Proceedings of International Conference on
Engineering and Information Technology
Sep. 17-19, 2012, Toronto, Canada

Information hiding in fingerprint image

Prof. Dr. Tawfiq A. Al-Asadi [a], MSC. Student Ali Abdul Azzez Mohammad Baker [b]

[a] Information Technology collage, Babylon University
[b] Department of computer science, Babylon University

Abstract

In this paper, we introduce a new method to hide information (such as face image, voice, signature…) in ridges of a fingerprint image. The place where the message is hidden depends on features extracted from the image (like core, delta, ending and bifurcation). The proposed system consists of two stages; the First stage is for hiding a message which means extracting features from a fingerprint image to determine useless regions, while the second stage is for extracting a message from a fingerprint image. Hiding information in a fingerprint image must never alter the positions and numbers of important regions or pixels .Hiding information helps the automated fingerprint identification system in making the individual identification decision more reliable. In this paper a new technique in the way of hiding information is used depending on the nature of the cover image by determining the edges of an object and hiding the message between these edges.

Keywords: Fingerprint, segmentation, Poincare index, Minutiae detect, hide information.

1-Introduction

A fingerprint image has been used for personal identification for many decades because:

1- It is highly reliable.

2-A fingerprint remains constant during people's life [2].

3-A fingerprint image has a unique property in that the fingerprints of the identical twins are different and so are the prints of each finger of the same person.

A fingerprint is a pattern of parallel ridges and valleys, the ridges (also called ridge lines) are dark whereas the valleys are bright as illustrated in figure (1).

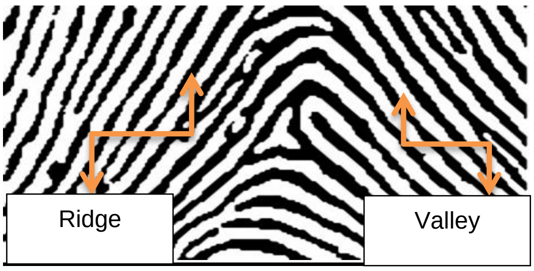
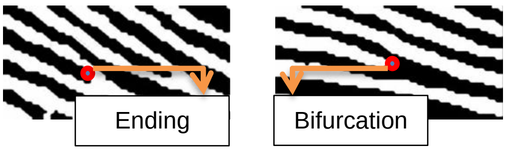


Figure (1) ridges and valleys pattern

Features in a fingerprint image are divided into two levels:

• The local level for minutiae detects (ending and bifurcation). Minutiae are local discontinuities in the fingerprint pattern (as shown in figure (2)) which are used in a fingerprint matching stage, and the similarity between two fingerprints is determined by comparing the two sets of minutiae points .



(a) Ridge ending (b) Ridge bifurcation
Figure (2) ridge ending and bifurcation

• The global level for a singular points (or singularities) detection (core and delta) ,as illustrated in figure (3).This process is a very important task used in a fingerprint classification system into almost (5) classes according to its geometric properties which are Arch, Tented Arch, Left loop, Right loop ,and Whorl as shown in figure(4).

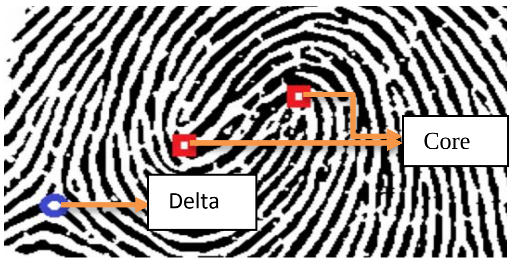
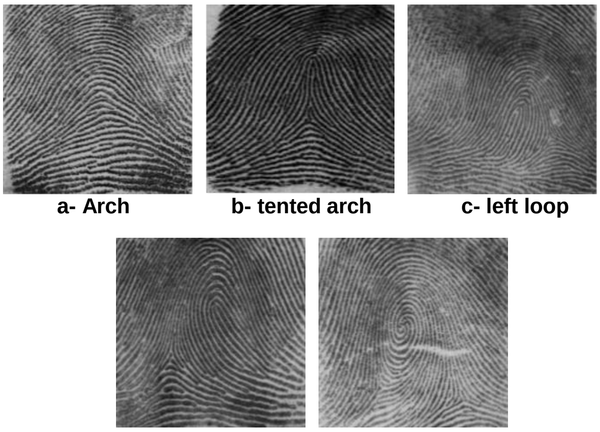


Figure (3) core(square) and delta(circle)

The classification of the fingerprint images into five classes depending on the singularities type, number, and position as illustrated in table (1).

Many algorithms were proposed for extracting features from the fingerprint image and a lot of techniques were used in the fingerprint identification system.



a- Arch **b- tented arch** **c- left loop**

d- Right loop **e- whorl**

Figure (4) classes of fingerprint images

Table (1) fingerprint pattern classes

Fingerprint Pattern class	Core numbers	Delta numbers	Delta position
arch	0	0	-
Left loop	1	1	right
Right loop	1	1	left
Tented arch	1	1	middle
whorl	2	2	Left and right

The proposed system extracts features from fingerprint pattern image in order to avoid ridges which contain or near these features and use the rest ridges in hiding information to avoid affect in the fingerprint pattern which may be lead to detect false properties or reject true properties compare with the original image ,this mean ,after hiding information stage the fingerprint image must has the same number and position for each feature (core, delta, ridge ending ,and bifurcation) as in original image, this new image will be stored in database and this information will be extracted only when automated fingerprint identification system fails in personal identification then this information helps the system in making more reliable decision.

2-The proposed system

In generl,the proposed system as illustrated in figure(5) consists of two steges.The first one is the message hiding stage in the fingerprint ridges and the second one is the message extracting stage.

2.1-The message hiding stage

This stage is divided into the following steps:

1- Fingerprint image enhancement.

2- Fingerprint image segmentation into isolated regions.

3- Singular points detection.

4- Minutiae detection.

5- Hiding information using a new algorithm into useless regions.

2.1.1-Fingerprintimage enhancement

Many methods are suggested to enhance the fingerprint image using ad hoc strategy to enhance the quality of the fingerprint image [3].

$$\mathrm{Eimag}(i,j) = \alpha + \Upsilon \times \left(\frac{[\mathrm{image}(i,j) - \mu]}{s}\right) \qquad (1)$$

Where,

Eimage is the enhancing image, Image is the original image, (μ) is the mean of the original image, (S) is the variance of the original image.

From empirical results α=150 and Y=95 [3], (S) must be less than 95 else the constant value (95) will be replaced with another number greater than S.

2.1.2- Fingerprint image segmentation

After the enhancement step we focus on the ridge segments by separating it into isolated regions using edge detection segmentation. Firstly, the enhanced image is converted into a binary image by dividing the image into (17×17) non overlap blocks and calculating the mean for each block using equation (2)

$$\text{block mean} = \frac{1}{17\times17}\sum_{i=0}^{16}\sum_{j=0}^{16}\mathrm{image}(i,j) \qquad (2)$$

Then the pixel in the binary image becomes:

The binary image (i, j) =255 if the enhanced image pixel (i, j) ≥ block mean.

The binary image (i, j) =0 if the enhanced image pixel (i, j) < block mean.

Then the edges detected by simple gradient method on the binary image with the vertical and horizontal masks [6].

1
-1

Vertical mask

-1	**1**

horizontal mask

If the binary image (i, j+1) -the binary image (i, j) =255 then the pixel (i, j) is the edge (boundary) pixel.

If binary image (i, j) – the binary image (i-1, j) =255 then pixel (i, j) is the edge pixel.

Figure (6a) shows the edges detected for the fingerprint image. When the edges are detected, the image is divided into isolated regions by collecting the pixels inside each closed boundary to construct each region with a label as shown in figure (6b and 6c).

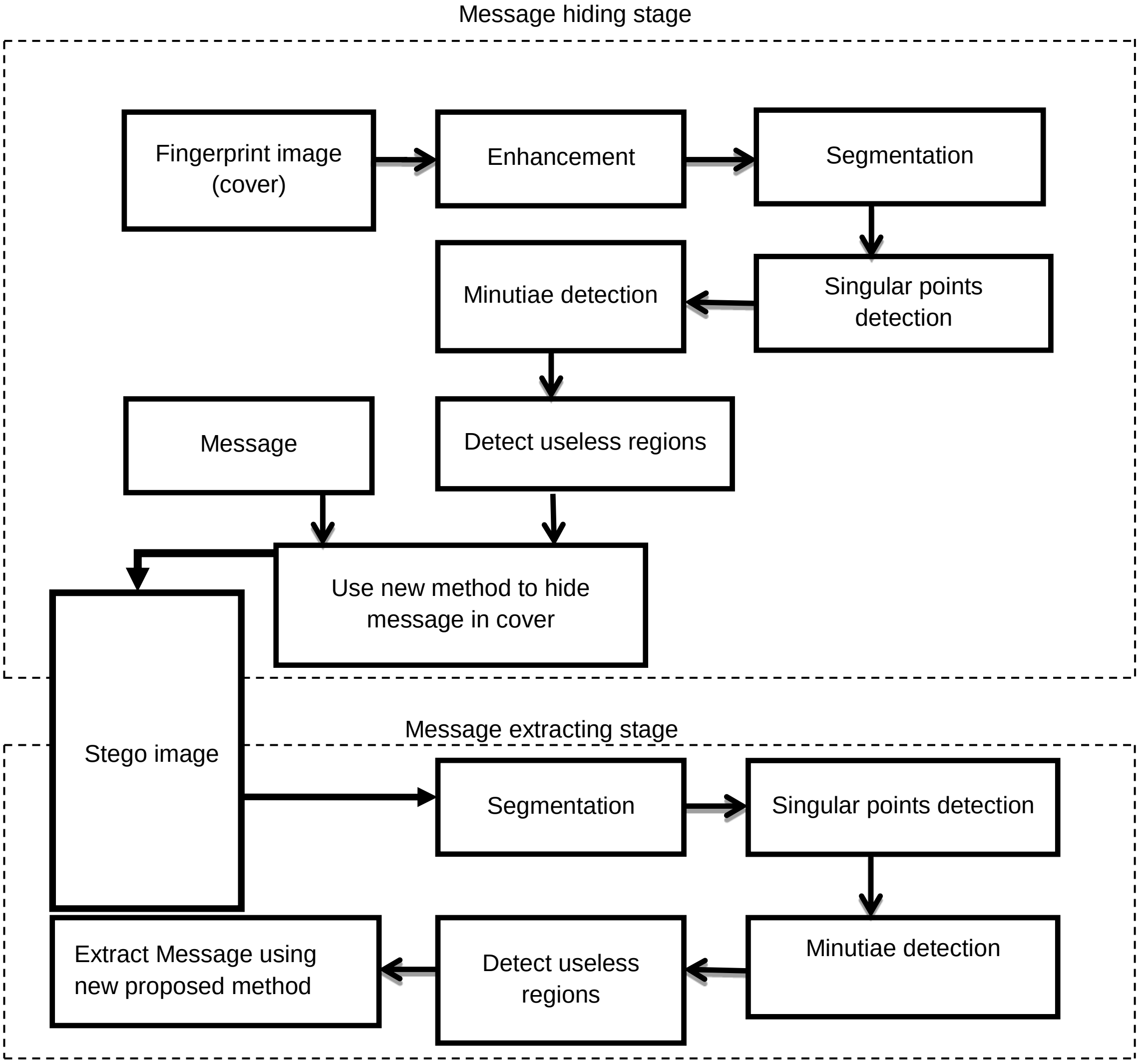


Figure (5) block diagram of proposed system

2.1.3-Singular points detection

For core and delta detection firstly, we compute the orientation for each block with selected size and then compute the Poincare index.

For orientation computation the horizontal and vertical gradient (Gx, Gy) will be calculated for every pixel in the enhanced image using sobel masks as follow

Z1	Z2	Z3
Z4	Z5	Z6
Z7	Z8	Z9

Image

-1	-2	-1
0	0	0
1	2	1

horizontal mask

-1	0	1
-2	0	2
-1	0	1

vertical mask

$$G_x = (Z7 + 2Z_8 + Z_9) - (Z1 + 2Z_2 + Z_3) \quad (3)$$

$$G_y = (Z3 + 2Z_6 + Z_9) - (Z1 + 2Z_4 + Z_7) \quad (4)$$

Then the fingerprint image will be divided into non overlap blocks of size (W×W) where W=16, the average gradient in each block R is

$$\begin{bmatrix} \tilde{\alpha}_x \\ \tilde{\alpha}_y \end{bmatrix} = \frac{1}{w^2} \sum_R \begin{bmatrix} Gx^2 & - & Gy^2 \\ 2 & Gx & Gy \end{bmatrix} \quad (5)$$

The block gradient direction θ is

$$\theta = \frac{1}{2}\, \delta\,(\tilde{\alpha}_x\ ,\ \tilde{\alpha}_y\,) \quad (6)$$

Where,

$$\delta(x,y) = \begin{cases} \tan^{-1}\left(\frac{y}{x}\right) & \text{If } X \geq 0 \\ \tan^{-1}\left(\frac{y}{x}\right) + \pi & \text{If } X < 0 \text{ and } Y \geq 0 \\ \tan^{-1}\left(\frac{y}{x}\right) - \pi & \text{If } X < 0 \text{ and } Y < 0 \end{cases} \quad (7)$$

For Poincare index calculation which detects singular points speedy and directly, the blocks size (16×16) will be used then the Poincare index is the summation of the angles difference for (8) neighboring blocks along counter-clockwise direction.

$$\text{poincare index}(i,j) = \frac{1}{2\pi} \sum_{k=1}^{8} \Delta(k) \quad (8)$$

a-edges detection **b-segmented image** **c- region drawing**

Figure (6) segmented regions

$$\Delta(k) = \begin{cases} \delta(k) & \text{if } |\delta(k)| < \frac{\pi}{2} \\ \pi + \delta(k) & \text{if } \delta(k) < \frac{-\pi}{2} \\ \pi - \delta(k) & \text{otherwise} \end{cases} \quad (9)$$

$$\delta(k) = \theta(x_{(k+1) \bmod 8}, y_{(k+1) \bmod 8}) - \theta(x_k, y_k) \quad (10)$$

δ(1)	δ(8)	δ(7)
δ(2)	(i , j)	δ(6)
δ(3)	δ(4)	δ(5)

If Poincare index of any block = 0.5 then this block contains core point.

If Poincare index of any block = -0.5 then this block contains delta point.

2.1.4-Minutiae detection

Most automatic fingerprint identification (verification) systems are based on minutiae matching; Minutiae are local discontinuities in the fingerprint pattern. A total of 150 different minutiae types have been identified. In practice only the ridge ending and the ridge bifurcation minutiae types are used in fingerprint recognition [4]. Examples of minutiae are shown in figure (2), in this step we use the same binary image determined in section 3.2 and the ridges in binary image are thinned to one pixel thick as in figure (7), we examine the neighborhoods of each pixel in the binary image and decide if the pixel can be deleted or not until one thick pixel ridge. After the thinning phase each pixel in the thinned image checked as follow:

If the pixel has only one black neighborhood of eight then the pixel is an end of the ridge and if the pixel has three black neighborhoods of eight then the pixel is a bifurcation of the ridge.

2.1.5-Hiding message in useless regions

After we detect core, delta, ending, and bifurcation each region has any feature will neglect and we will collect the other regions to hide in. We use a new technique to hide message in fingerprint image into useless regions by applying the following algorithm for each useless region

Figure (7) thin image

Hiding algorithm

Input: useless ridge regions, message (image).

Output: stego image

Begin

For i=1 to number of rows in region do

If all row (i) pixels <> value of boundary then

Call function to calculate mean of row (i) pixels.

Covert all row (i) pixels to equal the mean except the boundary pixels.

Convert the mean value into binary representation.

Save the mean value in the first pixel of row (i).

Calculate the KMSB (the number of most significant Bits Which equal zero)

Hide each bit in message each Byte of row (i) (in the Reset of 8- KMSB bits)

End for

End algorithm.

When we apply this algorithm the following states must be checked

1- If all pixels in the row are equal to the boundary value then the row must be neglected.
2- We leave the first and the last values of each row because they are edge.
3- For each row which verifies the two conditions above apply the following

A- Calculate the mean of row pixels.
B- Never hide in the first pixel of row.
C- Convert the mean to binary number and perform the following

i- Calculate the number of the most significant bits equal to zero.
ii- Reduce these bits and hide in the remaining bits of each pixel.

For region in figure (8)

1-We ignore the first row because all pixels in this row are edge.

Row number	Column number				
	1	2	3	4	5
1	90	90	90	90	
2	90	81	85	90	
3	90	55	50	54	90
4	90	80	82	84	90
5	90	78	80	82	90
6	90	90	90	90	90

a-input region

Row number	Column number				
	1	2	3	4	5
1	90	90	90	90	
2	90	83	83	90	
3	90	53	53	53	90
4	90	82	82	82	90
5	90	80	80	80	90
6	90	90	90	90	90

b-apply algorithm

Figure (8) applying algorithm for random region

2- For the second row the mean of the pixels (not boundary) equal to ((81+85)/2) =83 which represented in binary.

0	1	0	1	0	0	1	1

Then we found that the number of MSBs equal to zero is (1), the number of bits that will be used in hiding information equal to (7) and we hide only in the pixel in the column number (3), we let pixel in column (2) as a guide to retrieve information in the second stage.

For the third row the mean of the pixels equal to ((55+54+50)/3) =53, in the same way we hide in six least significant bits of the pixels in column (3) and (4) only.

0	0	1	1	0	1	0	1

2.2-The message extracting stage

In second stage we apply the same techniques used in section 2.1.2, 2.1.3, and 2.1.4 for fingerprint segmentation, Singular points detection, and Minutiae detection respectively to determine the useless regions to extract information from it, this means that the second stage for extracting information from useless regions in a fingerprint image consists of the following steps:-

1- Fingerprint image segmentation.
2- Singular points detection.
3- Minutiae detection.
4- Determine useless regions to extract message from them.

In the same way when we need to extract a message from a fingerprint image we read the first pixel in each row in the useless regions then from this value we know the number of the pixels used in hiding information by converting this value to binary and calculating the number of MSBs equal to zero, from this value we determine the number of the bits used in information hiding.

2.3-Experiment Results

Figure (9) explains the result of the proposed system. The dimensions of the original and stego images are (504 × 532) and the dimensions of the message image are (84×112). The peak signal to noise ratio for this state is (30.4) which is calculated using equation (11).

$$SNR_{peak} = 10\, log_{10} \frac{(L-1)2}{\sqrt{\frac{1}{N^2}\sum_{r=0}^{n-1}\sum_{c=0}^{n-1}[g(r,c)-I(r,c)]^2}} \qquad (11)$$

In figure (10) the dimensions of the original and stego images are (504×532) and the dimensions of the message image are (108*104).The peak signal to noise ratio for this state is (30.14).

a.original image **b. message** **c.stego image**

Figure (9) result (1) of the proposed system

a.original image **b. message** **c.stego image**

Figure (10) result (2) of the proposed system

3-Conclusion

The gray level of Some useless pixels closer to mean of the image so after hiding information these pixels may be accessed the image mean in both directions (black pixels become white in converting to binary image or white pixels become black) therefore these pixels must be handled to avoid the errors in extracting information.

Hide of information doesn't alter the main features of the fingerprint image so the system of fingerprint matching doesn't be affected and when automated fingerprint system fails in the personal identification, hiding information can play a major role to assist the decision for accepting or rejecting the matching decision.

Segmentation of image is the key for sound results; any way leads to under segmentation cause to merge many ridges in one region which effect in reducing the total number of the pixels available for hiding information while the results of over segmentation make the positions of pixels used in hiding information more randomly.

Acknowledgments

We thank all the participants to the International Conference on Engineering and Information Technology.

References

[1] Anil K. Jain, Patrick Flynn, and Arun A.ross, "Handbook of Biometrics", USA, 2008.

[2] Qinzhi Zhang, Kai Huang, and Hong Yan," Fingerprint Classification Based on Extraction and Analysis of Singularities and Pseudo ridges", Australia, 2006.

[3] Peihao Huang, chia-yung, chaur-chin Chen, "implementation of an automatic fingerprint identification system", Taiwan, 2007.

[4][American university of Beirut faculty of engineering and architecture department of electrical and computer engineering]," Fingerprint Identification – Project 2".

[5] Liu Wei, "Fingerprint Classification Using Singularities detection", international journal of mathematics and computers in simulation, 2008.

[6]Sumit Tandon,"edge detection", university of Texas at Arlington, 2005.