# An Efficient Singular Value Decomposition Based Classified Vector Quantization using Discrete Wavelet Transform and its Application to Image Compression

Ali Al-Fayadh[1] and Hadeel Majid[1]

[1] Department of Mathematics and Computer Applications, Al-Nahrain University, College of Science
Bagdad, Iraq
Emails: aalfayadh@yahoo.com , hadela77m@yahoo.com

*Abstract*—**In this paper we present an efficient image compression technique using singular value decomposition (SVD) based classified vector quantization (CVQ) and Discrete Wavelet Transform (DWT) in both the spatial and frequency domains for the efficient representation of still images. The proposed method combines the properties of SVD, CVQ, and DWT; while avoiding some of their limitations. A simple but efficient classifier based gradient method in the spatial domain, which employs only one threshold to determine the class of the input image block into one of finite number of classes, and uses only the first level of the DWT coefficients to determine the orientation of the block without employing any threshold that results in a good image quality was utilized. SVD method was used for efficient generation of the classified codebooks. The proposed technique was benchmarked with the conventional approach based VQ, existing methods using CVQ; and JPEG-2000 image compression techniques. Simulation results indicated that the proposed approach alleviates edge degradation and can reconstruct good visual quality images with higher Peak Signal-to Noise-Ratio (PSNR) than the benchmarked techniques.**

**Keywords-singular value decomposition, DWT, classified vector quantizer, image compression.**

## 1. Introduction

Wavelets are at the forefront of both mathematics and engineering, and essentially provide an alternative to classical Fourier methods for both one and two dimensional data analysis and synthesis. The wavelet transform is very powerful in localizing the global spatial (time) and frequency correlation. Applications of wavelets are quite diverse and include big parts of signal and image processing, data compression, and many other fields of science and engineering.

Images are very important representative objects. They can represent transmitted television or satellite pictures, medical or computer storage pictures and many more. When a two-dimensional light intensity signal is sampled and quantized to create a digital image, a huge amount of data is produced. The size of the digitized picture could be so great that results in impractical storage or transmission requirements. Image compression deals with this problem such that the information required to represent the image is reduced while maintaining an acceptable image quality thus making the transmission or storage requirements of images more practical.

One of the most important applications for the wavelet transform is image compression. Wavelet Image Processing enables computers to store an image in many scales of resolutions. Wavelets at different scales (resolutions) produce different results [1]. Wavelets are of real use in approximating data with sharp discontinuities such as images with lots of edges. An image can be decomposed into approximate, horizontal, vertical and diagonal details.

On the other hand, Singular value decomposition (SVD) is a well-known method in linear algebra. It plays an interesting fundamental role in many different applications such as dimensionality reduction and image compression [2]. The use of SVD in image compression is motivated by its excellent energy compaction property in the least square sense [3]. As a result, the use of SVD technique in image compression has been widely studied [4-5].

Another image compression technique is based on vector quantization (VQ). VQ is a well-known and very efficient approach to low bit-rate image compression [6]. A serious problem in ordinary VQ is edge degradation caused by employing the distortion measure, such as the mean square error (MSE), in searching for the closest codeword in the codebook, as mean square error does not accurately preserve the edge information. To tackle this problem, classified VQ (CVQ) based on a composite source model, has been introduced by Ramamurthi and Gersho [7]. In CVQ model, the image is represented by shade blocks and edge blocks. In CVQ, the classifier separates these two classes and then the blocks belong to a class are coded only with the codeword belong to the same class. An obvious problem with pixel-based classification is high noise-sensitivity because of the high spatial correlation among neighboring pixels. Furthermore, several researches [8-9] have proposed the use of CVQ in the transform domain because it has an excellent compaction-energy.

This paper proposes a combined approach for image compression scheme based on DWT, SVD and CVQ. This approach combines the strengths of the DWT, the SVD and the CVQ, while avoiding some of their limitations. It exploits the high correlation between the pixels inside the image blocks in the spatial domain as well as the energy compaction property of each of the DWT and the SVD techniques, to obtain high quality reconstructed images at low bit-rates. Classification using transform domain, in which a few energy-compacted transform coefficients are used to distinguish edge direction and location, and hence it is computationally simpler than classification in the spatial domain.

The novelty of the proposed method lies in the process of codebook generation using SVD method based CVQ, as well as using only one threshold instead of multiple threshold values as is the case with conventional classified VQ scheme. The remainder of the paper is as follows. In Section 2, a brief description of DWT and SVD concepts and their application to image compression is introduced. The main features of VQ and CVQ are shown in Section 3. Section 4 describes the proposed technique including the methodology to classify the image into different classes. Experimental results are given in Section 5 to demonstrate the potential of the proposed image compression scheme as well as a comparison between the proposed method and conventional approach based VQ, existing methods using CVQ; and JPEG-2000 image compression techniques. Finally, Section 6 presents some conclusions.

## 2. Discrete Wavelet Transform (DWT) and Singular Value Decomposition (SVD):

### *2.1 Discrete Wavelet Transform (DWT)*

Wavelets are mathematical functions that were developed for sorting the data by frequencies. A Wavelet transform converts data from the spatial into the frequency domain. It decomposes a signal (function) into a set of basis functions. These basis functions are called wavelets. The word ``wavelet'' stands for an orthogonal basis of a certain vector space. Wavelet transform involves representing a general function in terms of simple, fixed building blocks at different scales and positions. These building blocks are generated from a single fixed function called mother wavelet $\psi(t)$ by translation and dilation operations [10].

That is,

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right), \qquad a > 0, \qquad b \in \mathbb{R} \tag{2.1}$$

Where $\psi(t)$ has to satisfy

$$\int_{-\infty}^{\infty} \psi(t)dt = 0 \tag{2.2}$$

Which practically implies that $\Psi$ has to oscillate.

The mother wavelet function $\psi(t)$, the scaling variable $a$ and the translation variable $b$ are specifically chosen such that $\psi(t)$ constitutes an orthonormal bases for $L^2(\mathbb{R})$, where $\mathbb{R}$ denotes real numbers [11]. Actually, the mother wavelet $\psi(t)$ is the function with zero translation and a dilation of 1.

With this family we can represent an arbitrary function f(t) as a superposition of different scaled and translated wavelets. Such decomposition decomposes f into different scale levels, where each level is further decomposed with a specific resolution. While analyzing physical situations where the function has discontinuities and sharp spikes, they are efficient and advantageous over traditional Fourier methods.

The discrete wavelet transform (DWT) uses a discrete superposition, i.e., a finite sum instead of an integral, discretized parameters $a = a_0^m$ and $b = ab_0a_0^m$ for $m, n \in \mathbb{Z}$ and fixed $a_0 > 1$ and $b_0 > 1$. With this, we can write the (discrete) wavelet decomposition of a discrete function $f : \{0, \ldots, N-1\} \to \mathbb{N}$ of length N (and grid size h = 1) as

$$f(t) = \sum_{t \in \{0,\ldots,N-1\}} c_{m,n}\left(f(t)\right)\psi_{m,n}(t) \tag{2.3}$$

With the discrete wavelets

$$\psi_{m,n}(t) = \psi^{a_0^m}nb_0a_0^m(t) = a_0^{-m/2}\psi(a_0^{-m}t - nb_0) \tag{2.4}$$

If one chooses $a_0 = 2$ and $b_0 = 2$ , one ends up with an orthonormal basis built by the $\psi_{m,n}$ so that in this case we have

$$c_{m,n}(f(t)) = \langle \psi_{m,n}(t), f(t) \rangle = \int_{-\infty}^{\infty} \psi_{m,n}(t) f(t)\, dt \qquad (2.5)$$

Where $c_{m,n}(f(t))$ describe the projection of $f(t)$ onto the base formed by $\psi_{m,n}(t)$.

Such a basis corresponds to a multiresolution analysis invented by Mallat [12]. Wavelet Image compression decomposing an image into various levels and types of details and approximation with different-valued resolutions. The advantage of decomposing images to approximate and detail parts is that it enables to isolate and manipulate the data with specific properties. After passing the original image into high pass filter and low pass filter, image decomposes into four sub parts. The four sub-bands are: a low frequency sub-band of the approximate image (LL), a high frequency sub-band of the horizontal details of the image (HL), a high frequency sub-band of the vertical details of the image (LH) and a high frequency subband of the diagonal details of the image (HH). This process is called the first level of wavelet decomposition. The low frequency sub-band (LL) can be continually decomposed into four sub-bands. The wavelet decomposition models is shown in Figure 1.

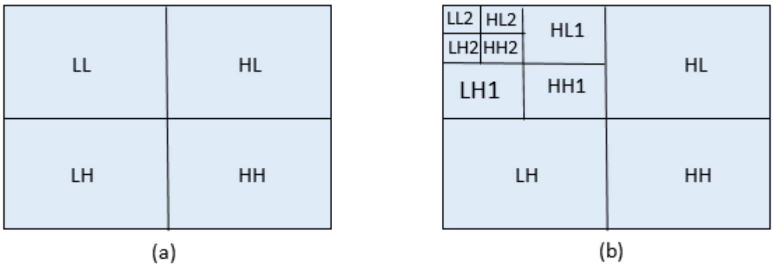


Figure 1. Wavelet decomposition (a) First level (b) Third level

*2.2 Singular Value Decomposition (SVD)*

Singular Value Decomposition is well-known method in linear algebra [2] to diagonalise a rectangular m×n matrix A by factorizing it into three matrices U, S, and V, such that,

$$A = USV^T \qquad (2.6)$$

Where S is a diagonal m×n matrix (the same dimensions as A) with elements $s_i$ along the diagonal and zeros everywhere else. U and V are orthonormal matrices with sizes m×m and n×n, respectively. The matrix U is called the left singular matrix, V is called the right singular matrix, and the diagonal matrix S is the singular values matrix. The singular vectors form orthonormal bases and lead to the following relationship:

$$Av_i = s_i u_i \qquad (2.7)$$

SVD is an approximation technique which effectively reduces any matrix into a smaller invertible and square matrix. Thus, one special feature of SVD is that it can be performed on any real m×n matrix. Equation (2.7) can be expressed as:

$$A = \sum_{i=1}^{p} u_i s_i v_i^T \qquad (2.8)$$

Where $u_i$ and $v_i$ are the i[th] column vectors of U and V respectively, $s_i$ are the singular values, and p = min{m , n}. If the singular values are ordered so that $s_1 \geq s_2 \ldots \geq s_p$, and if the matrix A has a rank r < p, then the last p – r singular values are equal to zero, and the matrix A can be approximated by a matrix A* with rank r ( i.e. the SVD becomes A* ) as the following equation

$$A^* = \sum_{i=1}^{r} u_i s_i v_i^T \qquad (2.9)$$

Hence, the approximation error matrix $E_r$ is dependent on the performance accuracy of the quantisation and/or truncation by parameter r, which can be described as $E_r = A - A^*$. The 2-norm of a matrix may be calculated from the singular values. The 2-norm of approximation error is calculated by

$$E_r^2 = \|A - A^*\|_2 = \left\| \sum_{i=1}^{p} u_i s_i v_i^T - \sum_{i=1}^{r} u_i s_i v_i^T \right\|_2 = \left\| \sum_{i=r+1}^{p} u_i s_i v_i^T \right\|_2$$

$$= \sum_{i=r+1}^{p} (s_i)^2 \qquad (2.10)$$

As the singular values are in descending order, it can be seen that the error decreases towards zero in the 2-norm sense. The property of SVD to provide the closest rank r approximation for a matrix A as shown in equation (2.9) can be used in image processing for compression and noise reduction. By setting the small singular values to zero, matrix approximations whose rank equals the number of remaining singular values can be obtained [13].

## 3. Vector Quantization and Classified Vector Quantization

Quantization is a many-to-one mapping that replaces a set of values with only one representative value. By definition, this scheme is lossy, because after this mapping the original value cannot be recovered exactly [14]. Image quantization is the process of reducing the image data by removing some of the detail information by mapping groups of data points to a single point.

Mathematically, a vector quantizer Q of dimension k and size N is a mapping of a vector in k-dimensional Euclidean space, $R^k$, to a finite subset Y of $R^k$ containing N reproduction points, called codevectors or codewords [6]. Thus,

$$Q: \mathbb{R}^k \rightarrow Y \qquad (3.1)$$

The finite set $Y = \{y_i : i = 1, 2, \ldots, N\}$, where N is the size of the set Y, is called a VQ codebook, and $y_i$ represents the $i^{th}$ codevector (codeword) in the codebook Y.

VQ consists of two parts: an encoder that assigns each input vector $x \in \mathbb{R}^k$ to an index i, and a decoder that finds the codevector by the transmitted index i [6]. VQ builds up a codebook, a dictionary of few codevectors, then each vector is coded with an index value of the closest codevector in the codebook. The best-matched codevector is chosen using a minimum distortion rule. The most widely used distortion measure in image VQ is the squared Euclidean distance between the input vector x and its corresponding codevector y. This measure is given by

$$d(x, y) = \sqrt{\left(\sum_{j=1}^{k} (x_j - y_j)^2\right)} \qquad (3.2)$$

Where $x_j$ and $y_j$ are the jth elements of the vectors x and y, respectively. Figure 2 depicts the basic processing steps of VQ system.

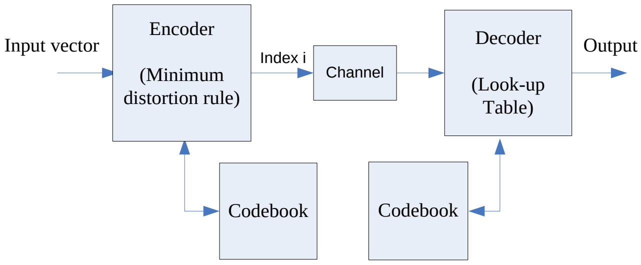


Figure 2. Basic processing steps of VQ system.

Edge is one of the most important features of visual information. Since the human visual system is highly sensitive to edges [15], human eyes are very susceptible to the degradation along edges [16]. This suggests that edges can provide an efficient image representation, making edge-based compression techniques very useful, even at high compression ratios. To improve edge fidelity, classified VQ (CVQ) was proposed [7] .

## 4. THE PROPOSED METHOD

### *4.1 Block Classification*

The proposed method assumes that any edge information within a small image block can be described by a straight line across the block with an abrupt change of intensity in the spatial domain. However, the assumption that any edge segment within a block is a straight line starts to fail for 6×6 blocks and bigger [17]. Therefore a block size of 4×4 was utilized in the proposed technique. The discrete gradient of the block is used as a measure of the edge content of the block in the spatial domain. The orientation of the edge is used to further classify the edge blocks in the frequency domain. Normally, for 4×4 image blocks, the orientations are restricted to four types: horizontal, vertical and two diagonals. Only first level of DWT decomposition is employed to determine the orientation of the edge block.

At the outset, the block mean value is calculated and subtracted from each pixel in the block.

Let $B = \{b_{ij};\ 1 \le i, j \le 4\}$ represents a 4x4 image block. In this case $b_{ij}$ is the gray level pixel value corresponding to position (i, j) of row i and column j in the image block B. The discrete gradients of the block B in the x and in the y directions are determined as follows:

$$G_x = \frac{1}{8}\left[\sum_{i=1}^{2}\sum_{j=1}^{4} b_{ij} - \sum_{i=3}^{4}\sum_{j=1}^{4} b_{ij}\right] \quad and \quad G_y = \frac{1}{8}\left[\sum_{i=1}^{4}\sum_{j=1}^{2} b_{ij} - \sum_{i=1}^{4}\sum_{j=3}^{4} b_{ij}\right] \qquad (4.1)$$

In general, for an even numbers n×m block size, the directional derivatives are:

$$G_x = \frac{2}{m \times n}\left[\sum_{i=1}^{\frac{m}{2}}\sum_{j=1}^{n} b_{ij} - \sum_{i=\frac{m}{2}+1}^{m}\sum_{j=1}^{n} b_{ij}\right] \qquad (4.2a)$$

$$G_y = \frac{2}{m \times n}\left[\sum_{i=1}^{m}\sum_{j=1}^{\frac{n}{2}} b_{ij} - \sum_{i=1}^{m}\sum_{j=\frac{n}{2}+1}^{n} b_{ij}\right] \qquad (4.2b)$$

Where i = 1, 2,…, m; j = 1, 2,…, n. The gradient magnitude within each image block is defined by:

$$|G| = \sqrt{\left(G_x^2 + G_y^2\right)} \qquad (4.3)$$

If the gradient magnitude |G| in equation (4.3) of the block B is smaller than a threshold T, the block contains no significant gradient and it is classified as a shade block; otherwise, it will be classified as an edge block. Once a block is classified as an edge block, the orientation of the edge pattern within the block will be computed using DWT. First level DWT decomposition will be used to determine the edge directions, namely, HL, LH, and HH as follows:

1) Compute each of the edge direction $\beta_\theta$ for $\theta \in \{0, \frac{\pi}{4}, \frac{\pi}{2}, \frac{3\pi}{4}\}$ such that

$$\left.\begin{aligned} \beta_0 &= 2|V| \\ \beta_{\frac{\pi}{4}} &= (4/3)max\{|H+V+D|, |H+V-D|\} \\ \beta_{\frac{\pi}{2}} &= 2|H| \\ \beta_{\frac{3\pi}{4}} &= (4/3)max\{|H-V+D|, |H-V-D|\} \end{aligned}\right\} \qquad (4.4)$$

Where

$$H = \frac{1}{2}A\ ,\ \ V = \frac{1}{2}B,\ \ D = \frac{1}{2}C,\ \ A = HL, B = LH, and\ C = HH.$$

2) The $\theta$ value whose measure is the largest will be selected as the block edge orientation of the block.

Once the block classification process has been completed, five different sub-codebooks are generated, representing the different orientations of edge block information and the shade block. SVD-based CVQ is used for designing the sub-codebooks corresponding to each class. Different rank values have been used in the codebook generating process according to the type of the codebook. Figure 3 shows the classification process.

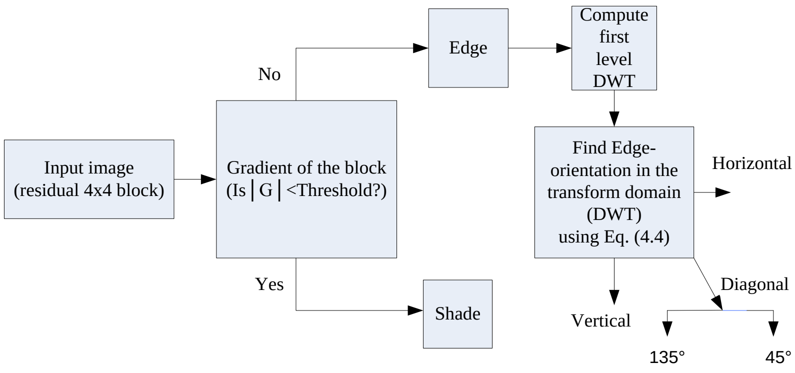


Figure 3: Block diagram of the classification process

*4.2 Construction of the Codebook*

Two standard 512×512 monochromatic images, Barbara and Peppers, are used for codebook constructions. The selected training images are divided into small non-overlapping blocks of size 4×4 pixels making the vector dimension equal to 16. The block mean value is computed and subtracted from each pixel in the block and then the classification process is performed as shown before. The process is resulted in a better utilization of codevectors for encoding shade blocks because they are mapped into small regions near the origin where they can be encoded efficiently. Visually sensitive edge blocks and shade blocks are encoded with a codebook specifically designed for that class of blocks so that distortion is minimized. The value of the threshold T is determined experimentally and set to 15 to obtain a reasonable percentage of the shade blocks. Once the block classification process has been completed, different sub-codebooks corresponding to the different classes are generated, using SVD-based CVQ technique with different rank values. As VQ scheme requires operations in multidimensional space, the utility of the SVD for dimensionality reduction purpose may alleviate the complexity of a scheme based VQ. The different five sub-codebooks corresponding to the classified classes were generated by projecting the n dimensional dataset of a class to the space spanned by the m ($m < n$) most significant singular vectors of the m ($m < n$) largest singular values. That is, the sub-codebooks were generated from the eigenvectors of a set of image in a corresponding class.

*4.3 Encoder*

The encoder of the proposed method operates on the residual blocks where the mean value of the input image block is subtracted from each pixel in the block to yield a residual block (vector). The mean values are then encoded separately. This is because the mean values are usually highly correlated with adjacent mean values. The mean values are encoded with 6-bits value for a 256 level gray scale image using prediction method together with an improved greyscale (IGS) quantization method [18].

The next step is to encode the image by decomposing the image blocks into shade and edge blocks using Equation (4.3). If a block is classified as an edge block, the orientation of the edge pattern within the block will be computed using Equation (4.4). Then it encodes the two types of blocks (the shade block and the edge-orientation block) separately using CVQ scheme. The input image is divided into small non-overlapping blocks of size 4×4 pixels which are then processed independently. Each input image block is compared with the closest codeword in the codebook of the same type using the MSE as the distortion measure. The codeword with the lowest MSE is selected as the block compressor. This technique reduces the computational complexity by comparing the input image block with only those of the same type to obtain the closest codeword. Figure 4 shows the block diagram of the proposed method encoder.

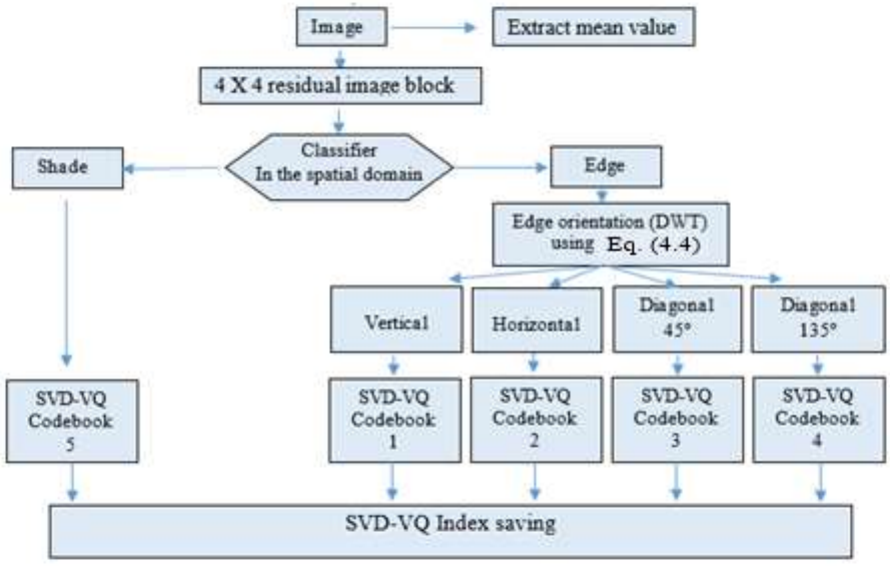


Figure 4. Block diagram of the proposed method encoder

As one of the classified codebook design problems is concerned with deciding the set of sizes of the codebook classes to minimize the overall distortion, approximately the same number of codevectors for all the edge classes has been selected. This assumption is motivated by the fact that all the edge classes are perceptually have equal importance. This implies that an edge block will appear equally distorted to the eye irrespective of its class. The shade class, however, can be given a less number of codevectors in comparison to the edge classes as it is an easily coded class.

### *4.4 Decoder*

The decoder performs simple table look-up operations to retrieve the corresponding codeword from the same codebook as the encoder used, and computes the inverse of the used transforms. As the residual block is used in the encoding process, the block mean value is added to the reconstructed image block.

## 5. SIMULATION RESULTS:

A database of five grey level images was developed to systematically evaluate the application of the proposed system. MATLAB code was written for the generation of the proposed method performed on Pentium (R) Core i7-4700MQ CPU@2.40GHz Windows 8.1 pro 64-bit. The training set used is obtained from two 512×512 monochromatic images of 8-bit intensity, Barbara512 and Peppers512. Three test images outside the training set: Baboon512, Lena512 and Goldhill256 as well as the two images inside the training set: Barbara512 and Peppers512 were used to evaluate the performance of the proposed method. These test images were coded by the proposed method employing the same codebooks that were used for coding the images Barbara512 and Peppers512. The performance of this scheme is usually characterized using different image quality metrics, the MSE and the Peak Signal- to Noise- Ratio (PSNR) as image quality metrics based error, as well as the Structural Similarity Index (SSIM) based approach proposed by Wang et al. [19].

The PSNR is defined as follows:

$$PSNR = 10log_{10}\left[\frac{(255)^2}{MSE}\right] \qquad (5.1)$$

Where MSE is the mean square of the error between the original and the reconstructed images, and it is defined as:

$$MSE = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(x_{ij} - y_{ij})^2 \qquad (5.2)$$

Where MN is the total number of pixels in the image.

The SSIM based approach used first and second order statistics of the original and distorted images and has been defined as follows [19].

$$SSIM(X,Y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \quad (5.3)$$

Where

$$C_1 = (K_1L)^2 \quad and \quad C_2 = (K_2L)^2$$

Where L is the dynamic range of the pixel values (255 for 8-bit images). The constants $C_1$ and $C_2$ are small positive constants where $K_1$ and $K_2$ are the same as in [19]: $K_1 = 0.01$ and $K_2 = 0.03$. The quantities $\mu_x$, $\sigma^2_x$ be the mean and variance of the reference image $X = \{x_i | i = 1,2,\ldots,N\}$, and $\mu_y$, $\sigma^2_y$ be the mean and variance of the distorted image $Y = \{y_i | i = 1,2,\ldots,N\}$ while $\sigma_{xy}$ be the covariance of the reference and distorted images and are given by

$$\mu_x = \frac{1}{N}\sum_{i=1}^{N} x_i, \quad \mu_y = \frac{1}{N}\sum_{i=1}^{N} y_i, \quad \sigma_x^2 = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)^2 \quad (5.4a)$$

$$\sigma_x^2 = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)^2, \quad \sigma_{xy} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y) \quad (5.4b)$$

The overall image quality is measured by the Mean SSIM (MSSIM) index which is given by

$$MSSIM(X,Y) = \frac{1}{M}\sum_{j=1}^{M} SSIM(x_j, y_j) \quad (5.5)$$

where $x_j$ and $y_j$ are the image contents at the j-th local window; and M is the number of local windows in the image.

Tables 1 and 2 show the performance of the proposed method characterised by the PSNR and the MSSIM based approaches respectively, when the codebook is constructed using SVD-based CVQ trained using the Barbara512 and the Peppers512 images, while Figure 5 show some of the reconstructed compressed images of the proposed method and their corresponding reconstructed error.

Table 1: Reconstruction performance for the proposed method trained on the top two images (Barbara512 and Peppers512) and generated to the rest of the images. This test is calculated by using approximately the same bitrate.

| | **Proposed Method** | | **VQ** | | **JPEG-2000** | |
|---|---|---|---|---|---|---|
| **Image** | **Bitrate (bpp)** | **PSNR (dB)** | **Bitrate (bpp)** | **PSNR (dB)** | **Bitrate (bpp)** | **PSNR (dB)** |
| Barbara512 | 0.5639 | 33.2273 | 0.5604 | 27.3250 | 0.5654 | 33.1516 |
| Peppers512 | 0.5434 | 35.8529 | 0.5446 | 28.5478 | 0.5333 | 35.8340 |
| Lena512 | 0.5270 | 36.1468 | 0.5282 | 28.7617 | 0.5114 | 36.7676 |
| Baboon512 | 0.6378 | 30.8584 | 0.6229 | 25.5914 | 0.6534 | 30.2985 |
| Goldhill256 | 0.7332 | 32.7935 | 0.7124 | 26.7216 | 0.7554 | 31.4860 |

Table 2: Reconstruction performance for the proposed method trained on the top two images (Barbara512 and Peppers512) and generated to the rest of the images. This test is calculated by using approximately the same bitrate.

| | **The Proposed Method** | | **VQ** | | **JPEG-2000** | |
|---|---|---|---|---|---|---|
| **Image** | **Bitrate (bpp)** | **MSSIM** | **Bitrate (bpp)** | **MSSIM** | **Bitrate (bpp)** | **MSSIM** |
| Barbara512 | 0.5639 | 0.8664 | 0.5604 | 0.1973 | 0.5654 | 0.9100 |
| Peppers512 | 0.5434 | 0.9229 | 0.5446 | 0.2761 | 0.5333 | 0.9208 |
| Lena512 | 0.5270 | 0.9300 | 0.5282 | 0.4742 | 0.5114 | 0.9427 |
| Baboon512 | 0.6378 | 0.7907 | 0.6229 | 0.1223 | 0.6534 | 0.7836 |
| Goldhill256 | 0.7332 | 0.8510 | 0.7124 | 0.1429 | 0.7554 | 0.8207 |

The results in Tables 1 and 2 show that, in all cases, the proposed method outperformed ordinary VQ and show competitive results in comparison to JPEG-2000 standard which was generated using MATLAB [18] in terms of both the PSNR and the MSSIM. However, the proposed model outperforms JPEG-2000 in some cases where the test images were of high detail type (the images Baboon512 and Goldhill256). This is because of the good approximation of the edge blocks which lie far away from the densely region near the origin. On the other hand, popular transform-based lossy compression techniques tend to introduce artifacts at high frequency signal components since such details often represent high frequency components in frequency domain.

The simulation results also indicated that the proposed technique needs shorter encoding CPU time (in seconds) than the ordinary VQ while the standard JPEG-2000 needs shorter encoding CPU time than the proposed method as indicated in Table 3.

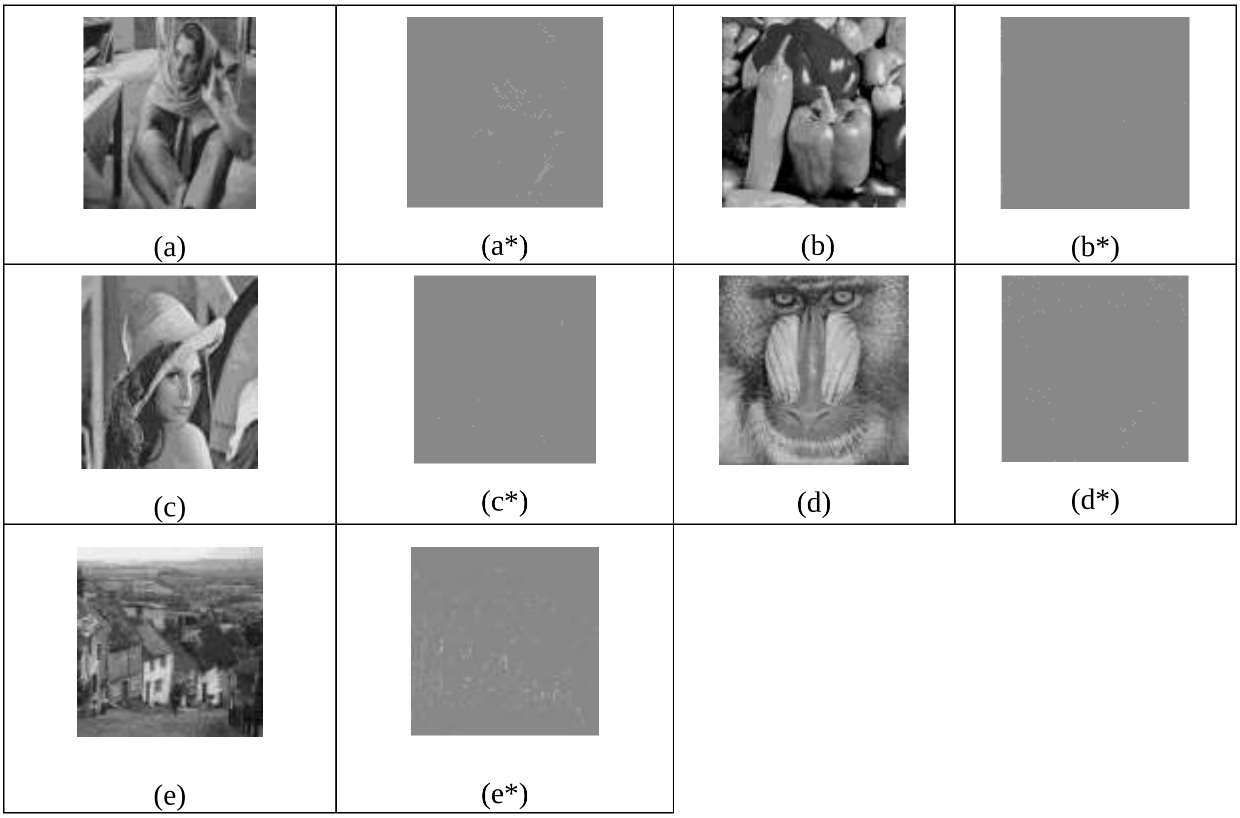


Figure 5. Some of the reconstructed compressed images by the proposed method.
(a) 512 × 512 reconstructed image Barbara512 at bitrate 0.5639bpp and psnr = 33.2273dB, b) 512 × 512 reconstructed image Peppers512 at bitrate 0.5434bpp and psnr = 35.8529dB, (c) 512 × 512 reconstructed image Lena512 at bitrate 0.5270bpp and psnr=36.1468dB, (d) 512 × 512 reconstructed image Baboon512 at bitrate 0.6378bpp and psnr = 30.8584dB, (e) 256 × 256 reconstructed image Goldhill256 at bitrate 0.7332bpp and psnr = 32.7935dB. The images a*, b*, c*, d*, e* are the differences between the original images and their reconstructed images a, b, c, d and e plus 128, respectively.

Table 3: Comparison of computing time between the proposed method and each of the ordinary VQ and JPEG-2000 technique at fixed bitrate.

| **Image** | **Bitrate (bpp)** | **Ordinary VQ using k-means Time (sec.)** | **proposed method Time (sec.)** | **JPEG-2000 Time (sec.)** |
|---|---|---|---|---|
| Barbara 512 | 0.5639 | 103.31 | 16.9573 | 5.32 |
| Peppers 12 | 0.5434 | 104.65 | 9.5317 | 5.17 |
| Lena 512 | 0.5270 | 103.08 | 11.5597 | 5.66 |
| Baboon 512 | 0.6378 | 104.05 | 24.5234 | 5.42 |
| Goldhill 256 | 0.7332 | 26.39 | 6.3960 | 1.78 |

By comparing the proposed scheme with more conventional CVQ methods [7], [20], [8], [21], and [22]. It has been noted that the proposed method maintains higher PSNR values for the same images at the same bitrate. For example, for the image Lena512, the comparisons are summarized in Table 4.

Table 4: Comparison results between the proposed method and more conventional CVQ for the image Lena512. This comparison is calculated by using approximately the same bitrate.

| | **PSNR (dB) values for the image Lena512 by different methods. This comparison is calculated by using approximately the same bitrate.** | | | | | |
|---|---|---|---|---|---|---|
| **Bitrate (bpp)** | **Reference [7]** | **Reference [20]** | **Reference [8]** | **Reference [21]** | **Reference [22]** | **Proposed Method** |
| 0.625 | - | 31.26 | - | 32.65 | 36.7972 | **37.0248** |
| 0.688 | - | 31.79 | - | 33.27 | 37.1509 | **37.3145** |
| 0.70 | 29.79 | - | - | - | 37.1885 | **37.3581** |
| 0.750 | - | 32.23 | - | 33.80 | 37.4166 | **37.4639** |
| 0.530 | - | - | 34.14 | - | 36.0838 | **36.3964** |
| 0.572 | - | - | 34.49 | - | 36.4723 | **36.7124** |
| 0.600 | - | - | 34.74 | - | 36.6557 | **36.7943** |

## 6. CONCLUSIONS

An efficient coding method for accurate reconstruction of still images at low bit-rates was presented. The method combines SVD based CVQ as well as the DWT in both the spatial and transform domains. The classification algorithm uses only one threshold value on the magnitude of the gradient across the image block. An edge-oriented classifier using only first level decomposition of DWT transform is utilized to determine the direction of the edge block. The method also shows an advantage in PSNR and MSSIM over the standard VQ method using the k-means algorithm and the existing methods using CVQ scheme; and competitive to the JPEG-2000, for similar values of the bit-rate

## 7.References

[1] D. Salomon, "Data Compression: The Complete Reference, 4th Ed.", Springer-Verlag London Limited 2007.

[2] S. Gilbert, "Introduction to Linear Algebra", Wellesley-Cambridge Press, (1998).

[3] M. Tian, Si-Wei Luo, and Ling-Zhi Liao, "An Investigation into Using Singular Value Decomposition As A Method of Image Compression", Machine Learning and Cybernetics, 2005. Proceedings of 2005 International Conference, Vol. 8, 5200-5204, 18-21 Aug.2005.

[4] A. Dapena and S. Ahalt, "A Hybrid DCT-SVD Image-Coding Algorithm", IEEE Trans. Circuits Syst. Video Technol., Vol. 12, No. 2, 114-121, Feb. 2002.

[5] S.O. Aase, J. H. Husoy and P. Waldemar, "A Critique of SVD-Based Image Coding Systems", IEEE International Symposium on Circuits and Systems VLSI, Vol. 4,13-16, Orlando FL, May 1999.

[6] A. Gersho and R. M. Gray, "Vector Quantization and Signal Compression" KluwerAcademic Publishers, 1992.

[7] B. Ramamurthi and A. Gersho "Classified Vector Quantization of Images", IEEE Trans. Commun. COM-34, 1105-1115, 1986.

[8] J. W. Kim and S. U. Lee, "A Transform Domain Classified Vector Quantizer for Image Coding", IEEE Trans. Circuits and Systems for Video Technology, Vol. 2, No. 1, 3-14, 1992.

[9] B. Shen and I. K. Sethi: Direct Feature Extraction from Compressed Images. In Proc. SPIE: Storage & Retrieval for Still Image and Video Databases IV, Vol. 2670, 404-414, Mar. 1996.

[10] V. Alarcon-Aquino and J. A. Barria, "Change detection in time series using the maximal overlap discrete wavelet transform", Latin Am. Appl. Res. An. Int. J., vol. 39, no. 2, 145-152, April 2009.

[11] I. Daubechies, "Ten Lectures on Wavelets", New York: SIAM, Philadelphia, 1992.

[12] S. Mallat. "A theory of multiresolution signal decomposition: the wavelet representation", IEEE Transactions on Pattern Analysis and Machine Intelligence, 11:674–693,1989.

[13] D. P. Berrar, W. Dubitzky, M. Granzow, eds., "Singular Value Decomposition and Principal Component Analysis", in a Practical Approach to Microarray Data Analysis, Kluwer: Norwell, MA, 91-109. LANL LA – UR – 02 – 4001, 2003.

[14] P.C Cosman, R.M Gray and M.Vetterli, "Vector Quantization of Image Sub bands: A Survey," IEEE Transactions on image processing, September 14th, 1995. http://citeseer.ist.psu.edu/cosman96vector.html.

[15] T. Cornsweet, "Visual perception", Academic Press, New York, 1970.

[16] D. H. Hubel and T. Weisel, "Brain Mechanisms of Vision", Sci. Amer., 130-146, Sep. 1979.

[17] N. Keskes, F. Kretz, and H. Maitre, "Statistical Study of Edges in TV Pictures", IEEE Trans., COM-27, 1239-1247, 1979.

[18] R. C. Gonzalez, R. E. Woods, and S. L. Eddins, "Digital Image Processing UsingMATLAB", 2004.

[19] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image Quality Assessment: From Error Visibility to Structural Similarity", IEEE Transactions on Image Processing, Vol. 13, No. 4, 1-14, 2004.

[20] D. S. Kim and S. U. Lee: Image Vector Quantizer Based on A Classification in The DCT Domain. IEEE Trans. Commun. COM-39, 549-556, 1991.

[21] M. K. Quweider and J. B. Farison, "Classification of Image Edge Blocks by Orthogonal Projection", Electron. Lett, Vol. 33, No. 22, 1856-1857, 23rd Oct. 1997.

[22] Ali Al-Fayadh, Abir Jaafar Hussain, Paulo Lisboa, and Dhiya Al-Jumeily, " Novel Hybrid Classified Vector Quantisation using Discrete Cosine Transform for Image compression", J. Electron. Imaging. 18(2), 023003 (April 21, 2009). Doi:10.1117/1.3116564.