American Journal of Computation, Communication and Control
2014; 1(5): 75-85
Published online January 10, 2015 (http://www.aascit.org/journal/ajccc)
ISSN: 2375-3943

Keywords
Wavelet Transform,
Compression,
Biometric,
Fingerprint,
Quantization,
Entropy Coding

Received: December 13, 2014
Revised: January 07, 2015
Accepted: January 08, 2015

Improved algorithm for biometric fingerprint image compression

Emmanuel B. S., Mu'azu M. B., Sani S. M., Garba S.

Department of Electrical and Computer Engineering, Ahmadu Bello University, Zaria, Kaduna State

Email address
sbemmanuel@yahoo.com (Emmanuel B. S.)

Citation
Emmanuel B. S., Mu'azu M. B., Sani S. M., Garba S.. Improved Algorithm for Biometric Fingerprint Image Compression. *American Journal of Computation, Communication and Control.* Vol. 1, No. 5, 2014, pp. 75-85.

Abstract
The implementation of the proposed biometric fingerprint image compression algorithm involved three stages, namely the transformation of biometric fingerprint image; non-uniform quantization of transformed image and the entropy coding which is the final stage. In order to determine the overall performance of the algorithm, Peak Signal to Noise Ratio (PSNR) and Compression Ratio (CR) were used as performance metrics. PSNR was used as a measure of the resultant image quality after compression and the Compression Ratio was used as a measure of the degree of compression achievable. A trade-off was made between the achievable compression ratio and the realizable image quality which is a function of the achievable PSNR in the overall compression process. The overall performance of the proposed compression algorithm achieved an improvement in terms of compression ratio of 20:1 over the existing compression algorithms for biometric applications which achieved a compression ratio of 15:1. The improvement was largely due to the new approach employed in this research work.

1. Introduction

A biometric fingerprint compression algorithm consists of two distinct structural blocks, namely: An image encoder and decoder [1]. The encoding process is the forward process of compressing the source image while the decoding process is the reverse process of reconstructing the image signal from the encoded bitstreams. The source image encoder is responsible for reducing or eliminating any coding, inter-pixel and psychovisual redundancies in the source image.

In the first stage of the source encoding process, the source image is transformed into a domain designed to reduce inter-pixel redundancies. This operation reduces the amount of data required to represent the image which directly results in data compression and the process is reversible [1]. In addition, the representation of an image by a set of transformed coefficients makes its inter-pixel redundancies more accessible for reduction to achieve compression.

The second stage of the encoding process is the quantization stage where lossy compression is achieved which results in the reduction of the quality of quantized image output. The quantization operation at this stage reduces the psychovisual redundancy of the source image and the process is irreversible [1]. In the third stage, of the source encoding process is the entropy coding which creates a fixed-length or variable-length codes to represent the quantizer output and maps the output to some codewords on the basis of the data source's statistical characteristics. The entropy coding is a lossless compression method which provides an effective mechanism to eliminate coding redundancy [2]. The primary objective of this lossless compression is to decompose a

data set into a sequence of events or symbols, then to encode the symbols using as few bits as possible. The idea is to assign short codewords to more probable symbols and longer codewords to less probable symbols [3]nd Vitter, 1994). Data compression can be achieved whenever some source symbols are more likely to occur than others. Entropy coding optimally compresses the redundancy present in source input data and make its encoding rate tend to the entropy of the source data [2]. Upon completion of the entropy coding process, the input source image has been processed to reduce or remove each of the three redundancies present in the image.

Source fingerprint image decoding process involves only two stages: a symbol or entropy decoder and an inverse discrete wavelet transform and they are performed in reverse order of the encoding process. Since the quantization process is a non-invertible operation and resulted in irreversible information loss, the idea of inverse quantization is not applicable in a lossy compression process.

2. Literature Review

This section is divided into two sub-sections. Section 2.1 describes the fundamental concepts pertinent to this research work. In Section 2.2, a critical review of similar research works is presented in chronological order.

2.1. Review of Fundamental Concepts

This section reviews the fundamental concepts pertinent to this research work.

2.1.1. Wavelet Transform in Two Dimensions

For analytic transformation of image signal, a two-dimensional (2-D) discrete wavelet transform is used which can easily be extended from a one-dimensional (1-D) wavelet transform. To achieve this, one 2-D scaling function, $\varphi(x,y)$, and three 2-D wavelets: $\Psi^H(x, y)$, $\Psi^V(x, y)$, and $\Psi^D(x, y)$ are required. Each is the product of 1-D scaling function φ and corresponding 1-D wavelets φ as shown [1]:

$$\varphi(x,y) = \varphi(x)\varphi(y) \quad (2.1)$$

$$\Psi^H(x,y) = \Psi(x)\varphi(y) \quad (2.2)$$

$$\Psi^V(x,y) = \varphi(x)\Psi(y) \quad (2.3)$$

$$\Psi^D(x,y) = \Psi(x)\Psi(y) \quad (2.4)$$

Equation 2.1 defines the separable scaling function, $\varphi(x,y)$. Equations 2.2 to 2.4 define the wavelet functions that measure the functional variations of intensity or grayscale for images along different directions: Ψ^H defines variation along columns (horizontal edges); Ψ^V defines variation along rows (vertical edges); Ψ^D defines variation along diagonals.

Given separable 2-D scaling and wavelet functions, 2-D DWT can be defined. First, we define the scaled and translated or shifted basis functions are defined as follows:

$$\varphi_{j,m,n}(x,y) = 2^{\frac{j}{2}}\varphi(2^j x - m, 2^j y - n) \quad (2.5)$$

$$\Psi^i{}_{j,m,n}(x,y) = 2^{\frac{j}{2}}\Psi^i(2^j x - m, 2^j y - n), \ \ i = \{H,V,D\} \quad (2.6)$$

Where, i = directional wavelet index

Therefore, 2-D DWT of function f(x, y) of size MxN is given by [1]:

$$w_\varphi(j_0,m,n) = \frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\,\varphi_{j_0,m,n}(x,y) \quad (2.7)$$

$$w^i{}_\Psi(j,m,n) = \frac{1}{\sqrt{MN}}\sum_{x=0}^{M-1}\sum_{y=0}^{N-1} f(x,y)\,\Psi^i{}_{j,m,n}(x,y), \ \ i = \{H,V,D\} \quad (2.8)$$

Where,

j_0 = Arbitrary starting scale ($j_0 = 0$)

$w_\varphi(j_0,m,n)$ = Approximation coefficients for f(x, y) at scale j_0

$w^i{}_\Psi(j,m,n)$ = Horizontal, vertical and diagonal details coefficients at scales $j \geq j_0$

$$M = N = 2^j, for\ j = 0,1,2,\dots,j-1$$

$$m,n = 0,1,2,\dots,2^j - 1$$

Given w_φ and $w^i{}_\Psi$, f(x, y) can be obtained from 2-D Inverse DWT as follows [1]:

$$f(x,y) = \frac{1}{\sqrt{MN}}\sum_m\sum_n w_\varphi(j_0,m,n)\,\varphi_{j_0,m,n}(x,y) + \frac{1}{\sqrt{MN}}\sum_{i=H,D,V}\sum_{j=j_0}^{\infty}\sum_m\sum_n w^i{}_\Psi(j,m,n)\,\Psi^i{}_{j,m,n}(x,y) \quad (2.9)$$

It should be noted that since image signal has two dimensional data structure, 2-D DWT will be implemented for fingerprint image transformation in this research work.

2.1.2. Lloyd Max Non-Uniform Quantization

Lloyd-Max quantization procedure defines an optimal approach to non-uniform quantization process. The basic idea of the Lloyd-Max quantization is to find the decision boundaries and reconstruction levels that minimize the mean square quantization error (MSQE). This approach solves the problem of finding the decision boundaries $\{b_j\}$ and the reconstruction levels $\{y_j\}$ given N-level quantizer Q(x) on [a, b] so that the MSE given by Equation 2.8 can be minimized [4].

$$MSQE = \sum_{j=1}^{N}\int_{b_{j-1}}^{b_j}\left(x - y_j\right)^2 f_x(x)dx \quad (2.10)$$

Where:

$f_x(x)$ = The Probability Density Function (PDF) of the source input, X

b_j = Decision boundary

y_j = Reconstruction level

N = Quantization level

Given the PDF of an input source x, the approach to designing the best non-uniform quantizer is to determine the values of the decision boundary and the reconstruction level

that minimize the mean square quantization error (MSQE).

Setting the derivative of Equation 2.10 with respect to y_j to zero, and solving for y_j:

$$\frac{\partial(MSQE)}{\partial y_j} = 0$$

$$\sum_{j=1}^{N} \frac{\partial}{\partial y_j} [\int_{b_{j-1}}^{b_j} (x - y_j)^2 f_x(x)dx] = 0$$

Using the Chain Rule of Differentiation: $\frac{dy}{dx} = \frac{dy}{du} * \frac{du}{dx}$

$$-2 \int_{b_{j-1}}^{b_j} (x - y_j) f_x(x)dx = 0$$

$$\int_{b_{j-1}}^{b_j} [2y_j f_x(x)dx - 2x f_x(x)dx] = 0$$

$$2 \int_{b_{j-1}}^{b_j} y_j f_x(x)dx - 2 \int_{b_{j-1}}^{b_j} x f_x(x)dx = 0$$

$$2 \int_{b_{j-1}}^{b_j} y_j f_x(x)dx = 2 \int_{b_{j-1}}^{b_j} x f_x(x)dx \quad j = 1, 2 \dots N$$

$$y_j = \frac{\int_{b_{j-1}}^{b_j} x f_x(x)dx}{\int_{b_{j-1}}^{b_j} f_x(x)dx} \tag{2.11}$$

The reconstruction point for each quantization interval is the centroid of the probability distribution of the interval. Taking the derivative Equation 2.8 with respect to b_j and setting it equal to zero, an expression for b_j is obtained as follows:

$$\frac{\partial(MSQE)}{\partial b_j} = 0$$

$$\frac{\partial}{\partial b_j} \{\int_{b_{j-1}}^{b_j} (x - y_j)^2 f_x(x)dx + \int_{b_j}^{b_{j+1}} (x - y_j)^2 f_x(x)dx\} = 0$$

$$(b_j - y_{j-1})^2 f_x(b_j) + (b_j - y_j)^2 f_x(b_j) = 0$$

$$(b_j - y_{j-1})^2 f_x(b_j) = -(b_j - y_j)^2 f_x(b_j)$$

$$(b_j - y_{j-1}) = \pm(b_j - y_j)$$

$$(b_j - y_{j-1}) = -(b_j - y_j)$$

$$b_j - y_{j-1} = -b_j + y_j$$

$$2b_j = y_j + y_{j-1}$$

$$b_j = \frac{y_j + y_{j-1}}{2} \tag{2.12}$$

Lloyd-Max algorithm iteratively solves for the values of b_j and y_j that minimize the MSQE.

2.1.3. Arithmetic Coding and Image Compression

Compression applications employ a wide variety of techniques and have different degrees of complexity, but share certain processes in common. Figure 2.1 shows a process block for data compression. These processes depend on the data type, and the stages may be in different order or combination. The pre-processing stage often includes transformation and quantization processes. The next stage, source modeling is used to account for variations in the statistical properties of the data. It is responsible for gathering statistics and identifying data contexts that make the source models more accurate and reliable [5]. What most compression systems have in common is the fact that the final process is the entropy coding, which is the process of representing information in the most compact form without any data loss [5]. In lossless compression scheme, entropy coding is responsible for the entire compression process. However, in lossy compression scheme, it complements what has been accomplished by previous stages such as transformation and quantization stages as it is the case in this research work.

Arithmetic coding stands out among existing entropy coding scheme in terms of effectiveness and versatility, since it is able to work most efficiently in the largest number of circumstances and purposes.

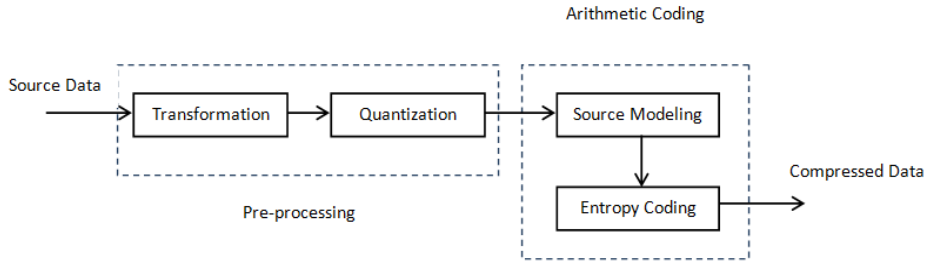

Figure 2.1. *Arithmetic entropy coding as the final stage of a compression process*

2.1.4. Encoder and Decoder

Arithmetic coding scheme codes one data symbol at a time, and assigns to each symbol a real-valued number of bits [5]. Code value or code word representation means coded messages mapped to real numbers in the interval [0, 1).The code value, v of a compressed data sequence is the real number with fractional digits equal to the sequence's symbols [5]. The code value representation can be used for any coding system and it provides a universal way to represent large amounts of information independent of the set of symbols used for the coding [5].

Fundamentally, the arithmetic encoding process consists of

creating a sequence of nested intervals in the form [5]:

$$\emptyset_k(S) = [\alpha_k,\ \beta_k), \quad k = 0, 1, \dots, N \tag{2.13}$$

Where:

$\emptyset_0(S)$ = encoded source sequence

$\alpha_k\ and\ \beta_k$ = real numbers such that $0 \le \alpha_k \le \alpha_{k+1}, and\ \beta_{k+1} \le \beta_k \le 1$

For simplicity, a new notation for arithmetic coding interval is represented in the form [b, l), where b is called the base or starting point of the interval, and l is the length of the interval. The relationship between the two interval notations is [5]:

$$[b, l) = [\alpha, \beta), \quad if\ b = \alpha\ and\ l = \beta - \alpha \tag{2.14}$$

The intervals used during the arithmetic coding process are defined by the set of recursive equations [5]:

$$\emptyset_0(S) = [b, l) = [0, 1) \tag{2.15}$$

$$\emptyset_0(S) = [b_k,\ l_k), = [b_{k-1} + c(s_k)l_{k-1},\ p(s_k)l_{k-1}),\ k = 1, 2, \dots, N \tag{2.16}$$

Where:

$\emptyset_0(S)$ = encoded source sequence

S = data symbol

p = symbol probability

c = cumulative distribution

b = interval base

l = interval length

The properties of intervals guarantee that $0 \le b_k \le b_{k+1} < 1, and\ 0 < l_{k+1} \le l_k \le 1$. The encoding process defined by Equations 2.14 and 2.15 is called Elias coding [5]. Equation 2.16 depicts that the size of the symbol's subinterval $[b_k,\ l_k)$ is proportional to the estimated probability of occurrence p of the symbol in accordance to the cumulative distribution function c of the source input. Put differently, the length of the final interval in the arithmetic coding subdivision process is equal to the product of the probability p of the particular sequence of symbols in the source input. The final interval is then assigned a codeword in bits.

In arithmetic coding, to decode the source sequence, the encoding process is reversed. The decoder input set is determined to restore the original sequence. In other words, the decoded sequence is determined solely by the code value v' used to represent the encoded sequence. Therefore, the decoded sequence is represented as [5][6]:

$$\hat{\mathrm{S}}(v') = \{\hat{\mathrm{s}}_1(v'), \hat{\mathrm{s}}_2(v'), \dots, \hat{\mathrm{s}}_N(v')\} \tag{2.17}$$

In the decoding process, code value v' is the set of codewords used for decoding the correct symbol sequence $\hat{\mathrm{S}}(v') = S$ of the encoded source input. The decoding process recovers the data symbols in the same sequence that they were encoded coded. The numerical solution to the decoding process is determined by defining a sequence of code values $\{\tilde{v}_1, \tilde{v}_2, \dots, \tilde{v}_N\}$. Starting with $\tilde{v}_1 = \tilde{v}$, $\hat{\mathrm{s}}_k$ and $\tilde{v}_k$ are sequentially determined and $\tilde{v}_{k+1}$ is computed from $\hat{\mathrm{s}}_k$ and $\tilde{v}_k$. Equations 2.18 to 2.21 define the numerical solution for decoding process [5]:

$$\tilde{v}_1 = \tilde{v} \tag{2.18}$$

$$\hat{\mathrm{s}}_k(v') = s\ for\ c(s) \le c(s+1),\ \ k = 1, 2, \dots, N. \tag{2.19}$$

$$\tilde{v}_{k+1} = \frac{\tilde{v}_k - c(\hat{\mathrm{s}}_k(v'))}{p(\hat{\mathrm{s}}_k(v'))}, \quad k = 1, 2, \dots, N-1. \tag{2.20}$$

$$\tilde{v}_k = \frac{v' - b_{k-1}}{l_{k-1}} \tag{2.21}$$

Where:

$\hat{\mathrm{S}}(v')$ = Decoded symbol sequence

$\tilde{v}_k$ = Sequence of decoded code values

v' = Encoded code values of the source input

S = Numerical values of symbols that satisfy the inequality, $c(s) \le c(s+1)$

In summary, the decoding process recovers the decoded sequence $\hat{\mathrm{S}}(v')$ by sequentially determining the sequence of decoded code values $\tilde{v}_k$ from the encoded code values v' of the source input.

2.1.5. Cumulative Distribution Function (CDF)

Arithmetic coding essentially entails the determination of the cumulative distribution function (CDF) of the probability of a sequence of symbols. The length of the final subinterval in the arithmetic coding process is equal to the product of the cumulative probabilities of the individual source symbol. The cumulative distribution function (CDF) is given by [3]:

$$CDF(m_k) = \sum_{k=1}^{m-1} P_k \tag{2.22}$$

Where:

P_k = probability of individual symbol

m_k = Length of symbol

2.1.6. Entropy Coding Efficiency

Source coding efficiency is defined as the ratio (in percentage) of the entropy of a source to the average code length and it is given by [7]:

$$E(S, L) = \frac{H(S)}{l'(S,L)} * 100\% \tag{2.23}$$

Where:

$E(S, L)$ = Coding Efficiency

$H(S)$ = Source entropy

$l'(S, L)$ = Average code length

The codeword is the number of bits used to represent a source's symbol in an entropy coding process. Given the length of codewords $L = (l_1, l_2, \dots, l_n)$ resulting from an entropy coding process of a source $S = (s_1, s_2, \dots, s_n)$, the average length of the code is given by [7]:

$$l' = \sum_{j=1}^{n} s_j l_j \tag{2.24}$$

Where:

s_j = Symbol's probability of occurrence

l_j = Length of codeword

The entropy provides a theoretical bound of the minimum number of bits that can be used to represent source data.

Given the source $S = (s_1, s_2, \dots, s_n)$, the source entropy is given by [7]:

$$H(S) = \sum_{j=1}^{n} s_j log_2 s_j \quad (2.25)$$

Where:

s_j = Symbol's probability of occurrence

Information theory states that the best a lossless symbolic compression scheme can achieve is to encode a source with an average number of bits equal to the entropy of the source. However, when the difference between the entropy and the average length of the code increases, the code efficiency decreases. It follows therefore that an entropy coding process is optimal, if the code efficiency reaches 100%. In other words, a code is optimal if the average length of the codewords equals the entropy of the source [7].

2.1.7. Validation and Performance Measures

Two measures will be used to validate the perceptual quality of the compressed fingerprint images, namely: Compression Ratio (CR) and Peak Signal to Noise Ratio (PSNR). The PSNR metric is normally used to measure the performance of image coding algorithm.

i) Compression Ratio: The compression ratio (CR) or C_r is used as a performance measure to assess the efficiency of a compression algorithm and this is given by [2]:

$$C_r = \frac{\sum_{i=1}^{N}\sum_{j=1}^{N} r_{b(i,j)}}{\sum_{i=1}^{N}\sum_{j=1}^{N} r_{c(i,j)}} \quad (2.26)$$

Where:

r_b = bits per pixel of original image

r_c = bits per pixel of compressed image

ii) The Peak Signal to Noise Ratio (PSNR): It represents a measure of the peak error and is expressed in decibels. It is defined by [8]:

$$PSNR = 20 \log\left(\frac{2^B - 1}{MSE}\right) \quad (2.27)$$

Where: B is the bit depth of the image. For an 8-bit image the PSNR is given by [8]:

$$PSNR = 20 \log\left(\frac{255}{MSE}\right) \quad (2.28)$$

The higher the *PSNR value*, the closer the quality of the compressed or reconstructed image to the original source image. Typical values for lossy compression of an image are between 30 and 50 dB and when the *PSNR* is greater than 40 dB, then the two images are indistinguishable. Images with PSNR higher than 30 dB are considered to be perceptually lossless [8]. The Mean Square Error (MSE): It represents the mean squared error between the compressed and the original image. The lower the MSE value, the lower the distortion or degradation incurred in the compression process. For a source image $X_{i,j}$ of size N x M pixels and a reconstructed/processed image $Y_{i,j}$ also of size N x M, the Mean Square Error, MSE, is given by [8]:

$$MSE = \frac{1}{MN}\sum_{j}^{N-1}\sum_{i}^{M-1}(X_{i,j} - Y_{i,j})^2 \quad (2.29)$$

2.2. Review of Similar Works

The approach adopted for a critical review of similar research works is based on the compliance with the requirements of a standard compression scheme which involved three processes, namely: transformation, quantization and entropy coding. In addition, the review is based on the performance metrics achieved which includes: compression ratio (CR) and peak signal to noise ratio (PSNR). Table 2.1 summarizes the reviewed similar works done on wavelet based image compression methods with their critical analysis.

Table 2.1. *Summary of the Review of Similar Works Based on Wavelet Image Compression Methods*

Reference	Transformation	Quantization	Coding scheme	Critique
Khuwaja and Tolba, [9]	Coiflet and Biorthogonal wavelets	Not implemented	Not implemented	Compression ratio was not estimated to determine the level of achievable compression. The work is not compliant with standard compression scheme requirement.
Winger and Venetsanopoulos, [10]	Cooklet and CDF	Not implemented	Not implemented	The source data used for the analysis were the MATLAB experimental Lena, Barbara and Goldhill images. The work did not implement the quantization and entropy coding stages of a standard lossy compression scheme.
Li and Bayoumi, [11]	Wavelet block transformation	Uniform scalar quantization	EBCOT encoding	Complex algorithm due to EBCOT implementation.
Khalifa, [12]	Daubechies wavelet-based decomposition	Vector quantization and DPCM	Huffman	High computation cost due to codebooks generated from Vector quantization and Huffman coding
Sudhakar et al, [13]	Multiwavelet filters	Not implemented	SPIHT based on Embedded Zerotree encoder	A higher PSNR value can be obtained if a quantization

Reference	Transformation	Quantization	Coding scheme	Critique
				scheme is implemented.
Hsin et al, [14]	Wavelet transform	Not implemented	Context-based EBCOT encoding based on arithmetic coder	High computation cost due to EBCOT. Compression ratio was not estimated to determine the level of achievable compression. The work is not compliant with standard compression scheme requirement.
Chang et al, [15]	DWT based Daubechies wavelet	Uniform scalar quantization	EBCOT encoding based on arithmetic coder	High computation cost of complex algorithm due to EBCOT
Mushen et al, [16]	Not covered	Vector quantization	Not implemented	High computation cost due to codebook resulting from vector quantization. Compression ratio was not estimated to determine the level of achievable compression. The work is not compliant with standard compression scheme requirement.
Rawat et al, [17]	Biorthogonal wavelet transform	SOFM based vector quantization with code book	SPIHT based on Embedded Zerotree encoder	Complex algorithm due to vector quantization
Haddad et al [18]	Curvelet and wave atom	Not implemented	Not implemented	The compression process fell short of the requirement of a standard compression scheme.
Zhao and Wang [19]	Contourlet transform	Uniform scalar quantization	Arithmetic coding	The compression ratio was not estimated and therefore, there is no basis for determining the extent of compression in the process.
Kumar et al, [20]	DWT based on Biorthogonal wavelets	Not implemented	SPIHT based on Embedded Zerotree encoder	The compression ratio was not estimated and therefore, there is no basis for determining the extent of compression in the process. Poor preservation of biometric features by SPIHT
Ashok et al, [21]	Wave atom decomposition	Vector quantization	Arithmetic entropy coding	Complex algorithm due to codebook generated from vector quantization process
Krishnaiah et al, [22]	5/3 wavelet transform	Not implemented	SPIHT based on Embedded Zerotree encoder	Low compression ratio due lossless scheme employed.
Muhsen et al, [23]	9/7 Wavelet filter	Vector quantization and re-quantization with codebook	Run-length encoding	High computation cost and complex algorithm due to codebook generation resulting from vector quantization is a challenge
Gangwar, [24]	Haar wavelet	Not implemented	Not implemented	Failed to comply with compression standards
Shanavaz et al, [25]	Daubechies Wavelet lifting	Not implemented	SPIHT based on Embedded Zerotree encoder	Complex algorithm
Shakhakarmi, [26]	Wavelets multiscale analysis	Not implemented	Not implemented	Poor comparative study with FFT and DCT
Libert et al, [27]	WSQ Versus JPEG2000	N/A	N/A	JPEG2000 achieved poor biometric quality. WSQ is limited in compression ratio
Islam et al, [28]	Coiflet wavelet implemented with global threshold strategy	Not implemented	Not implemented	Compression ratio was not estimated to determine the level of achievable compression. The work is not compliant with standard compression scheme requirement.
Singla et al, [29]	Haar, Coiflet, Daubechies, Biorthogonal wavelets	Not implemented	Not implemented	Wavelets were implemented with global threshold strategy.
Selvakumarasamy,	Biorthogonal, Symlet,	Not implemented	Not implemented	The entropy coding process of

Reference	Transformation	Quantization	Coding scheme	Critique
et al [30]	Daubechies and Coiflet wavelets for lossless compression			a standard lossless compression system was not implemented. Lossy compression scheme is a better approach to achieving a much higher compression ratio.
Guangqi et al [31]	Wave Atom dictionary for sparse representation	Uniform scalar quantization	Arithmetic coding	Uniform scalar quantization used in this work is not the most efficient quantization scheme available.

The reviewed research works employed wavelet filter banks such as Haar, Symlet, Wave Atom, Coiflet implemented with global thresholding, Daubechies and their variants for image decomposition and these have been identified to possess limited properties that can achieve optimal image compression. Vector quantization and uniform scalar quantization schemes were implemented in most of the compression scheme under review. Additionally, the existing techniques generated codebooks or lookup tables for image coding and this invariably increased the complexity of the algorithms and hence, increased computation or implementation cost. Therefore, the proposed biometric fingerprint image compression algorithm will employ Coiflet wavelet implemented with level-dependent threshold strategy for image transformation as against the global threshold strategy used in the existing wavelet-based image transformation scheme to achieve improved energy de-correlation for a more efficient compression process. In addition, non-uniform quantization scheme will be used at the quantization stage as against the uniform scalar and vector quantization methods employed in the existing compression techniques. This combined approach, to the best of my understanding based on all the reviewed literature, is novel as it has not been applied for any image compression standard so far.

3. Statement of Problems

From the survey of the existing wavelet-based image compression methods, the problems that have been identified include: the limitation of WSQ standard to a compression ratio of 15:1 which could be improved with better algorithm. High complexity of image encoding process of the existing techniques is also a problem. Most of the existing methods require the generation of codebooks or lookup tables which require additional computational cost for implementation. More significantly, the variant of Daubechies wavelet, that is, Cohen Daubechies Feauveau (CDF) wavelet basis adopted for image decomposition and de-correlation in JPEG2000 and WSQ standards lacks the vital property of symmetry necessary for perfect reconstruction of image signal. Additionally, significant degradation in the biometric features of fingerprint at compression ratio higher than 15:1 remains a major challenge. For instance, at compression ratio higher than 15:1, the WSQ compression technique starts to yield unsatisfactory result. Therefore, the investigation of an efficient compression method that can significantly reduce fingerprint image size while preserving its biometric properties (the core, ridge endings and bifurcations) with compression ratio higher than 15:1 is justified.

4. Aim and Objectives

The aim of this presentation is to carry out the entropy coding of quantized source fingerprint image to complement the lossy compression process. The objectives are as follows:

i) Transformation of the source fingerprint image to lower the correlation of its pixel values and eliminate interpixel redundancy;
ii) Representation of large image pixel values with smaller quantized values to achieve efficient lossy compression of fingerprint image;
iii) Representation of quantized fingerprint image symbols with binary codewords for optimal storage;
iv) Evaluation of the overall performance of proposed compression algorithm on the basis of image quality measure;
v) Computation of the compression ratio for the compression algorithm.

5. Methodology

This progress report covered stage three to five of the proposed research methodology, which include:

i) Source fingerprint image transformation with Coiflet wavelet filters;
ii) Non-uniform quantization of transformed source coefficients using Lloyd-Max algorithm;
iii) Arithmetic entropy coding of quantized fingerprint source symbols;
iv) Reconstruction of source fingerprint image from compressed bit-stream;
v) Validation and evaluation of the performance of the proposed compression algorithm using PSNR and Compression Ratio (CR) performance metrics.

For the proposed algorithm to work, raw-bits, uncompressed fingerprint dataset is required. Hence, the need to employ the use of the National Institute of Technology and Standards (NIST) fingerprint datasets [32]. The proposed compression process involved the transformation of biometric fingerprint image using Coiflet-based discrete wavelet transform to reduce interpixel redundancy; non-uniform quantization of transformed image using Lloyd-Max algorithm to reduce psychovisual

redundancy and entropy coding into bit-streams as the final encoding stage using arithmetic coding to reduce coding redundancy. The transformed image formed the input to the non-uniform quantization process and the quantized image formed the input to the entropy coding process. In order to reconstruct the compressed fingerprint image from the encoded bit-streams, the processes of transformation, quantization and entropy encoding were reversed with arithmetic entropy decoding; de-quantization and inverse discrete wavelet transform (IDWT). The forward and reverse processes are depicted in the program flowchart as shown in Figure 5.1 and the algorithm was implemented in MATLAB image processing toolbox.

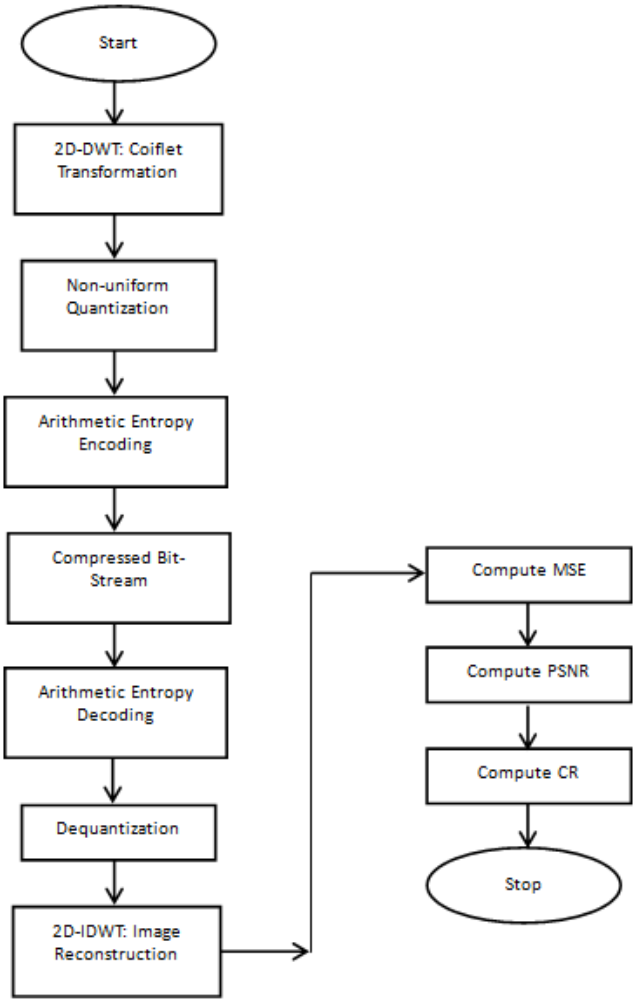

Figure 5.1. *Program Flowchart of the Proposed Coiflet-based Lossy Compression Algorithm for Biometric Fingerprint Image*

In order to determine the performance of the proposed biometric fingerprint compression algorithm, Peak Signal to Noise Ratio (PSNR) and Compression Ratio (CR) were used as performance metrics. PSNR was used as a measure of the resultant image quality after compression and the Compression Ratio was used as a measure of the degree of compression achievable. In the performance analysis a trade-off was achieved between the achievable compression ratio and the allowable degradation which is a function of achievable PSNR in the compression process. Compression ratio values were computed with the intent of maintaining compressed fingerprint image quality consistent with the requirements of biometric application. The results of the performance metrics are shown in Tables 6.1 and 6.2. The original 8-bit source fingerprint images were transformed using the Coiflet wavelets at a decomposition level of three as shown in Figure 5.2. Compression ratio of 15:1, 20:1, 40:1 and 80:1 were applied and the corresponding values of the PSNR ratio as a measure of image quality after compression were estimated as shown in Figures 5.3 to 5.5.

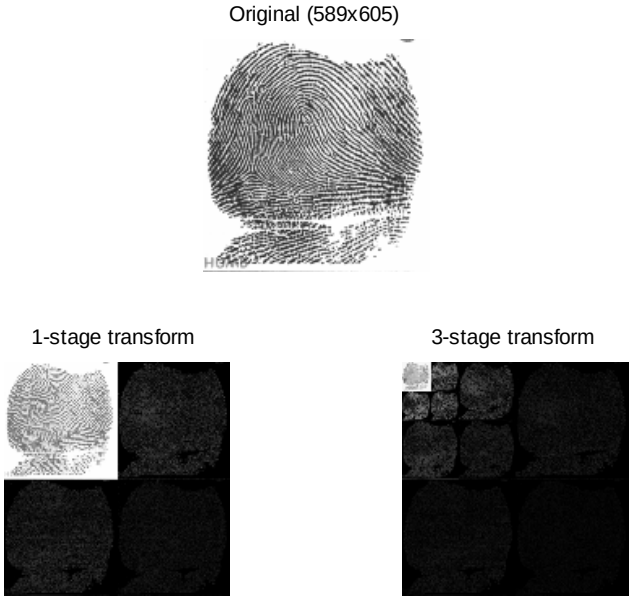

Figure 5.2. *3-level Decomposition of Fingerprint Image in the Compression Process*

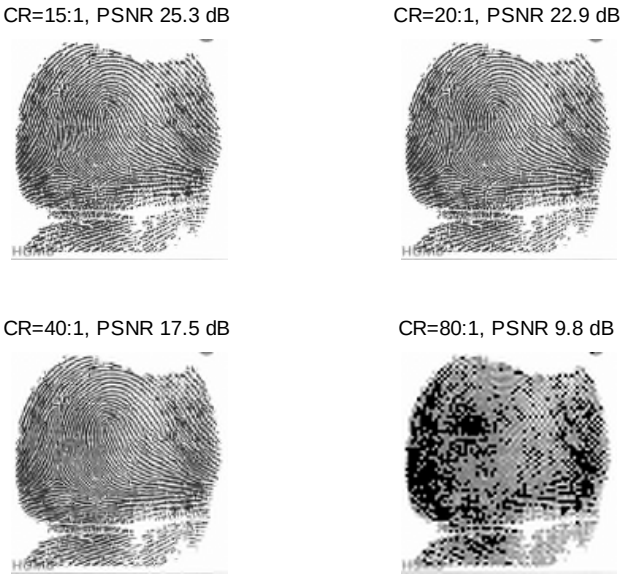

Figure 5.3. *Compression of Fingerprint Image with Values of Compression Ratio and their Corresponding PSNR Values*

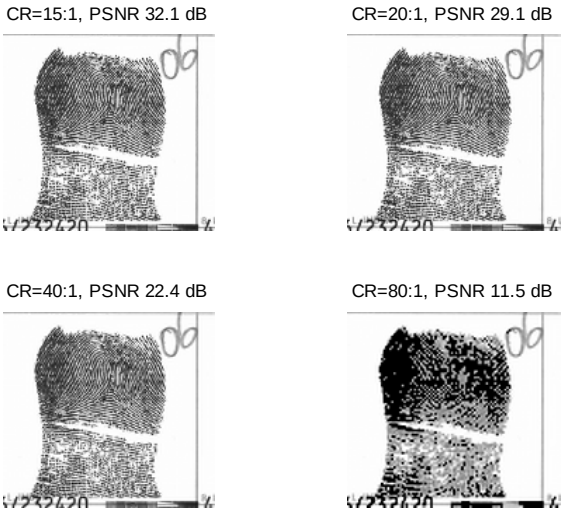

Figure 5.4. *Compression of Single Fingerprint Image with Values of Compression Ratio and their Corresponding PSNR Values*

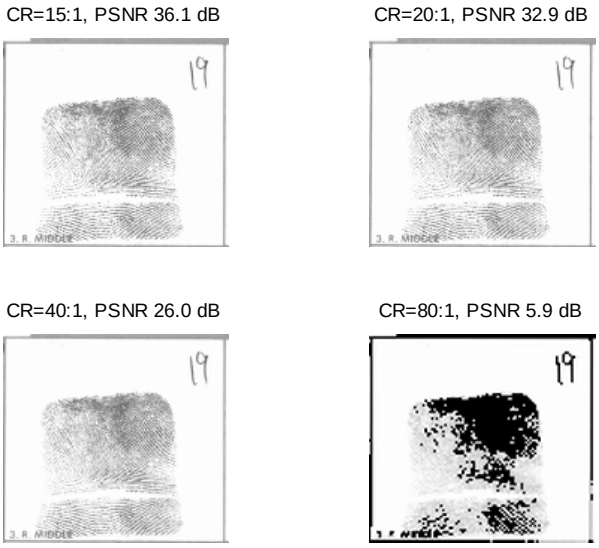

Figure 5.5. *Compression of Fingerprint Image with Values of Compression Ratio and their Corresponding PSNR Values*

6. Results and Discussions

Table 6.1. *Performance Analysis of the Proposed Compression Algorithm on Fingerprint Images at Different Compression Ratios*

Source Images	Size of Original Image	Size of Quantized Image	Compression Ratio (CR)	PSNR (dB)
Cmp00001.pgm	356360	23757	15:1	25.30
Cmp00001.pgm	356360	17818	20:1	22.90
Cmp00001.pgm	356360	8909	40:1	17.50
Cmp00001.pgm	356360	4455	80:1	9.80
Cmp00002.pgm	638991	42599	15:1	32.10
Cmp00002.pgm	638991	31949	20:1	29.10
Cmp00002.pgm	638991	15974	40:1	22.40
Cmp00002.pgm	638991	7987	80:1	11.50

The performance analysis results of the proposed fingerprint compression algorithm on the basis of compression ratio and peak signal to noise ratio are as shown in Tables 6.1 and 6.2.

Figure 6.1 represents the results obtained from the application of the proposed fingerprint compression algorithm on NIST fingerprint dataset for compression ratios of 15:1, 20:1, 40:1 and 80:1 and the corresponding values of the PSNR ratio as a measure of compressed image quality. It was observed that as the compression ratio increased for a particular source image from 15:1 to 80:1, the corresponding values of PSNR decreased from 25.3 dB for CR of 15:1 to 11.50 dB for CR of 80:1 and the result for other fingerprint images followed the same trend. This is evident in the plot of CR against PSNR as shown in Figures 6.1 and 6.2.

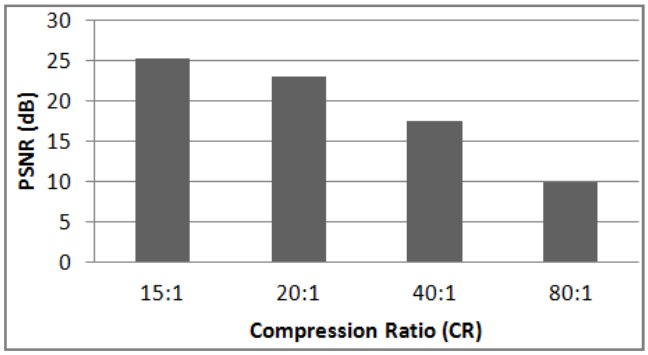

Figure 6.1. *Bar chart Plot of PNSR Values against their corresponding Compression Ratio (CR) for fingerprint image cmp00001.pgm*

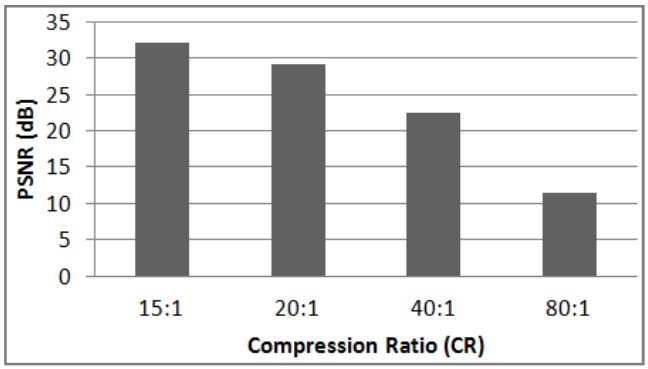

Figure 6.2. *Bar chart Plot of PNSR Values against their corresponding Compression Ratio (CR) for fingerprint image cmp00002.pgm*

The significance of these results is that as the CR value increased beyond 20:1, the fingerprint images began to lose their biometric features due to increased degradation which resulted from the compression process. However, at compression ratio of 20:1, the compressed fingerprint images still retained their biometric attributes and this was visually evident from the compressed image output at the minimum PSNR value of 22.90 dB. The existing fingerprint algorithm such Joint Photographic Expert Group (JPEG2000) and Wavelet Scalar Quantization (WSQ) were reported to yield unsatisfactory results beyond a compression ratio of 15:1 and at CR of 20:1, the biometric features of the compressed images were completely lost. Therefore, all the NIST fingerprint dataset were compressed at a compression ratio of 20:1 and their PSNR values were tabulated as shown in Table 6.2.

The minimum PSNR value was 22.9 dB and the maximum PSNR value is 35.50 dB. The variation in the values of PSNR for the NIST dataset at the same compression ratio was due to different level of source fingerprint image quality which was visually evident in the compressed output display. This is because the source fingerprint images were acquired at different levels of image quality. The presence of background noise also accounted for the variation in the measurement of overall image quality in the compression process. Generally, the overall performance of the proposed compression algorithm achieved an improvement in terms of compression ratio of 20:1 over the existing algorithms which achieved a compression ratio of 15:1.

Table 6.2. *PSNR Values Obtained for a Compression Ratio of 20:1*

Source Images	Size of Original Image	Size of Quantized Image	Compression Ratio (CR)	PSNR (dB)
Cmp00001.pgm	356360	17818	20:1	22.90
Cmp00002.pgm	638991	31949	20:1	29.10
Cmp00003.pgm	638991	31949	20:1	32.90
Cmp00004.pgm	612895	30645	20:1	33.30
Cmp00005.pgm	638991	31949	20:1	35.50
Cmp00006.pgm	638991	31949	20:1	28.00
Cmp00007.pgm	347725	17386	20:1	25.60
Cmp00008.pgm	600015	30001	20:1	28.80
Cmp00009.pgm	347151	17358	20:1	25.10
Cmp00010.pgm	197265	9863	20:1	25.40
Cmp00011.pgm	440253	22012	20:1	25.40
Cmp00012.pgm	369471	18473	20:1	24.90
Cmp00013.pgm	350904	17545	20:1	24.90
Cmp00014.pgm	269363	13468	20:1	25.70
Cmp00015.pgm	292135	14606	20:1	30.40
Cmp00016.pgm	504843	25242	20:1	31.70
Cmp00017.pgm	347001	17350	20:1	26.00
a001.pgm	1520081	76004	20:1	27.10
a002.pgm	1460729	73036	20:1	25.50
a018.pgm	1534580	76729	20:1	30.10
a039.pgm	458175	22908	20:1	24.90
a070.pgm	1605297	80264	20:1	26.50
a076.pgm	612848	30642	20:1	27.60
a089.pgm	1574914	78745	20:1	24.10
a107.pgm	1862081	93104	20:1	27.10
a129.pgm	365777	18288	20:1	27.10
a165.pgm	387047	19352	20:1	26.30
b124.pgm	1510456	75522	20:1	25.50
b157.pgm	1572731	78636	20:1	26.30
b186.pgm	371349	18567	20:1	27.10

7. Conclusion

In conclusion, the implementation of the proposed biometric fingerprint image compression algorithm involved:

(i) the transformation of biometric fingerprint image to reduce interpixel redundancy;

(ii) non-uniform quantization of transformed image to reduce psychovisual redundancy;

(iii) arithmetic entropy coding to reduce coding redundancy.

In order to determine the overall performance of the algorithm, Peak Signal to Noise Ratio (PSNR) and Compression Ratio (CR) were used as performance metrics. PSNR was used as a measure of compressed image quality and it was estimated as a function of the mean square error (MSE) which is a measure of the image distortion that resulted from the compression process. The Compression Ratio was used as a measure of the degree of compression achievable to evaluate the performance of the proposed compression algorithm against the achievable compression ratio for the existing fingerprint compression schemes. In the system performance analysis, a trade-off was made between the achievable compression ratio and the allowable degradation which is a function of achievable PSNR in the compression process. Compression ratio values were computed with the intent of maintaining compressed fingerprint image quality consistent with the requirement to preserve the biometric features in the fingerprint images. From the performance analysis results, the overall performance of the proposed compression algorithm achieved an improvement in terms of compression ratio of 20:1 over the existing algorithms which achieved a compression ratio of 15:1 for biometric application. This overall improvement was largely due to the improvement achieved at the transformation stage by employing Coiflet wavelet basis for image energy decorrelation as opposed to Daubechies wavelet in the existing system. In addition, the improvement achieved at the quantization stage through the use of non-uniform quantization as opposed to uniform scalar quantization used in the existing compression algorithm also contributed to the overall performance.

This research work encountered the following challenges:

(i) the non-availability of raw-bits and uncompressed fingerprint image acquisition device. As a result, uncompressed fingerprint dataset were obtained from the database of the National Institute of Standards and Technology (NIST), USA for experimental analysis.

(ii) the presence of background noise in some of the experimental fingerprint images obtained from NIST which accounted for the differences in the overall Peak Signal to Noise Ratio (PSNR) values obtained for fingerprint images at the same compression ratio.

In view of the challenge of image background noise, the denoising of fingerprint image as a pre-processing stage to the lossy compression system is recommended for further research work.

References

[1] Gonzalez, R. C. and Woods, R. E. (2002) *Digital Image Processing*, 2rd Edition, Prentice Hall, Upper Saddle River, New Jersey, pp. 411-486

[2] Xing-Yuan, W., Jiao-Jiao, Y. and Yong-Lei, Z. (2011) An efficient adaptive arithmetic coding image compression technology, Chinese Physical Society and IOP, Vol. 20, No. 10, pp. 104203-1 – 104203-7.

[3] Howard, P. G. and Vitter, J. S. (1994) Arithmetic coding for data compression, Proceedings of the IEEE, Vol. 82, No. 6, pp. 857-865

[4] Sayood, K. (2012) *Introduction to data compression*, Third Edition, Morgan Kaufmann - Elsevier, Massachusetts, USA

[5] Said, A. (2004) *Introduction to Arithmetic Coding - Theory and Practice*; Imaging Systems Laboratory; Academic Press; HP Laboratories Palo Alto;

[6] Bodden, E, Clasen and Kneis, J. (2007) Arithmetic coding revealed, Sable Technical Report No. 2007-5; Retrieved from www.sable.mcgill.ca in July, 2014

[7] Pu, I. M. (2006) *Fundamental Data Compression*; Butterworth-Heinemann, Burlington, MA

[8] Strang, G. and Nguyen, T. (1996) Wavelets and filter banks, Wellesley-Cambridge Press, Massachusetts, USA

[9] Khuwaja, G. A. and Tolba, A.S. (2000) Fingerprint image compression; IEEEexplore; Vol. 2, No1, pp517-526, Retrieved from http://ieeexplore.ieee.org/xpl/login.jsp?tp=&arnumber=890129&url=http%3A%2F%2Fieeexplore.ieee.org%2Fxpls%2Fabs_all.jsp%3Farnumber%3D890129 in December 2014

[10] Winger, L.L. and Venetsanopoulos, A.N. (1998) Biorthogonal modified coiflet filters for image compression; Proceedings of the 1998 IEEE International Conference on, Vol. 5; Retrieved from http://www.researchgate.net/publication/3746977_Biorthogonal_modified_ coiflet_filters_for_image_compression in December, 2014

[11] Li, Y. and Bayoumi, M. (2005) Three-level parallel high speed architecture for EBCOT in JPEG2000, IEEE, pp. 5-8

[12] Khalifa, O. (2005) Wavelet coding design for image data compression, International Journal of Information Technology, Vol. 2, No. 2, pp. 118-128

[13] Sudhakar, R. and Jayaraman, S. (2006) Fingerprint compression using multiwavelets, International Journal of Information and Communication Engineering, Vol. 2, No. 2

[14] Hsin, H. C., Lien, J. J. and Sung, T. Y. (2007) A hybrid SPIHT-EBC image coder, International Journal of Computer Science, Vol. 34, No. 1

[15] Chang, C. C., Chen, S. G. and Chiang, J. C. (2007) Efficient encoder design for JPEG2000 EBCOT context formulation, 15th European Signal Processing Conference, EUSIPCO, Poznan, pp. 644-648

[16] Mushen, Z., Dababneh, M., Al Nsour, A. (2008) Requantization codebook using fingerprint; Science Publications, Journal of Computer Science 4 (11): 959-962

[17] Rawat, C. S. D. and Meher, S. (2009) A hybrid coding scheme combining SPIHT and SOFM based vector quantization for effective image compression, European Journal of Scientific Research, Vol. 38, No. 3, pp. 425-440

[18] Haddad, Z., Beghdadi, A. Serir, A. and Mokraoui, A. (2009) A new fingerprint image compression based on wave atoms transform; IEEE; pp.89-94

[19] Zhao, S. And Wang, X. (2009) Fingerprint image compression based on directional filter banks and TCQ; IEEE Computer Society; Second International Workshop on Knowledge Discovery and Data Mining; pp660-663

[20] Kumar, C., Shekhar, C., Das, S. and Bhandari, S. (2010) Compression algorithm by using wavelets, International Journal of Engineering Science and Technology, Vol. 2, No. 10, pp. 4978-4982

[21] Ashok, J., Shailaja, T. V., Somayajula, S. P. K. (2010) Wave atoms decomposition based fingerprint image compression, IJCSNS, Vol. 10, No. 9, pp. 57-61

[22] Krishnaiah, G. C., Jayachandraprasad, T. and Prasad, M. N. G. (2011) Evolved wavelets for improved fingerprint compression, IJCST, Vol. 2, No. 3, 361-366

[23] Mushen, Z., Dababneh, M., Al Nsour, A. (2011) Wavelet and optimal re-quantization methodology for lossy fingerprint compression, International Arab Journal of Information Technology, Vol. 8, No. 4, pp. 383-387

[24] Gangwar, D. K. (2012) Compression and development analysis on an image using Haar wavelet transform, IJATER, Vol. 2 No. 3, pp225-229

[25] Shanavaz, K.T. and Mythili, P. (2012) Faster techniques to evolve wavelet coefficients for better fingerprint image compression, International Journal of Electronics, Vol. 100 No 5; Accessed from http://www.tandfonline.com/doi/pdf/10.1080/00207217.2012.720944 on April 2013

[26] Shakhakarmi, N. (2012) Quantitative multiscale analysis using different wavelets in 1D voice signal and 2D image, IJCSI International Journal of Computer Science, Vol 9, No. 2, Accessed from http://www.IJCSI.org on April 2013

[27] Libert, J. M., Orandi, S. and Grantham J. D. (2012) Comparison of the WSQ and JPEG 2000 Image Compression Algorithms On 500 ppi Fingerprint Imagery, NIST Interagency Report 7781, Accessed from http://www.nist.gov/customcf/get_pdf.cfm?pub_id=910658 on April 2013

[28] Islam, M. R. et al (2012) Performance analysis of Coiflet-type wavelets for a fingerprint image compression by using wavelet and wavelet packet transform, International Journal of Computer Science & Engineering Survey(IJCSES), Vol. 3, No. 2, pp. 79-87; Retrieved from http://airccse.org/journal/ijcses/papers/ 3212ijcses09.pdf in February 2013

[29] Singla, N. and Sharma, S (2013) A Review on Wavelet based Compression using Medical Images; International Journal of Innovative Research in Computer and Communication Engineering, IJIRCCE; Vol. 1, Issue 8; pp 1649-1653: Retrieved from http://ijircce.com/upload/2013/october/7AReview.pdf in March 2014

[30] Selvakumarasamy, K. et al (2013) Performance Analysis of biorthogonal wavelets for fingerprint image compression, JECE, Vol. 1, No 4, pp2320-8945

[31] Guangqi, S., Yanping, W., Yang, A., Xiao, L. And Tiande, G. (2014) Fingerprint compression based on sparse representation, IEEE Transaction on Image Processing, Vol. 23, No.2, pp.489-501

[32] NIST (2011) WSQ Fingerprint Image Compression Encoder/Decoder Compliance Guidelines, National Institute of Standards and Technology, Accessed from http://nigos.nist.gov:8080/wsq/reference_images_v2.0_pgm.tar on March 2013