

# Fingerprint Patterns Recognition System Using Huffman Coding

Abdurazzag Ali Aburas and Salem Ali Rehiel, *Member, IAENG*

## Abstract

Computers don't match fingerprints the way human beings do. Instead of looking at the patterns of arches, loops, and whorls, AFISs (Automated Fingerprint Identification Systems) reduce the fingerprint image to a table of two-dimensional vectors. Called *minutiae*, these vectors correspond to the places on a fingerprint where a ridge begins, ends, or splits from one ridge into two. Each minutia has an exact (x,y) position within the fingerprint, as well as a direction in which it points. In despite of this improvement which is adopted by the Federal Bureau of Investigation (FBI), the fact still is "The larger the fingerprint files became, the harder it was to identify somebody from their fingerprints alone. Moreover, the fingerprint requires one of the largest data templates in the biometric field". The finger data template can range anywhere from several hundred bytes to over 1,000 bytes depending upon the level of security that is required and the method that is used to scan one's fingerprint. For these reasons this work is motivated to present another way to tackle the problem that is relies on the properties of Huffman coding algorithm. No additional verifications are needed. All you need is the image itself and go ahead. The obtained results are very promising in terms of simplicity, reliability, and cost (time & storage).

*Index term* AFIS, biometric, Huffman coding, image compression, FBI

## I. INTRODUCTION

Fingerprint imaging technology has been in existence for centuries [1]. It has been estimated that the chance of two people, including twins, having the same fingerprint is less than one-in-a-billion [2][3]. Fingerprint imaging technology looks to capture or read the unique pattern of lines on the tip of one's finger. These unique patterns of lines can either be in a loop, whorl, or arch pattern [4]. There are several methods in accomplishing the process of identifying one's fingerprint [5]. The two major applications of fingerprint recognition are fingerprint verification and fingerprint identification. Verification is known as one-to-one fingerprint matching, whereas, fingerprint identification is known as one-to-many matching. Verification is used for access applications and identification is used for investigation purposes. Many successful approaches have been presented for verification applications, where the identification field is still open challenging area because of many problems. The main common problem that the verification faces is the huge database. This problem can be classified into two categories: first is establishing identify based on single print i.e. comparing a latent print from the scene of a crime with prints from a file. Second is establishing identify based on a set of ten fingerprint

records i.e. record. Applying the Huffman code for handwriting Arabic character recognition such that the Huffman code could is producing a unique code representing corresponding image [6]. After arrest or employment are requirements. For reasons discussed in [7], the problem of the first category is extremely difficult, whereas the second one has greater quantity of information available from ten-print file searching. The traditional systems have sequential structure which suffers from a problem that the error propagates and information is blocked which means that the second stage can not use the information available at the first stage but it may suffer from the errors caused by the first stage. Moreover, conventionally, the matching process may use the following features: ridges or valleys (or both), minutia i.e. its type, location and direction, and global matching. Common Features are shown in Table 1 and Figure 1 respectively.

Table 1. Common Features for Matching Process

|                 |  |
|-----------------|--|
| Ridges          | Various patterns across fingerprint                    |
| Valleys         | Spaces between ridges                                  |
| Minutia         | Type (ending, bifurcation) Location (x,y)<br>Direction |
| Global Patterns | Arch Loop Whorl  |

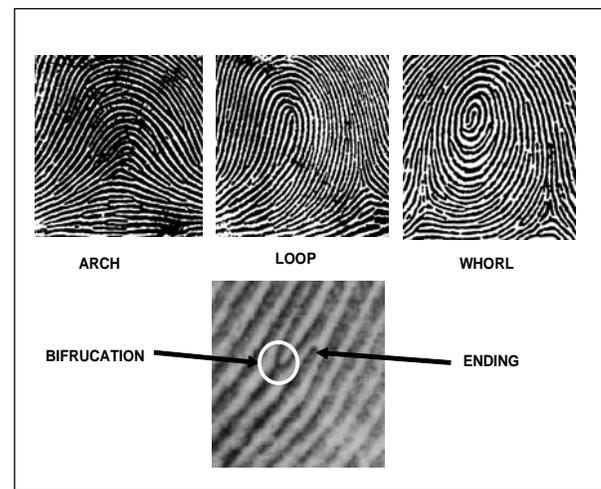


Figure 1. Common global Patterns (above), common Minutia types (below)

This work proposes new feature that can be used for matching. The proposed feature is a vector generated after the fingerprint image is compressed by Huffman Coding approach. This vector is uniquely representing the entire image. Thus, it can be effectively used for matching purposes. The proposed algorithm is illustrated in section 3 and tested in section 4. Section 5 concludes this work and gives some recommendations. Before all, brief introduction to Huffman coding is given in next section.

## II. HUFFMAN CODING

After it is presented by David A. Huffman in a 1952 paper [9], this method attracted an overwhelming amount of research and it is used in many applications such as fax machines and data compression techniques, especially image compression, which is the main contribution of this work. The two important properties of Huffman coding they are used usefully in this work are: unique prefix property, where no Huffman code is prefix of any other Huffman code, and optimality, where the Huffman code is minimum-redundancy code as shown in Huffman's 1952 paper [9].

## III. PROPOSED SYTEM

### A. Algorithm Flowchart

In Figure 3. depicts the general mechanism of the proposed system. After the fingerprint image is fed to the system, it will be converted to binary image as a preprocessing stage. Then the binary image is compressed using Huffman coding approach. A unique vector will be generated; this unique vector is then matched to all available stored vectors in codebook which is the database. Optimally, the distance between the entire vector and the corresponding vector in codebook is equal to zero, but because of the entire image has not always the exact appearance to the image that captured during the stage of building the codebook, the distance may be greater. However, the shortest distance can be used to point to the corresponding vector and then fingerprint is identified.

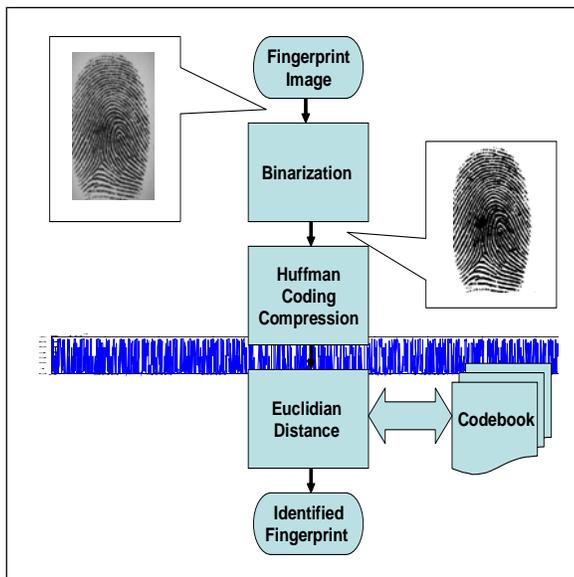


Figure 3. Flowchart for the proposed algorithm

### B. Algorithm Components

The two important components for the system shown above are the Huffman coding compression stage and the codebook. At compression stage the entire image which is binary matrix is compressed using Huffman coding algorithm to produce a vector of almost 1/16 image size. For instance, for the fingerprint image of size 560x296 or 165760 which uses 165760 bytes, the generated vector length is 10360 that is using 20720 bytes. Therefore, the memory size is decreased by 1/8, from 165760 bytes for original image to 20720 bytes for corresponding vector. To build the codebook, same procedure can be followed, by converting the database (fingerprint images) to code vectors and assigning each one to its corresponding owner. Thus the codebook contains vectors rather than images. For instance, if we have 1 million finger images for 1 million persons, this database is traditionally using 560x296x1000000 = 165760000000 bytes ~ = 154 Gigabytes, their corresponding vectors use only 20720x1000000 bytes ~ = 19 Gigabytes. These simple calculations lead to conclude that the identification rate has gained many benefits such as high speed as well as low False Acceptance Rate (FAR) / False Rejection Rate (FRR).

## IV. EXPERIMENTAL SETUP AND RESULTS

Real and synthetic compensation of database has been used. MATLAB 7 is used for experimental purpose. After building the codebook the system is now ready to be tested. Since there are three images for each person in the database, the database is categorized into three groups. Group A that builds the codebook, groups B & C are used to test the system. Randomly feeding the system by images from groups B & C. Each image is processed as shown in Figure 3.1, and then vector is generated. Euclidian distance is measured between the generated vector and all vectors available in codebook. Thus, the number of operations is equal to codebook entries. Euclidean distance between two vectors (the generated one (x) and the vector in codebook(y)) can be calculated using following formula (1):

$$d = \sum \sqrt{(x - y)^2} \quad (1)$$

Obtained results are illustrated in Table 2. The performance of Group A is perfect because the system used the same images that have been used to build its codebook; therefore, the Euclidean distances are equal to zero when the two vectors are matched. But this is not the case with group B & C because images of these groups have not exact appearance to those were used to build the codebook. Therefore, Euclidian distances have some value rather than zero. However, the minimum value (shortest distance) will be selected to point to the corresponding entry.

Table 2. Obtained results

| Test     | Group A | Group B | Group C | Average |
|----------|---------|---------|---------|---------|
| FAR      | 0.0%    | 1.2%    | 1.0%    | 0.733%  |
| FRR      | 0.0%    | 3.6%    | 4.2%    | 2.6%    |
| Accuracy | 100%    | 95.2%   | 94.8%   | 96.66%  |

#### V. CONCLUSION

This work presented new feature that can be used for fingerprint identification systems. The traditional and exist systems use features such as ridges, valleys, minutia points, and global patterns and few others to identify fingerprints, as well as they use very extreme database and the matching process matches between image and images. This work uses vector which is generated from Huffman coding compression process. Therefore, the matching process is done between code (vector) and codes (vectors) and the database is sharply decreased. The obtained results are considerably promising since very low FAR i.e. 0.733%, FRR i.e. 2.6% and high accuracy i.e. approximately 97%. The weakness of the system comes from the point of different captured environments for the images. However, this drawback may be overcome if thinning process has been adapted.

#### REFERENCES

- [1] Ruggles, T., (1996). Comparison of biometrics, Retrieved August 13, 2007: <http://bioconsulting.com/bio.htm>
- [2] Komarinski, P., Automated Fingerprint Identification Systems (AFIS), Academic Press, December 2004
- [3] Ashbourn, J., Biometrics: Advanced Identity Verification: The complete Guide, Springer, 1<sup>st</sup> edition, October 2000
- [4] Randell, N., Biometric basics, PC Magazine, April 1999, pp. 193-202.
- [5] The Center for Unified Biometrics and Sensors (CUBS), Fingerprint recognition, retrieved August 13, 2007: <http://www.cubs.buffalo.edu/fingerrecog.shtml>. Accessed January 2008
- [6] Abdurazzag Ali Aburas and Salem Ali Reheil, "New Promising Tool for Off Line Arabic Handwritten Character Recognition Based On JPEG2000 Image Compression", IEEE-3<sup>rd</sup> International Conference on Information & Communication Technologies: From theory to Applications (ICTTA'08), Damascus-Syria, April 47-11 2008
- [7] McMahon, D.H.; Johnson, G.L.; Teeter, S.L.; Whitney, C.G. (April 1975) A Hybrid Optical Computer Processing Technique for Fingerprint Identification, IEEE Transactions on Computers Volume C-24, Issue 4, Page(s):358 - 369
- [8] O'Gorman, L., An overview of fingerprint verification technologies, Information security Technical Report, 1998, vol.3(1), pp. 21-32.

- [9] Huffman, D.A., A Method for the construction of Minimum Redundancy Codes, Proceedings of the IRE, 1952, pp. 1098-1101

#### About the Authors:



**Abdurazzag Ali Aburas** received his Bachelor's degree in Computer Sciences Al Fattah University, Tripoli-Libya in 1987. He obtained his Master degree in Computer Information Technology and PhD in Digital Image Processing from Dundee University and DeMontFort University, UK in 1993 and 1997 respectively. He worked in Jordan and UAE universities for five years and he is currently an assistant professor in Electrical and Computer Engineering Department, International Islamic University Malaysia. He has more than 30 publications in different international conferences and several papers in indexed international journals. Recently, he gives consultation for IT Company as senior software developer. His areas of research interest are Human Computer Interaction, E and M Learning, wireless Communications, Digital Signal / image / video processing, Coding and Compression, Wavelets, Fractal and Image / Voice Pattern Recognition. He is a member in IEEE, HCI-UK IMA and ARISE Societies. Winner of IUM-KOE-Promising Researcher Award 2008



**Salem M A Reheil**, graduated in 1991 from Engineering Academy Tajura, Tripoli, Libya. Training courses in broadcasting computerized systems, LANs, and digital communication at RACAL College in Heckfield Place UK. Fifteen years of working experience as group leader Engineer. Currently is a Master candidate at IUM Department of Computer Engineering. The interest areas include DIP, Pattern recognition, AI and Biometric applications