Print-Scan Resilient Text Image Watermarking Based on Stroke Direction Modulation for Chinese Document Authentication

Lina TAN [1,2], Xingming SUN [3,1], Guang SUN [4,1]*

[1] College of Information Science and Engineering, Hunan University, Changsha, 410082, China
[2] Information Department, Hunan University of Commerce, Changsha, 410205, China
[3] Jiangsu Engineering Center of Network Monitoring, Nanjing University of Information Science and Technology, Nanjing, 210044, China
[4] Information Management Department, Hunan University of Finance and Economics, Changsha, 410205, China

loebmail@126.com, *sunnudt@163.com, simon5115@163.com

Abstract. *Print-scan resilient watermarking has emerged as an attractive way for document security. This paper proposes a stroke direction modulation technique for watermarking in Chinese text images. The watermark produced by the idea offers robustness to print-photocopy-scan, yet provides relatively high embedding capacity without losing the transparency. During the embedding phase, the angles of rotatable strokes are quantized to embed the bits. This requires several stages of preprocessing, including stroke generation, junction searching, rotatable stroke decision and character partition. Moreover, shuffling is applied to equalize the uneven embedding capacity. For the data detection, denoising and deskewing mechanisms are used to compensate for the distortions induced by hardcopy. Experimental results show that our technique attains high detection accuracy against distortions resulting from print-scan operations, good quality photocopies and benign attacks in accord with the future goal of soft authentication.*

Keywords

Text image watermarking, print-photocopy-scan, stroke direction modulation, junctions.

1. Introduction

To date, a variety of important documents, such as social security records, insurance information, and financial documents, have been digitized and stored. Because of the ease to copy and edit digital images, digital forgery of text images is becoming more widespread and sophisticated. An increasing crime rate due to the counterfeiting or tampering of sensitive information creates an urgent need for protecting important documents against fraudulent alterations [1], [2]. Considering these requirements, approaches incorporating digital signatures or digital watermarking for tamper proofing are introduced. Cryptographic signatures may securely guarantee the integrity and authenticity of multimedia content. They reject any modifications to the media signals, which can be grouped under hard authentication [3]. However, in many applications, digital documents must be converted to hardcopies and vice versa. These manipulations are not accepted by hard authentication. Consequently, it brings about unprecedented opportunities of soft authentication, which passes certain incidental modifications such as compression, enhancement, scaling, and hardcopy operations.

Unlike other media, text documents show a marked trait that the foregrounds contrast clearly with the background areas, so that little information redundancy can be employed for embedding. Besides, printed documents suffer wear and tear from normal use and digitization, which disadvantages the message extraction. Recent developments of watermarking resilient to print-scan (PS for abbr.) or geometric attacks focus on continuous-tone images [4-6]. Nevertheless, the binarization operation, which is commonly applied to most of scanned and computer-generated text images, may destroy the watermarks produced by these methods. Thus, a different class of watermarking techniques for printed binary images, though difficult, is getting higher demands for practical considerations.

Watermarking on binary images can be carried out at different levels:

- Pixel-wise. This level deals with individual pixels where human perception is taken into account [7-10].
- Block-wise. The methods manipulate flippable pixels to modify some features of each image block, e.g., the parity or the percentage of white/black pixels in each block [11-14].
- Component-wise. These approaches handle the characteristics of some pixel groups such as strokes, connected components, characters, words, etc [15-19].

Level (1) and (2) mainly address the issue of hard authentication in electronic form and accordingly cannot resist the PS process. Most watermarking techniques for embedding robust copyright labels are component-wise, of level (3). This level often handles data with a relatively lower capacity compared to level (1) and (2). In many cases, the performance of the level (3) approaches highly depends on the state of the cover images.

There exists a variety of binary image watermarking techniques resilient to PS. The major attempts in the literature are on level (3). Earlier methods were based on line, word or character shifting [15, 16, 18, 20]. Although such schemes are reportedly robust to severe distortions introduced by processes such as PS and facsimile transmission, the hidden bit rates are pretty low for the message length is under the number of text lines, words or characters. A feature calibration scheme modifies the average stroke width along text line partitions [17]. Since it counterbalances the cumulative effects on the features of distortions caused by PS, the displacement can be reliably detected. Discouragingly, the problem of poor embedding capacity remains unsettled. A block-wise algorithm claimed to withstand high-resolution PS was developed by Wu et al [14]. It falls in a kind of visible registration using the marks to accurately determine the image boundary, skewing, and scaling due to PS. An alternative solution to authenticate a printed document by appending a visible 2-D barcode was proposed in [21]. This is achieved by using a digital signature on the character features that are relatively insensitive to PS. Nevertheless, the marks suffer from intentional removal for their visibility and might be a practical limitation for numerous applications. Kim et al [22] employed tiny unnoticeable dots to carry the information and rectified geometric distortions from PS. Despite an improved capacity, the technique is vulnerable to noise. Further, since the marks are embedded outside the text content, they can be attacked or deleted easily. A review of the above methods shows that data-hiding capacity tends to increase with reduced robustness, and vice versa. Approaches to improve both of them are still undergoing development.

Observe that few methods handle Chinese documents, which forms a large chunk of text data spread over the internet. We propose a watermarking method for Chinese hardcopy documents, which is intended to be applied to soft authentication. The robustness against hardcopy processes is guaranteed by the robustness of the stroke features of characters. Compared to the noise induced by hardcopy operations, strokes are lower frequency components that can be used to identify a character. It is these features that help human to distinguish a character with another. In another aspect, soft authentication doesn't require a highly robust data-hiding algorithm, since the intentional removal of the embedded data allows us to detect a security problem. By considering these facts, we design the method in following ways:

- Make rotatable stroke decision for angular modulation to achieve efficient trade-offs among watermark capacity (rate), the amount of embedding-induced distortion, and the robustness to unintentional attacks;
- Study the rotation-invariant feature of connected components to ensure blind watermark extraction;
- Introduce a shuffling program to tackle the uneven embedding capacity;
- Explore the compensation mechanisms for the distortions caused by incidental manipulations.

The rest of the paper is organized as follows. Section 2 is devoted to the proposed stroke direction modulation technique, which involves a series of stages with explicit details to settle the aforementioned issues. Section 3 presents experimental results and comparisons with other existing techniques. Section 4 is conclusions and future works.

2. Proposed Method

Our method manipulates "rotatable" strokes to unobtrusively hide the message. Data are embedded into individual characters by modifying the directional characteristic of rotatable strokes without causing noticeable artifacts. The entire process of watermark insertion and extraction is shown in Fig. 1 and described in detail in this section.

2.1 Preliminaries

2.1.1 Stroke Feature Extraction

A Chinese character can be partitioned into some basic components (refer to Def. 2.1). After converting the text image to be binary, all the basic components can be easily labeled throughout the document. In order to abstract the stroke features, we take two phases for each basic component, including stroke generation and junction searching. The outcomes of the two phases may be depicted as in Fig. 2 with an example of the character "太".

Definition 2.1 *Basic component*

A basic component is a connected component composed of several strokes, and it may be a Chinese character or a part of a Chinese character.

1) *Stroke Generation.* Our approach requires an efficient extraction of character strokes. Existing approaches [23, 24] usually take three stages: first obtaining character skeletonsby a thinning algorithm, then splitting the skeleton of a character into stroke segments by directional filtering, and finally adding a further refinement for wrong or inaccurate segmentation. Fig. 2(b), (c) and (d) present the results of using the above steps.

2) *Junction Searching.* In this phase, feature points are identified based on stroke structure analysis. There are six major types of feature points (also known as junctions) summarized in [23]. Multi-fork junctions (m-fork junctions, $1 \le m \le 6$), a vital sort of feature points, of an illustrational character are shown in Fig. 2(e).

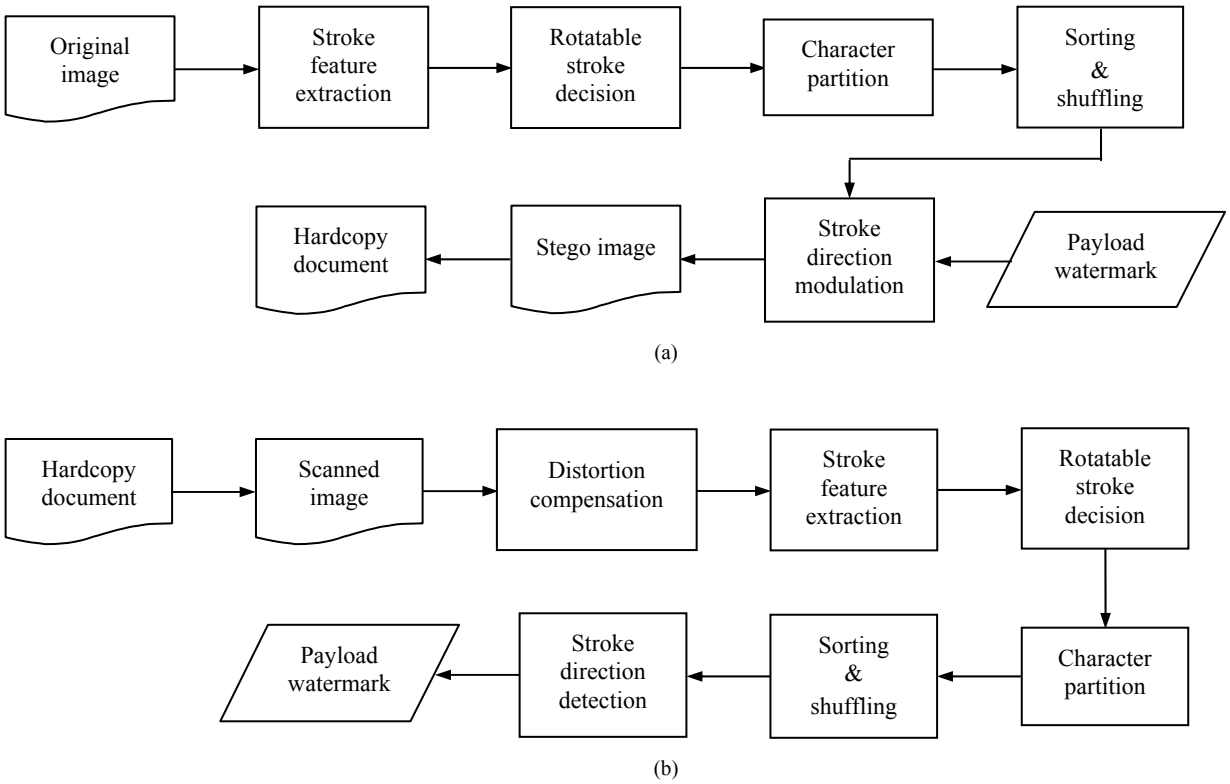


Fig. 1. Flow chart of the proposed system. (a) data embedding, (b) data detection.

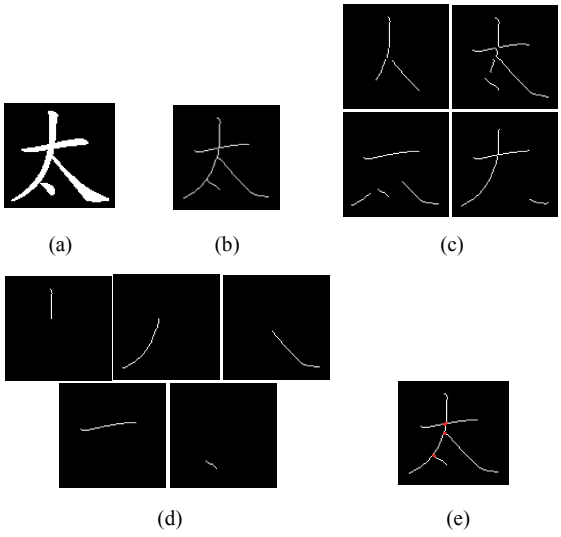


Fig. 2. Example of stroke feature abstraction. (a) original image, (b) skeleton image, (c) directional filtered images, (d) refined strokes, (e) skeleton image with multi-fork junctions (marked by red points).

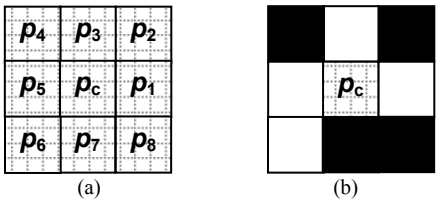
Broad classification of junctions into two main types is adequate for our method, namely the endpoints (1-fork) and the multi-forks (m-fork, $m > 1$). A model of "Cross Number" (CN) is formulated here for this problem. CN refers to the transitions of the eight neighbors around the considered pixel. We assume that the neighbors of the center pixel p_c in a 3×3 neighborhood are $p_1, p_2, \ldots, p_7$, and p_8, displayed in Fig. 3(a). Let's use "1" and "0" to be white and black pixels, respectively. Pixels in grid denote those whose values are free. CN of p_c is given by

$$CN(p_c) = \sum_{k=1}^{8} \left| p_{(k+1) \bmod 8} - p_k \right| / 2, \tag{1}$$

where $|\cdot|$ returns the absolute value. It can be seen that CN is independent of the center pixel value. Take Fig. 3(b) as an example, $CN(p_c) = 3$. Supposing that the junction type of p_c in the skeleton image is m-fork ($m = 0,\ldots,6$), we obtain

$$m \begin{cases} = 1, & if\ CN(p_c) = 1 \\ = 0\ or\ 2, & if\ CN(p_c) = 2 \\ > 2, & if\ CN(p_c) > 2 \end{cases}. \tag{2}$$

For $CN(p_c) = 1$, the pixel is considered as a candidate endpoint. If the candidates of a stroke exceed two, we qualify the farthest two as final endpoints. For $CN(p_c) > 2$, there is no exact mapping from $CN(p_c)$ to m, which matters little for our analysis since it surely serves as a multi-fork junction. For $CN(p_c) = 2$, p_c is either a 2-fork junction or else an ordinary point excluded from junctions (0-fork), which requires further confirmation. In this case, we detect 2-fork junctions by judging the associated endpoints. Those who are joined at a junction but on different strokes are called associated points [23]. Up to now, we meet the target for identifying all the feature points, which will play an important role for subsequent stages.



(a) (b)

Fig. 3. Illustration of a 3 × 3 neighborhood used for "Cross Number". (a) a 3 × 3 neighborhood marked with identifiers, (b) a 3 × 3 block pattern centered at p_c.

2.1.2 Rotatable Stroke Decision

Definition 2.2 *Stroke direction*

Stroke direction is the vector angle of a stroke pointing in the direction of its left-to-right endpoints. Let $EP1$, $EP2$ be the left and right endpoints of a stroke I, respectively. The angular function of I is defined based on the x- and y-coordinates on the image as

$$DT_I = \tan^{-1} \frac{y_{EP1} - y_{EP2}}{x_{EP1} - x_{EP2} + \varepsilon}, \tag{3}$$

where $\varepsilon \approx e^{-16}$ is a minimal positive number falling into $(0,1)$.

Definition 2.3 *Stroke magnitude*

Assume that h_I and w_I are the vertical and horizontal projection of a stroke I, respectively. The magnitude m_I of I is defined as

$$m_I = \sqrt{h_I^2 + w_I^2}. \tag{4}$$

Emphasis is placed on the decision of rotatable strokes. Suppose that a stroke I is derived from the skeleton of a basic component BC. Let H_I represent the average vertical profile value [18] of the relevant text line. I is considered to be a rotatable stroke when it satisfies all the restraints below.

(i) $|DT_I| \in \left(\theta, \frac{\pi}{2} - \theta\right)$, where $\theta \in (0°, 5°]$ is a selected angle.

(ii) If the number of multi-fork junctions on I is nonzero, $\lambda_1 \cdot \max(h_{BC}, w_{BC}) \le m_I \le \lambda_2 \cdot \max(h_{BC}, w_{BC})$, where $\max(\cdot)$ finds the largest value; otherwise $m_I \le \lambda_2 H_I$.

(iii) If there are more than one multi-fork junction on I, the maximal distance between these junctions is smaller than $\lambda_3 m_I$.

Within the above conditions, λ_1, λ_2, λ_3 are adjustable parameters subject to the character style, perceptibility and robustness, etc. Their values are discussed in the experimental section. As stated above, the condition (i) allows for the orientational influence of I. Either I approaching horizontal or vertical may be refused since any slight change to it is noticeable. (ii) indicates that I is eliminated when its magnitude is either too large or too small. The larger the magnitude of I is, the higher perceptibility it achieves when modified; whereas a much smaller value may reduce the accuracy of stroke segmentation after embedding. For a specific case of I without any multi-fork junction, its corresponding basic component must consist of only one stroke, namely I itself. Here the visual impact caused by modifying I relies on the whole text line where it presents. (iii) takes account of the case that I may be connected with at least two other strokes meeting at different junctions. If these junctions get far apart from each other, rotation of I may introduce junction distortion, skeleton deformation, or spurious branches.

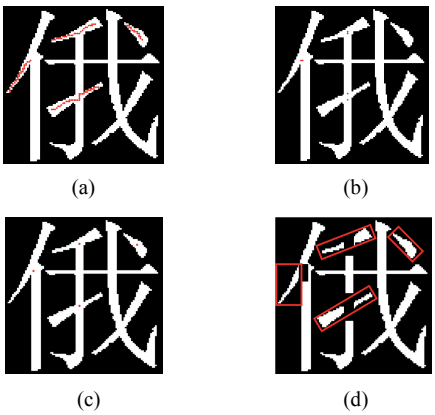
2.1.3 Character Partition

This stage is performed to extract the embeddable stroke pieces, which are the pixel groups covering a rotatable stroke but unconnected with other strokes, as shown in Fig. 4(d). All the rotatable strokes of a character "俄" are labeled in Fig. 4(a). The embeddable stroke pieces can be partitioned off from the basic components based on multi-fork junctions. In this way, we can merely modulate the rotatable strokes while keeping other strokes untouched.

The previous stages have found a way to locate all the multi-fork junctions and rotatable strokes. There may be one or more multi-fork junctions on a rotatable stroke, as described in Fig. 4(b). We define a rotating center on each rotatable stroke using the multi-fork junctions. Let NJ_I be the number of multi-fork junctions on a rotatable stroke I, and RC_I denote the rotating center of I, obtaining

$$RC_I = \begin{cases} \left[Centroid\left(I\right)\right], & if\ NJ_I = 0 \\ \Gamma, & if\ NJ_I = 1 \\ Median\left(\Gamma\right), & if\ NJ_I > 1 \end{cases}, \tag{5}$$

where Γ represents the set of multi-fork junctions on I; $[\cdot]$ rounds to the nearest integer; $Centroid(\cdot)$ measures the centroid coordinate; and $Median(\cdot)$returns the median value. The results are illustrated in Fig. 4(c).



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

Fig. 4. Character "俄" labeled with (a) rotatable strokes, (b) multi-fork junctions on rotatable strokes, (c) rotating centers on rotatable strokes, (d) embeddable stroke pieces (inside the red bounding boxes).

In order to explicate the extraction of embeddable stroke pieces from a basic component, we take Fig. 5 as a typical example. It illustrates the division of links between the embeddable pieces and others. An apt area surrounding the rotating center should be set black to get separate pieces for cutting off the links. By then finding the separate pieces where the rotatable stroke lies, the embeddable stroke pieces can be perfectly extracted. A detailed description is presented in Appendix A.

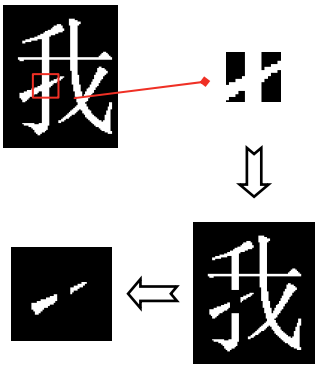


Fig. 5. Extraction of embeddable stroke pieces with an example.

2.2 Print-Scan Resilient Embedding

With the goal of providing a way of embedding bits to survive the PS process, we propose a quantization-based method on the stroke feature level. Let the orientation range $[-\pi/2, \pi/2]$ be discretized into a number of intervals. The bounded metric space is quantized using a series of uniform quantization lattices accordingly to the message symbol to be embedded ('0' or '1'), given by

$$\Lambda_0 = 2\Delta_\theta Z, \tag{6}$$

$$\Lambda_0 = 2\Delta_\theta Z + \Delta_\theta, \tag{7}$$

where Δ_θ is the quantizer step. The quantizer step size determines the amount of tolerable interference before decoding errors occur. As an empirical study, the modifications are optimized by selecting a step size of $\pi/12$ to achieve very favorable rate-distortion-robustness tradeoffs. Fig. 6 conceptually shows this.

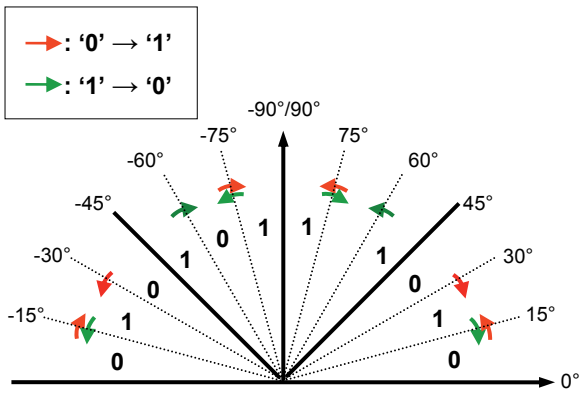


Fig. 6. Stroke direction modulation for $\Delta_\theta = \pi/12$.

The stroke direction modulation technique can be formulated as a problem of modulating the rotatable strokes into the appropriate lattices. If the considered stroke is contained in a lattice mismatching the bit to be embedded, it should be rotated by some degrees in a clockwise or counterclockwise direction around its rotating center to match the bit. Otherwise, it will maintain its existing state. Red and green arrows shown in Fig. 6 definitely specify the steering orientation. To rotate the stroke clockwise, one should specify a negative adjusting value for angle, otherwise take a positive value. Generally, human eyes are capable of detecting the oblique rotation across the specific angles at $\pm\pi/4$ easily. On account of the visual sensitivity to angular deviation, the angles $\pm\pi/4$ are used to divide the direction-space into four subspaces, each as a bounded closure covering three intervals. Such closure is defined in a scope of the variations in stroke direction, which means that any modulated stroke should remain consistent in a directional subspace with the original.

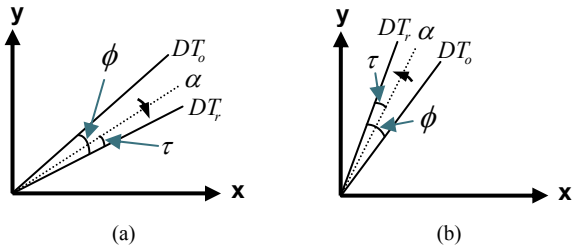


Fig. 7. Representation of rotation angles for (a) clockwise running, (b) counterclockwise running.

Prior to embedding, the rotation angle ϕ is calculated in order to handle the quantitative and orientational modulation. A vector parameter τ is taken into account for the watermark strength. A larger magnitude of τ would increase the robustness, but with more visual impact. Let DT_o and DT_r be the direction angle of a stroke before and after rotation, respectively. The correlative reconstruction point is located at α. According to the steering orientations provided by Fig. 6, two cases, clockwise and counterclockwise running, should be considered. In the first case shown in Fig. 7(a), τ is a negative, and in the second case illustrated in Fig. 7(b), τ is a positive. For both cases, it satisfies

$$\phi = Q_{m[i]}(\theta, \Delta_\theta) - DT_o + \tau \tag{8}$$

where $Q(\cdot)$ is the quantizer; $m[i] \in \{0,1\}$ is one of the message bits. The sign of ϕ accords with the steering orientation for rotation.

After ϕ has been determined, the embeddable stroke pieces isolated from the basic components are then rotated around the rotating centers by the order of embedded bits. A fast rotation algorithm [25] is adopted here. Substitution operation of the original embeddable stroke pieces with rotated ones is conducted directly by logical "OR" function with the fixed components (For more details on "fixed component", please refer to Appendix A). Some applicable

morphological operations are used to bridge unconnected pixels between them.

The general flow of data embedding is briefly illustrated in Fig. 1(a). Two more problems have yet to be addressed as follows.

2.3 Sorting and Shuffling

Watermarking a character with multiple bits seemed to be a fair solution, yet ranging a proper order of strokes to hide bits remains some uncertainty. In order to guarantee the correctness of the watermark extraction, ensuring the ranking of strokes and their related basic components invariant before and after embedding is requisite. Besides, the distribution of rotatable strokes may vary dramatically throughout the document, which leads to uneven embedding capacity. Centroids are proven to be robust against various types of common signal processing, thus they can act as a rotation-invariant feature for location resynchronization between watermark embedding and detection. Regarding that angular modulation of any stroke may slightly change the centroid location of its basic component, we handle the sorting and shuffling of centroids calculated from their fixed components.

The sorting and shuffling process consists of three steps:

Firstly, sort the centroids of fixed components by y-coordinate, and further by x-coordinate for those centroids which have "equal" y-coordinate values. Here we regard $|y_1 - y_2| < \kappa_1$ as $y_1 = y_2$, where κ_1 is an empirical threshold related to PS distortions. In practice, we set $\kappa_1 = 1$. During the experimentation, it was discovered that the amount of centroid deviation of a fixed component throughout the data embedding and detection in electronic forms is negligible.

Secondly, sort the rotatable strokes of each basic component by ranking their junctions along the y-axis, and further by ranking the length of strokes for those junctions which have "equal" y-coordinate values. Here we regard $|y_1 - y_2| < \kappa_2$ as $y_1 = y_2$, with recommended value $\kappa_2 = 3$. If this measurement is not taken, some minor perturbation may be brought to the junctions after embedding, which leads to disturbed order of extracted data bits. Fig. 8 gives a specific example. As seen from Fig. 8(e), (f), the order of rotatable strokes by immediate stroke extraction without sorting in the embedding and detection stages is different in the first two strokes, for their multi-forks locate very close to each other. After sorting, the rotatable strokes are ranked in uniform order in both stages, as shown in Fig. 8(g), (h).

Finally, shuffle the arranged rotatable strokes by pseudo random generators. After shuffling, the distribution of rotatable strokes used for embedding is almost even throughout the text. This also improves the security against steganalysis attack by holding the seed key used to produce a correct shuffling sequence for watermark extraction.

One of the major obstacles to the proposed method is the perceptual distortion that stems from altering the same characters in several nearby places. The shuffling program is effective to overcome it. However, once the shuffling is not functioning adequately, the problem still remains. Under such circumstance, a solution based on simple statistics of neighboring strokes with geometrical similarity can be applied. Any stroke *I* reserved for embedding is considered if the number of its multi-fork junctions is nonzero. In case there exists another rotatable stroke in the local neighborhood, whose direction, magnitude and junction number are identical to *I*, they are both identified as "unembeddable". A small error scope of stroke magnitude is acceptable for tolerance of noise. The neighborhood size is calculated by a multiple of the average vertical profile value of text lines.

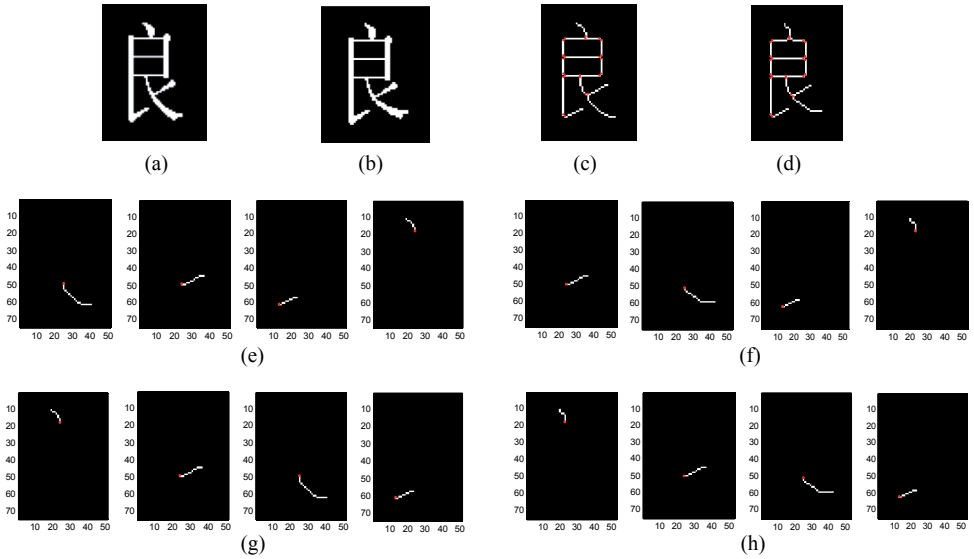
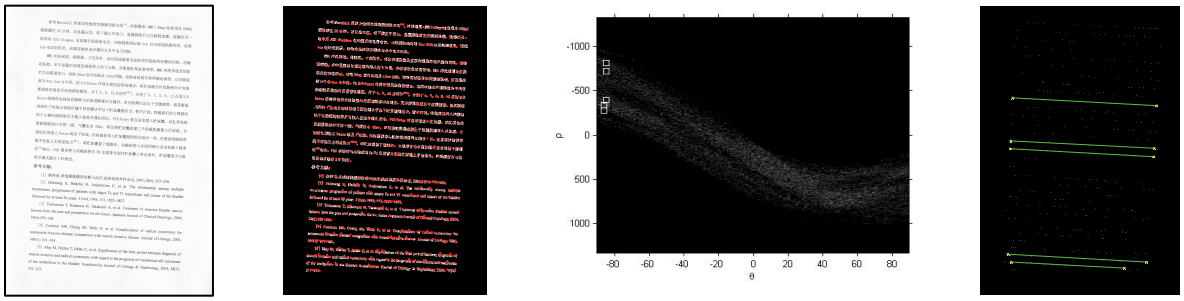


Fig. 8. Example of rotatable stroke sorting (with all multi-fork junctions marked by red points). (a) original image, (b) marked copy with data '0111', (c) skeleton of the original image, (d) skeleton of the marked image, (e) initial order of rotatable strokes of the original image, (f) initial order of rotatable strokes of the marked image, (g) sorted order of rotatable strokes of the original image, (h) sorted order of rotatable strokes of the marked image.



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

Fig. 9. Skew estimation: (a) scanned image, (b) centroid labeling, (c) HT of the centroid image, (d) Hough-based line finding.

2.4 Recovery of Embedded Data

Fig. 1(b) depicts the watermark detection process. With regards to possible attacks such as additive noise and baseline skewing, some compensation measures have to be taken into account.

After the scanned image gets binarized, to suppress the noise, tiny dots that have fewer pixels than a certain number are removed. The classical Hough transform (HT) is universally used as a line detector [26]. It is extended here to identify skew angle of the text lines. Each centroid (x, y) of a basic component is mapped to all values of ρ, θ which satisfy

$$\rho = x\cos\theta + y\sin\theta . \tag{9}$$

The variable ρ is the distance from the origin. θ is the orientation of their normal vector. This parameterization maps each centroid onto a sinusoidal curve in the two-dimensional (ρ,θ) parameter space indicated in Fig. 9(c). The detected lines at the actual peaks are overlaid on the original centroid image in Fig. 9(d). The average value of angles at selected peaks is acquired as the skew angle θ_s. And then compute the biased angle θ_b by

$$\theta_b = \mathrm{sgn}(\theta_s)\cdot\left(90^\circ - \left|\theta_s\right|\right) \tag{10}$$

where sgn(·) represents the signum function. We take θ_b as a control parameter to tackle a possible document misalignment while scanning. Then the orientations performed by directional filters, stroke direction calculation, as well as the centroid location must be rectified by the angle of $-\theta_b$.

Finally, the embedded data are decoded by finding the intervals where the quantizer reconstruction points are contained based on the diagram in Fig. 6.

3. Experimental Results

3.1 Parameters

In our system, a dynamic parameter τ renders a flexible way as a tradeoff between the robustness and visual distortion. The value of τ should be chosen under different circumstances. The static parameters obtained empirically were desirable in most cases of our experiments on a large database of character images. The main parameters were chosen as listed in Tab. 1.

3.2 Visual Perceptibility

We evaluated the visual quality of the watermarked images in a digital-only environment. Fig. 10 shows the 600 dpi binary images converted from a multipage MS-Word document. The document was formatted to different fonts and sizes. We extracted the hidden data from the watermarked binary images without PS. The results were successful. It can be observed from the figure that the proposed technique causes minute perceptual degradation in the documents, even in extreme conditions such as full embedding. Shuffling disperses the locations to be embedded, thus bringing on better visual quality. Under the similarity constraints of neighboring strokes, visual difference caused by altering the same characters in the local is minimized. The same seed key was used for shuffling in Fig. 10(f) and (i). The neighborhood size is a fivefold amount of the average vertical profile value of the text lines. The results indicate that the total benefits gained from the constraints are higher than the total costs, especially for the case of high rate embedding.

	Parameter	Value	Specification
Dynamic	τ for rotation angle calculation	$\tau \in (0,5]$	Determined by the actual needs for robustness.
Static	λ_1 for rotatable stroke decision	λ_1=0.1	Determined by tests showing consistent stroke segmentation before and after embedding.
	λ_2 for rotatable stroke decision	λ_2=0.4	Determined by subjective evaluation of visual effect.
	λ_3 for rotatable stroke decision	λ_3=0.4	Determined by a proper distance which can distinguish between different junctions on a stroke.

Tab. 1. Parameters used in implementation.

Fig. 10. Visual impact tests. (a) original text written in "Song Typeface 12pt", (b) original text written in "Blackbody 10pt", (c) original text written in "Regular Script 11pt", (d) marked text of (a) with 100% embedding rate (hiding 191 bits), (e) marked text of (b) without shuffling (hiding 50 bits), (f) marked text of (c) with shuffling (hiding 50 bits), (g) marked text of (a) with 100% embedding rate under the similarity constraints (hiding 183 bits), (h) marked text of (b) without shuffling under the similarity constraints (hiding 50 bits), (i) marked text of (c) with shuffling under the similarity constraints (hiding 50 bits).

3.3 Attack Resistance

It is worth stressing that our technique is intended for soft authentication. The goal is to deal with admissible manipulations rather than malicious ones. To test the resistance to unintentional attacks, typical image processing operations were executed as shown in Fig. 11. Extensive experiments were conducted, giving a document that contained over 2,000 Chinese characters with three different fonts ("Song typeface", "Blackbody" and "Regular script", all set to 11pt). The bit error rate (BER) was used to test the detection accuracy against attacks:

$$BER = \frac{N_E}{N_T} \left[\times 100\%\right], \tag{11}$$

where N_E is the number of detected bit errors; N_T denotes the total number of embedded bits. If the length of watermark was inconsistent during the detection and embedding phase, the extracted string was padded with zeros or deleted with the surplus part. The results are depicted in Tab. 2. From the results obtained so far, we can conclude that our technique is:

- As expected, robust against rotations within a rational range. It is essentially practical that the amount of rotation an image suffered during PS is generally little.
- Tolerate moderate amount of noise. This is due to the coarse denoising algorithm adopted before detection. When the noise is strong, it may affect the stroke arrangements and leads to error propagation from bit to bit.
- Very robust against JPEG lossy compression regardless of the compression level, even to the greatest extent. The quality factor Q mentioned here is inversely proportional to the compression level.
- Robust against a certain degree of rescaling. The cropping is achieved by uniform pixel reduction under constant resolution.
- Robust against median filtering using an appropriate neighborhood. When the neighborhood is too large, the image is over blurred so that the watermark is doomed to failure.

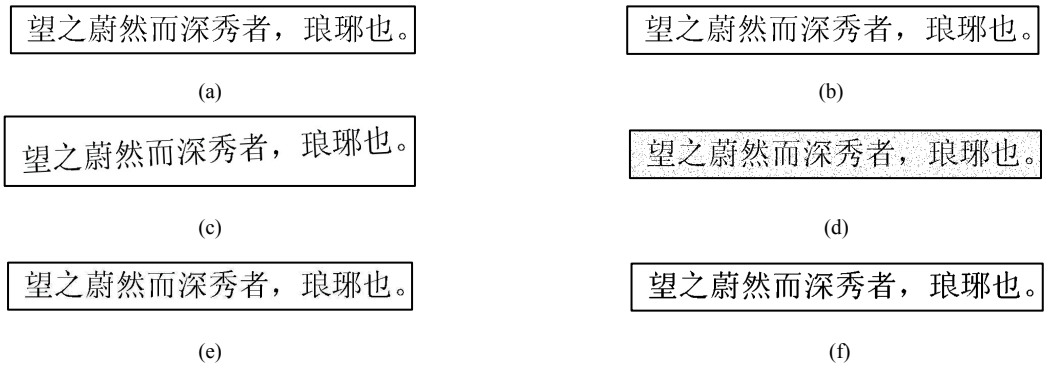


Fig. 11. Various types of attacks. (a) original text; (b) marked text with data '10101111101'; (c) 3° rotation; (d) 3% salt & pepper noise; (e) JPEG lossy compression with Q = 10%; (f) median filtering using 2 × 2 window size.

Operations	*BER* (×100%)	Operations	*BER* (×100%)
1° rotation	1.000	JPEG_10%	1.000
3° rotation	1.000	JPEG_50%	1.000
5° rotation	1.000	Cropped by 10%	1.000
1% noise	1.000	Cropped by 30%	0.840
3% noise	0.996	Cropped by 50%	0.623
5% noise	0.749	Median filtering	0.972

Tab. 2. Bit error rates under various attacks.

3.4 Print-Scan Tests

We used a HP Laserjet P1505n for printing the documents at a default resolution of 600 dpi. A Canon IR1022 was used for high quality photocopying. The hardcopy documents were scanned on a Microtek scanner at 600 dpi. Copies were created for diverse fonts and sizes with full embedding. The image library includes five typical computer fonts ("Song typeface", "Blackbody", "Regular script", "Yoyuan" and "Lisu"), in three different sizes (10pt, 11pt and 12pt). Fig. 12 gives the results of *BER* against print-scan and print-photocopy-scan.

Fig. 12 shows that it is possible to achieve a result close to zero bit error. Our method is not limited to a particular font, but under the impact of the stroke extraction accuracy. Small differences between the print-scanned results and print-photocopy-scanned results were indicated less than 4%. Song typeface provided the most consistent results, superior to any other types. Lisu font did not perform well, since it encountered the problem of improper stroke generation. *BER* of the other fonts are all below 10%. The most frequent errors occur at the embedded bits corresponding to the locations of short strokes. This highlights the fact that the principle of this system works, and more satisfying performance requires a development of the relevant algorithms such as flexible binarization and reliable stroke generation.

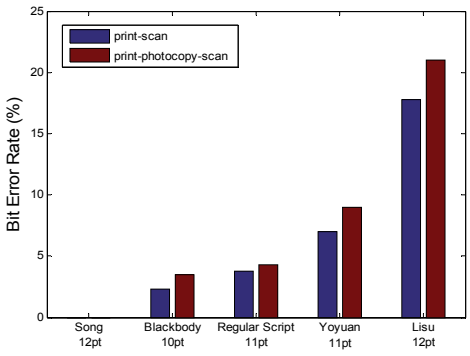


Fig. 12. Detection results for different fonts. The blue bars represent the print-scanned results; the red bars represent the print-photocopy-scanned results.

3.5 Comparison of Techniques

A comparison between our method with other existing techniques were summarized in Tab. 3. To our knowledge, most state-of-the art techniques were developed for English texts. It brings difficulty in comparing the capacity of various methods with a single standard. Previous work [27] deals only with documents in electronic form, and can only provide partial robustness against geometric distortions. Besides, it provides the lowest hiding capacity. The other techniques survive in both electronic and printed

forms. The dot inserting approach [22] has steady and high capacity since it is totally independent of the text content. Unfortunately, it can barely resist median filtering and noise interference. Moreover, the embedded marks can be attacked or deleted easily. Culnane et al. improved over the idea of word shifting. Still, it has lower capacity. The proposed technique and [2] offer robustness to PS and copying, yet provides increased data hiding capacity, which partly depends on the number of characters in the text. To ensure the detection accuracy, an error correcting code and an OCR aided detector are required in [2]. It can be concluded that our approach significantly outperforms other methods in terms of robustness, hiding capacity, and limitations.

Ref.	Techniques	Robustness	Capacity	Limitations
Proposed	Stroke direction modulation	Print-photocopy-scan	High/Medium: about 2 bits per Chinese character	Stroke-based languages only
Y. -W. Kim et al. [27]	Modification of edge direction histogram	Geometric attacks	Low: approaching 1 bit per line	Electronic form only
Culnane et al. [18, 28]	Word space modulation	Print-scan	Low: $\leq 1/6$ bit per word	Alphabet-based languages only
H. Y. Kim et al. [22]	Insertion of tiny registration dots	Print-photocopy-scan	High: >1000 bits in an A4-sized document	Vulnerable to median filtering and noise
Varna et al. [2]	Shape modification of vertical strokes	Print-photocopy-scan	High/Medium: about 1 bit per English character	The need for ECC and an additional helper database

Tab. 3. Performance comparison.

4. Conclusion

This paper has presented a novel watermarking technique for printed binary documents, in which a scheme of stroke direction modulation is employed. It embeds the information by quantizing the angle formed by a host signal vector, i.e. the stroke direction. This needs to be founded on automatic partition of the embeddable stroke pieces from their original characters. "Cross Number" criterion is modeled in search of multi-fork junctions which are considered as a key feature for the overall system. No side information is involved for introducing the invariant feature of robust data embedding, namely the centroids of fixed components. The shuffling and similarity constraints are developed for modification with less visual impact. In the detection process, noise reduction and skew estimation are used for correcting geometric distortions. Experimental results indicate that the proposed approach has good imperceptibility, high embedding rate, and desirable level of robustness against PS operations and accidental attacks.

But then, our technique requires reliable extraction of character strokes. We believe that the proposed scheme can be generalized to more created fonts if more efforts to improve the stroke extraction accuracy are made. Likewise, an adaptive binarization algorithm is beneficial to enhance the detection performance. Further, it leaves more motivated minds to conceive an adaptive quantizer for achieving much better watermark invisibility compared to the uniform quantizer. Incorporating error correcting codes is worth trying as well, although it will result in sacrificing some hiding capacity.

Until now, soft authentication watermarking for binary images, especially binary hardcopy documents, remains to be a difficult task. The main requirements for this application are: (a) a watermarking technique with sufficient embedding rate, which can resist hardcopy operations; (b) a perceptual hash as a fingerprint of the document, derived from visual features of its content; (c) a secure authentication mechanism lying on the secrecy of a private key. This paper is dedicated to the first item of the coherent framework. Considerable more work will be done on this area.

Acknowledgements

This work was supported by the National Basic Research Program of China (Grant No. 2009CB326202, 2010CB334706 and 2011CB311808), National Natural Science Foundation of China (Grant No. 60736016, 60973128, 60973113, 61073191, 61070195, 61070196 and 61103215), and PAPD.

References

[1] ANAN, T., KURAKI, K., NAKAGATA, S. Watermarking technologies for security-enhanced printed documents. *Fujitsu Scientific and Technical Journal,* 2007, vol. 43, no. 2, p. 197 - 203.

[2] VARNA, A. L., RANE, S., VETRO, A. Data hiding in hard-copy text documents robust to print, scan and photocopy operations. In *International Conference on Acoustics, Speech, and Signal Processing*. Taipei (Taiwan), 2009, p. 1397 - 1400.

[3] ZHU, B. B., SWANSON, M. D., TEWFIK, A. H. When seeing isn't believing. *IEEE Signal Processing Magazine,* 2004, vol. 21, no. 2, p. 40 - 49.

[4] WANG, X., WU, J., NIU, P. A new digital image watermarking algorithm resilient to desynchronization attacks. *IEEE Transactions on Information Forensics and Security,* 2007, vol. 2, no. 4, p. 655 - 663.

[5] SOLANKI, K., MADHOW, U., MANJUNATH, B. S., CHANDRASEKARAN, S., EL-KHALIL, I. 'Print and scan' resilient data hiding in images. *IEEE Transactions on Information Forensics and Security,* 2006, vol. 1, no. 4, p. 464 - 478.

[6] ZHENG, D., WANG, S., ZHAO, J. RST invariant image watermarking algorithm with mathematical modeling and analysis of the watermarking processes. *IEEE Transactions on Image Processing,* 2009, vol. 18, no. 5, p. 1055 - 1068.

[7] LI, R. J., CHANG, L. W. Data hiding in binary images for annotation by parity check. In *International Symposium on Intelligent Signal Processing and Communications Systems*. Tottori (Japan), 2006, p. 764 - 767.

[8] PUHAN, N. B., HO, A. T. S. Binary document image watermarking for secure authentication using perceptual modeling. In *IEEE Symposium on Signal Processing and Information Technology*. Athens (Greece), 2005, p. 393 - 398.

[9] MEI, Q., WONG, E. K., MEMON, N. Data hiding in binary text documents. In *Proceedings of SPIE - The International Society for Optical Engineering*. San Jose (USA), 2001, p. 369 - 375.

[10] KIM, H. Y., AFIF, A. A secure authentication watermarking for halftone and binary images. *International Journal of Imaging Systems and Technology,* 2004, vol. 14, no. 4, p. 147 - 152.

[11] YANG, H., KOT, A. C., RAHARDJA, S. Orthogonal data embedding for binary images in morphological transform domain - a high-capacity approach. *IEEE Transactions on Multimedia,* 2008, vol. 10, no. 3, p. 339 - 351.

[12] YANG, H., KOT, A. C. Pattern-based data hiding for binary image authentication by connectivity-preserving. *IEEE Transactions on Multimedia,* 2007, vol. 9, no. 3, p. 475 - 486.

[13] CHANG, C.-C., TSENG, C.-S., LIN, C.-C. Hiding data in binary images. *Lecture Notes in Computer Science,* 2005, vol. 3439, p. 338 - 349.

[14] WU, M., LIU, B. Data hiding in binary image for authentication and annotation. *IEEE Transactions on Multimedia,* 2004, vol. 6, no. 4, p. 528 - 538.

[15] BRASSIL, J. T., LOW, S., MAXEMCHUK, N. F. Copyright protection for the electronic distribution of text documents. *Proceedings of the IEEE,* 1999, vol. 87, no. 7, p. 1181 - 1196.

[16] HUANG, D., YAN, H. Interword distance changes represented by sine waves for watermarking text images. *IEEE Transactions on Circuits and Systems for Video Technology,* 2001, vol. 11, no. 12, p. 1237 - 1245.

[17] AMANO, T., MISAKI, D. A feature calibration method for watermarking of document images. In *International Conference on Document Analysis and Recognition.* Los Alamitos (USA), 1999, p. 91 - 94.

[18] CULNANE, C., TREHARNE, H., HO, A. T. S. A new multi-set modulation technique for increasing hiding capacity of binary watermark for print and scan processes. In *International Workshop on Digital Watermarking*. Jeju Island (Korea), 2006, p. 96 - 110.

[19] VILLÁN, R., VOLOSHYNOVSKIY, S., KOVAL, O., VILA, J., TOPAK, E., DEGUILLAUME, F., RYTSAR, Y., PUN, T. Text data-hiding for digital and printed documents: Theoretical and practical considerations. In *Proceedings of SPIE - The International Society for Optical Engineering*. San Jose (USA), 2006, p. 406 - 416.

[20] BRASSIL, J. T., LOW, S., MAXEMCHUK, N. F., O'GORMAN, L. Electronic marking and identification techniques to discourage document copying. *IEEE Transactions on Selected Areas in Communications,* 1995, vol. 13, no. 8, p. 1495 - 1504.

[21] JIANG, M., WONG, E. K., MEMON, N. Robust document image authentication. In *IEEE International Conference on Multimedia and Expo*. Beijing (China), 2007, p. 1131 - 1134.

[22] HAE, Y. K., MAYER, J. Data hiding for binary documents robust to print-scan, photocopy and geometric distortions. In *XX Brazilian Symposium on Computer Graphics and Image Processing.* Belo Horizonte (Brazil), 2007, p. 105 - 112.

[23] SU, Y.-M., WANG, J.-F. A learning process to the identification of feature points on Chinese characters. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans,* 2003, vol. 33, no. 3, p. 386 - 395.

[24] ZHANG, X., SONG, J., DAI, G., LYU, M. R. Extraction of Line Segments and Circular Arcs From Freehand Strokes Based on Segmental Homogeneity Features. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics,* 2006, vol. 36, no. 2, p. 300 - 311.

[25] CHIEN, S.-I., BAEK, Y.-M. A fast black run rotation algorithm for binary images. *Pattern Recognition Letters,* 1998, vol. 19, no. 5, p. 455 - 459.

[26] AGGARWAL, N., KARL, W. C. Line detection in images through regularized Hough transform. *IEEE Transactions on Image Processing,* 2006, vol. 15, no. 3, p. 582 - 591.

[27] KIM, Y.-W., OH, I.-S. Watermarking text document images using edge direction histograms. *Pattern Recognition Letters,* 2004, vol. 25, no.11, p. 1243 - 1251.

[28] CULNANE, C., TREHARNE, H., HO, A. T. S. Improving multi-set formatted binary text watermarking using continuous line embedding. In *Proceedings of the 2nd International Conference on Innovative Computing, Information and Control.* Kumamoto (Japan), 2008, p. 287 - 292.

About Authors ...

Lina TAN is currently pursuing her PhD in computer science and technology at the College of Information Science and Engineering, Hunan University, China. Her research interests include information security, digital watermarking, image processing and pattern recognition.

Xingming SUN is currently a professor in Nanjing University of Information Science and Technology, China. His research interests include network and information security, digital watermarking, digital forensic and natural language processing.

Guang SUN is currently pursuing his PhD in computer science and technology at the College of Information Science and Engineering, Hunan University, China. He is now teaching at Hunan University of Finance and Economics. His main research interests are information security, software watermarking, software birthmarking and complex network.

Appendix A

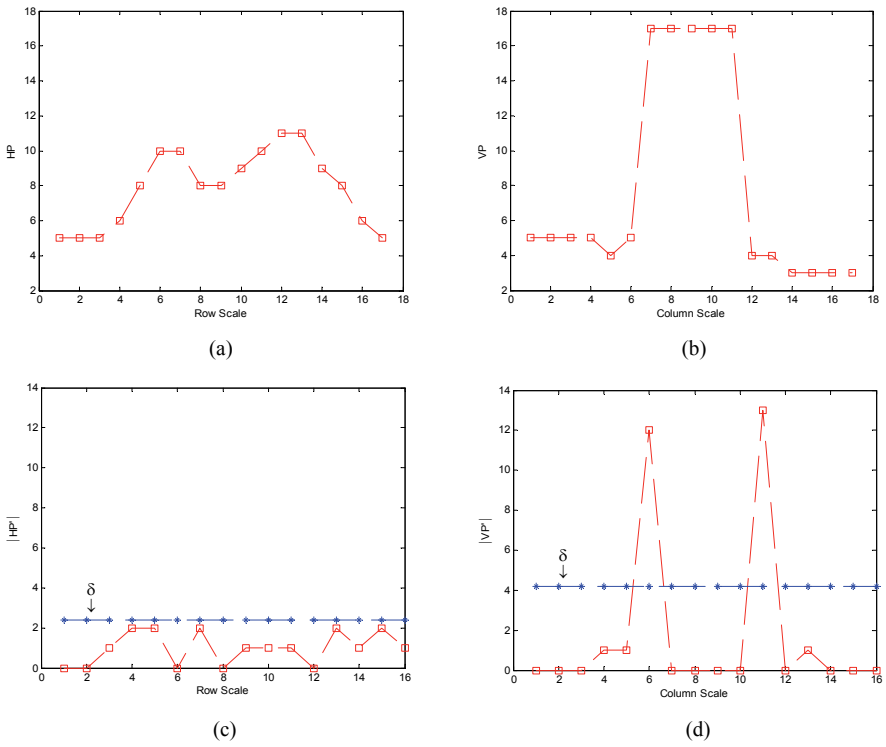


Fig. 13. Graph of functions utilized for dividing boundary location. (a) horizontal profile, (b) vertical profile, (c) $|HP'|$ and δ, (d) $|VP'|$ and δ.

Here an approach is developed to mark out the dividing boundaries inside where the area is to be set blank. Let HP and VP be the horizontal and vertical profiles of an image block (refer to [18]), respectively. Fig. 13(a), (b) exhibit the counterpart profiles of a block within the red bounding box shown in Fig. 5. The block size is chosen directly proportional to the stroke magnitude. We observed that there is generally great quantity variance on either of the two profiles nearby the character boundaries, where are fit for setting the dividing boundaries. This is achievable by computing the absolute first difference of the couple profiles, denoted by $|HP'|$ and $|VP'|$.

Fig. 13(c), (d) present the results of $|HP'|$, $|VP'|$ for the data worked out in Fig. 13(a), (b). As can be seen in Fig. 13(d), there are two clear peaks in the exact locations where we can detect distinct variations in the original image as well. A threshold δ adequate to determine the peaks is given below. We refer to the expected value as the mean, denoted by $E(\cdot)$.

$$\delta = \omega E\left(|\gamma'|\right), \tag{12}$$

where $\omega = 2.4$ is an empirical constant, and $\gamma \in \{HP, VP\}$. From Fig. 13(c), we noted that $|HP'|$ distributed under δ all along, indicating that the amplitude variation of HP is subtle and almost even. Accordingly, it is hardly possible to examine abrupt peaks and work from these points to decide the boundaries. Our solution to cope with the issue is posed in Algorithm 1. In the algorithm, trip point is a nonzero pixel whose nearest neighbor pairs in one direction (horizontal, vertical or diagonal) is opposite.

Once the dividing boundaries are determined, all the pixels inside them are assigned zero. Thereafter, the embeddable stroke pieces are easily picked up from the split basic component by labeling connected components that overlap each rotatable stroke. The remaining part given by subtracting all the embeddable stroke pieces from the basic component is expressed as a fixed component.

Algorithm 1: Dividing boundary location around a rotating center RC **;**

IF there are two clear peaks above δ in the function of $|HP'|$ (or $|VP'|$),

The row (or column) boundaries are set at the exact location of the peaks;

ELSE

The row (or column) boundaries are set at the exact row (or column) scale of the nearest trip point towards RC and with the same column (or row) as RC , and restricted in size of the block;

END IF