i})$, $(x_3, y_3) = (x_{HL_k}, y_{HL_k})$, $(x_4, y_4) = (x_{HR_k}, y_{HR_k})$.

Step 12: Determine the positions of each prominent intersection of the double (H & V) lines that run across each text-character line or blank area with tones lower or higher than the 245 greyscale level when more effective embedding and detection is needed. These prominent positions are specified as the embedded bits, one or zero, of invisible zero watermarks ($Evh_{i,k}$) when their tones are lower or higher than 245, respectively, as indicated in Fig. 5d.

DETECTION SCHEME

Detection of watermark bits at the embedded intersection points

To detect a watermark in a text image, first a watermarked paper has to be scanned and then its scanned text-image output would be passed through our MATLAB program to detect its four image corner points. This detection can be achieved using any existing corner detection algorithm. After detecting the four corner points, watermark marking points must be identified. Each selected intersection point used to embed a watermark data bit can be calculated using a method similar to that in the marking stage, as in Step 4 of Fig. 3. By extracting and transforming the pixel bit values corresponding to these watermark marking points, a watermark can be detected using any existing watermark detector. We adopt the basic matching percentage method described in "The matching percentage" to compare these detected watermark positions with their originally marked positions, one-by-one, until all specified positions are covered. We predefine the set of cross-ratio values to be used in subsequent steps.

Translation of watermark bits into the embedded secret data

After obtaining the extracted bit values of each layer, the next step is translating them into characters or embedded watermark secret data, as shown in Fig. 3 (Step 5), and comparing them with the original embedded secret data. These comparisons are made block-by-block or layer-by-layer by checking the matching percentage.

Verification of text image integrity and change detection

If the extracted data do not match the original embedded secret data, the unmatched bits lead us to their positions, (x_i, y_i), where we can check for modifications, i.e., new text additions, reordering or deletions, as in Fig. 3 (Step 6). This helps the copyright owners to trace any changes in their original text images. This technique can also prevent copyright owners from accepting ownership of counterfeit text images that embedding their verified secret watermark data. This situation may occur when the owners have applied a physical watermarking technique like inter-word or line shifting, which some hackers may observe and know how to manipulate by not altering the watermark embedding positions. The correctly watermarked data are still detected and extracted, although they have been modified. This false acceptance would rarely occur using this new technique because all text characters are finely marked, stored and detected, so a small bit change can eventually be identified.

EXPERIMENTS

Text images tested in multiple languages

We tested a set of 35 text images of greyscale multi-language text, including Thai, English, Chinese and Arabic, with a size of 1240×1754 pixels and a resolution of 150 dpi.

Testing the number of grid-line intersection layers for imperceptivity and hiding-bit capacity

To obtain a high hiding-bit capacity for watermarking, each page of these greyscale-text images, we first virtually created the three main grid-line intersection layers, containing the main horizontal (H), vertical (V) and double (H & V) grid-line layers, as in Fig. 1. Each layer was then derived into eight 2-pixel-shifted grid-line intersection layers, including two shifted horizontal grid-line intersection layers, two shifted vertical grid-line intersection layers and four shifted double (H & V) grid-line intersection layers. There are 11 total grid-line intersection layers for testing and contributing the watermark hiding-bit capacity.

Watermark secret data embedding test

In this test, the prominent grid-line intersection points created on each grid-line intersection layer described above, i.e., the points where the main and shifted single horizontal, single vertical or double (H & V) lines run across the text-character skeleton lines and blank areas, were selected to embed the watermark secret data bits. One selected prominent intersection point represented one embedded data bit with cross-ratio controlling. Here, the word 'COPYRIGHT' was encoded in the secret data bits to be embedded in each grid-line intersection layer, as in Figs. 3 and 4. Here, each instance of 'COPYRIGHT' requires 72 bits: 010 000 110 100 111 101 010 000 010 110 010 101 001 001 001 001 010 001 110 100 100 001 010 100.

Testing for robustness against manipulation attacks on embedded watermark detection and integrity verification

The grid-line cross-ratio watermarking robustness against some possible attacks such as compression, sharpness, noise signal adding, shearing and rotating and three text-group manipulations, including ten text groups with addition, reordering and deletion manipulations, were used to test the embedded watermark detection and integrity verification scheme. The percentage of matching was used as the criterion for this testing.

RESULTS

The first result was from the embedding and detection testing of the controlled watermarked text images without manipulation attack. Comparing the plotted watermark positioning pattern and extracting the embedded secret data, this method obtained a 100% matching percentage and a zero error percentage. For each of the following tests, we briefly describe the result according to our objectives.

Language-independent testing

The results of this test clearly demonstrated that this multiple grid-line intersection layer technique can be applied to all Thai, English, Chinese and Arabic text images. This technique requires only the grid-line intersection points where the virtual reference lines run across text-character skeleton lines and blank areas to embed watermark data bits without physically modifying the original text images or using any special technique for any specific language. Each tested language affects only the number of intersection points created due to the different character skeleton line structures.

Imperceptivity testing

Virtual lines are drawn according to these grid lines only in the program process, and they are not drawn on each text image page. Thus it would not immediately be possible to observe the lines and watermarks after these text images were completely watermark embedded, even with many secret data bits. Conversely, the pages look exactly like their original text images before watermarking.

Hiding-bit capacity testing

The testing revealed that the 7th, 8th, 9th, 10th and 11th layers of the main and shifted double (H & V) line intersections generated a maximum hiding-bit capacity volume of 240 608 bits/layer/A4 page of text image (see Table 2), whereas the first, second, third, 4th, 5th and 6th layers of the main and shifted single horizontal and vertical line intersections generated a maximum hiding-bit capacity volume of only 77 334 bits/layer/A4 page of text image.

With respect to each language, this experiment demonstrated that Arabic text generates high hiding-bit capacities under both single horizontal and vertical grid-line intersection layers (see Table 2). The tested languages did not affect their hiding-bit capacities under the double (H & V) grid-line intersection layer, as all intersection points, whether running across text-character skeleton lines or blank areas, were counted.

The above results describe the hiding-bit capacities generated by eleven main and shifted grid-line intersection layers, as in Table 2. To count the total hiding-bit capacity per text image page using this technique, we would count all the hiding-bit capacities in all layers. Arabic script generates the maximum hiding-bit capacity per page: up to 1 545 738 bits/A4 page of text image can be hidden.

Robustness, copyright ownership and integrity verification testing

After the manipulation tests, with compression, sharpness, noise signal adding, shearing and rotating and ten text groups added, reordered and deleted, each layer could be used to detect and extract the embedded watermark and verify its integrity or modification, measured as an error percentage, as shown their higher performance results in Figs. 6, 7, 8, 9 and 10 and summarized in the robustness testing results below, comparing with the performance of the existing text-image watermarking methods summarized in Table 1.

Table 2 The hiding-bit capacity volume of each language and layer.

Layer No.	Type of grid-line intersection	Hiding-bit capacity (bits)			
		Thai	English	Chinese	Arabic
1	Main Horizontal (H)	64 278	65 125	61 614	75 329
2	Lower-shifted H.	67 244	64 327	61 541	75 802
3	Upper-shifted H.	61 829	64 457	61 893	77 334
4	Main Vertical (V)	32 234	32 449	30 829	38 053
5	Left-shifted V.	32 188	32 204	30 978	38 048
6	Right-shifted V.	32 253	32 302	30 717	38 132
7	Main Double (H & V)	240 608	240 608	240 608	240 608
8	Upper-left Shifted Double (H & V)	240 608	240 608	240 608	240 608
9	Upper-right Shifted Double (H & V)	240 608	240 608	240 608	240 608
10	Lower-left Shifted Double (H & V)	240 608	240 608	240 608	240 608
11	Lower-right Shifted Double (H & V)	240 608	240 608	240 608	240 608
	Total	1 431 237	1 493 904	1 480 612	1 545 738

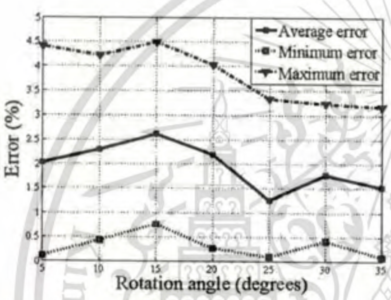


Fig. 6 Error level 4% of single horizontal reference grid-line zero watermark detection after rotation angles ranging from 5–35°.

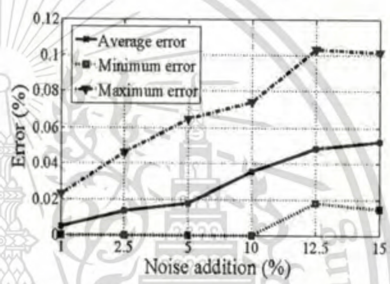


Fig. 7 Low error (%) level of single horizontal reference grid-line zero watermark detection after adding salt and pepper noise signals.

Figs. 6, 7 and the robustness testing results below show the remarkable performance of the cross-ratio theory in strengthening the robustness of text-image watermarking against some possible geometric distortion attacks, including shearing and rotating and manipulating attacks, such as noise signal adding, compression and sharpness. These attacks are the result of applying the single horizontal reference grid-line watermarking method while controlling the watermark data embedding and detecting using the cross ratio of four collinear points on each the horizontal reference grid lines that run across each text character skeleton line. The results show that these attacks did not significantly affect watermark detection. The rotating attack, for example, affected the embedded zero watermark by detecting only 0.15–4% of errors. This error indicates that the matching percentage of watermark detected is generally > 96%.

The robustness testing results are as follows. Sharpness: 0–100%; low error (0.5–2%) - JPEG compression: 0–100%; low error (0–2%) - blur: 3x3 - 15x15 mask size; low error (0.083%) - contrasting: 10–50%; no error - noise signal adding:pepper; low error (0–0.11%) - shearing:x right shifted 0–5%; low error (0–2%) - rotation: 5–35°; error level: 0.15–4% - scaling: 10–130%; error level: 0–6% - text adding detection: by double (H & V) lines; error Level: 0.008–0.067% - text deleting detection: by horizontal (H) lines; error Level: 0.25–1% - text reordering detection:by horizontal (H) lines; error Level: 0.18–0.82%.

Figs. 8, 9, 10 and the robustness testing results above present the ability of horizontal and vertical grid line intersection with cross-ratio theory to strengthen

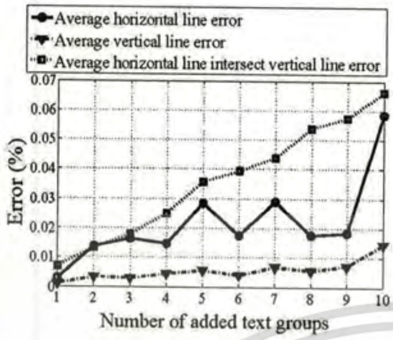


Fig. 8 The percent of errors, under the double reference grid-line intersection watermarking method and text adding manipulations attack, were more effectively detected than when using single reference grid-line watermarking.

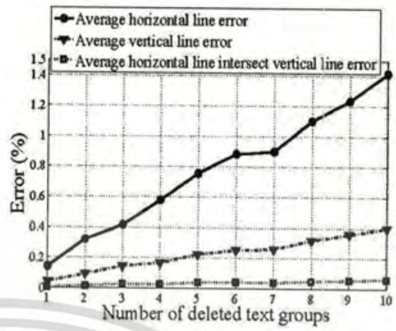


Fig. 10 The single reference grid-line intersection watermarking of the horizontal grid line was mostly suitable for detecting the percent of errors after being attacked with text deletion.

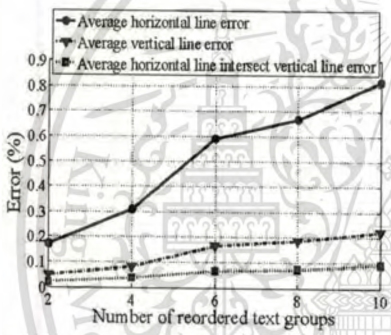


Fig. 9 The single reference grid-line intersection watermarking of the horizontal grid line was mostly suitable for detecting percent of errors after being attacked with text reordering.

the original text verification performance against three manipulations: text adding, reordering and deleting. The results indicate that the horizontal and vertical grid line intersection method with cross-ratio theory can still sensitively detect even a few changes, such as adding, reordering and deleting one to ten text groups, for which the observed maximum errors are 0.067%, 0.82% and 1% respectively.

These results show that the percent error of detection depends on the exact positions selected for zero watermark embedding or data collecting. For example, the intersected positions of the double reference, vertical and horizontal grid lines, which run across or intersect one another on the blank area of a text image, could be used as blank reference points to trap any added text from the text adding attack. This is why the double reference grid lines are more effective than the single reference grid line, (Fig. 8). However, for text reordering and deleting, the horizontal single reference grid-line intersection watermarking method performs better than the other techniques (Figs. 9 and 10).

DISCUSSION

The results of all of the experiments mentioned above prove that the novel method, which applies the cross-ratio theory and grid-line intersection of the virtual horizontal and vertical grid lines that run across text-character skeleton lines and blank areas, could lead to a major breakthrough in overcoming all of the major limitations of text image watermarking (see Table 1 and the robustness testing results above). Thus all five objectives of this study have been achieved, and the following discussions have been drawn.

(1) This method can be applied to watermark any text image in any language with the grid-line intersection points of either horizontal or vertical reference grid-lines or both grid lines virtually intersected the text-character skeleton lines or blank areas, which are used as the embedding points of watermark data. This novel technique does not depend on the type of text

images used or language, even though these experiments tested for only four languages: Thai, English, Chinese and Arabic. Nevertheless, the method can be applied to other languages because this novel technique does not require any text alignment or syntactic or semantic structural modifications.

(2) This method can be applied to make watermarks absolutely invisible or unobservable because no actual colour-tone pixel of a watermark is embedded and no physical text structure is modified, as mentioned above. The intersection points used as reference embedding points of watermark data are virtual horizontal and vertical grid lines that run across all text-character skeleton lines and blank areas but cannot be seen. These grid-line intersection points can be used as the reference positions of data bits to combine them as the embedding patterns of secret watermark data, zero watermarks, and can also be used as reference points to later detect and interpret the combined secret data bits.

(3) This method is able to generate a high hiding-bit data capacity of up to 1545738 bits/A4 page of Arabic text image, which safely and effectively creates bit combination redundancy to embed a large amount of secret watermark data on one A4 page of the text image. This enormous hiding-bit capacity is obtained from the number of intersection points of virtual reference grid lines and text-character skeleton lines, on each layer, where one intersection point is defined as one bit of secret watermark data. Hence, an increase in the number of grid-line intersection points indicates an increase in the amount of hiding bits obtained. However, our results show that the most suitable interval between each line is 3 pixels because the method requires some clear space for buffering and tolerating variations under the attack of some geometric distortions. Nevertheless, we can still dramatically increase the hiding-bit capacity by adding more watermarking grid-line intersection plains such as 12 layers of diagonal grid-lines.

(4) This method can be applied to make watermarks more robust to or more likely to survive many possible attacks, especially geometric distortions such as scaling, shearing and rotating. Specifying the cross ratio of four collinear points for each reference horizontal and vertical grid line and each diagonal line is first set to control and direct the embedding position of each secret watermark data bit (Fig. 4), which makes it easy to refer to the embedded position, even though they may be attacked. Figs. 6, 7 and the robustness testing results above confirm that the cross ratio could generate a watermark that is reasonably robust to both geometric distortions, as mentioned earlier, and graphical manipulations, such as sharpness, compression, blur, contrast and the addition of noise signals. This robustness is due to the cross-ratio marking pattern being able to precisely lock all watermark embedding positions to easily and exactly detect and retrieve them after they have been attacked.

(5) This method can be applied to simultaneously identify any change in the original text image and verify the image's integrity during watermark detection. This performance is directly related to the increase in the hiding-bit capacity. This relationship indicates that the more hiding bits or intersection points we have, the more checkpoints for available for verifying the image's integrity. Using this technique, the horizontal and vertical grid line intersection points on the text-character skeleton lines can be used as check points for text deleting and reordering, while their grid-line intersection points on the blank area can be used as the checkpoints for text adding. From these experiments, horizontal and vertical grid-line intersections were shown to be very effective in verifying the addition of new text (Fig. 8). This result is because the intersected positions of these double reference grid lines, vertical and horizontal grid lines, run across or intersect one another in the blank area of a text image, which is used as a reference point and can trap any additional text introduced by a text adding attack and precisely pinpoint the exact positions of text additions. Meanwhile, for text reordering and deleting, a single horizontal reference grid line would perform better than a single vertical reference grid line and double reference grid lines (Figs. 9 and 10).

Thus the cross-ratio theory and grid-line intersection are able to effectively generate high text-image watermarking performance.

CONCLUSIONS

Although this study involved many experiments, many ways remain to apply and benefit from these grid-line intersections and cross-ratio theory, especially for obtaining greater hiding-bit capacity by applying further pattern grid-line intersections including diagonal, sine-curve and unique barcode lines. The real identity of the original text image is also proven by directing the watermark embedding points using the relative vector along the grid-line intersection points to more efficiently verify original text modifications and enhance the robustness of this scheme. Two double reference grid-line intersection points may be selected, and lines may be virtually drawn to connect them, storing their relative vector data to form a unique web or mesh network of virtual watermarking points, locking them using the cross-ratio format for each text document image. This unique mesh network acts as a watermarking pattern fingerprint that can effectively be applied for both integrity verification and watermark retrieval after an attack. Our future work will focus on these advanced techniques.

Acknowledgements: This study was performed under the computer science doctoral study program of the Department of Computer Science, Faculty of Science, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand, where all lecturers and officers supported us in completing this study.

REFERENCES

1. Zhang W, Zeng Z, Pu G, Zhu H (2006) Chinese Text Watermarking Based on Occlusive Components. In: *Proceedings of the 2nd Information and Communication Technologies*, pp 1850–4.
2. Shirali-Shahreza MH, Shirali-Shahreza M (2006) A New Approach to Persian/Arabic Text Steganography. In: *Proceedings of the 5th IEEE/ACIS International Conference on Computer and Information Science*, pp 310–5.
3. Samphaiboon N, Dailey MN (2008) Steganography in Thai text. In: *Proceedings of the 5th International Conference on Electrical Engineering/Electronics, Computer Telecommunications and Information Technology*, pp 133–6.
4. Jalil Z, Mirza AM (2009) A review of digital watermarking techniques for text documents. In: *Proceedings of the International Conference on Information and Multimedia Technology*, pp 230–4.
5. Ranganathan S, Ali AJ, Kathirvel K, Mohan KM (2010) Combined text watermarking. *Int J Comput Sci Inform Tech* 1, 414–6.
6. Low SH, Maxemchuk NF, Brassil JT, O'Gorman L (1995) Document marking and identification using both line and word shifting. In: *Proceedings of the 14th Annual Joint Conference of the IEEE Computer and Communications Societies*, pp 853–60.
7. Alattar AM, Alattar OM (2004) Watermarking electronic text documents containing justified paragraphs and irregular line spacing. In: *Proceedings of the SPIE International Conference on Security, Steganography, and Watermarking of Multimedia Contents VI* vol 5306, pp 685–94.
8. Kim Y, Moon K, Oh I (2003) A text watermarking algorithm based on word classification and inter-word space statistics. In: *Proceedings of the 7th International Conference on Document Analysis and Recognition (ICDAR'03)*, pp 775–9.
9. Huang D, Yan H (2001) Interword distance changes represented by sine waves for watermarking text images. *IEEE Trans Circ Syst Video Tech* 11, 1237–45.
10. Brassil JT, Low S, Maxemchuk NF, O'Gorman L (1995) Electronic marking and identification techniques to discourage document copying. *IEEE J Sel Area Comm* 13, 1495–504.
11. Du M, Zhao Q (2011) Text watermarking algorithm based on human visual redundancy. *Adv Inform Sci Service Sci* 3, 229–35.
12. Jalil Z, Mirza AM, Iqbal T (2010) A zero-watermarking algorithm for text documents using structural components. In: *Proceedings of the International Conference on Information and Emerging Technologies (ICIET 2010)*.
13. Topkara U, Topkara M, Atallah MJ (2006) The hiding virtues of ambiguity: quantifiably resilient watermarking of natural language text through synonym substitutions. In: *Proceedings of the 8th workshop on Multimedia and Security*, pp 164–74.
14. Jaseena KU, Anita J (2011) An invisible zero watermarking algorithm using combined image and text for protecting text documents. *Int J Comput Sci Eng* 3, 2265–72.
15. Jalil Z, Farooq M, Zafar H, Sabir M, Ashraf E (2010) Improved zero text watermarking algorithm against meaning preservation attacks. In: *Proceedings of the World Academy of Science, Engineering and Technology*, pp 593–7.
16. Coxeter HSM, Greitzer SL (1967) Collinearity and concurrence. In: *Geometry Revisited*, Math. Assoc. Amer, pp 55–77.
17. Mohr R, Morin L (1991) Relative positioning from geometric invariants. In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp 139–44.
18. Antonio F (1999) Faster line segment intersection. In: *Graphics Gems III*, Academic Press.

BIOGRAPHY

PERSONNEL INFORMATION

Thai Name: นางสาววิยดา ยะไวทย์

English Name: Ms. Wiyada Yawai

Date of Birth: 14 May 1970

Permanent Address: 602/28 soi 9 Moo 6, Home Gardenvill, Ratanapitarn rd., Choho, Nakhon Ratchasima, 30310, Thailand.

Office Address: Faculty of Science and Technology, Nakhon Ratchasima Rajabhat University, Nakhon Ratchasima, 30000, Thailand.

Telephone: (+66) 080-6294717

E-mail: yawai.wiyada@gmail.com

EDUCATION

2006 **Studying in Doctor of Philosophy in Computer Science**
Faculty of Science,
King Mongkut's Institute of Technology Ladkrabang,
Bangkok, Thailand 10520.

2000 – 2005 **Master of Science in Information Technology**
Faculty of Information Technology,
King Mongkut's Institute of Technology Ladkrabang,
Bangkok, Thailand 10520.

1991 – 1993 **Bachelor of Education in Computer Education**
Faculty of Science and Technology,
Mahasarakham Teacher College,
Mahasarakham, Thailand 44000.