Abstract— Image coding requires a small bit rate for high-speed data transmission and a small space for data storage. Simultaneously, the peak signal to noise ratio (PSNR) has to be maintained. In this paper, we proposed a method of image coding design using wavelet transform (WT). By applying the WT for defining groups of pixels with the same intensity in spatial domain, the groups of pixels are allocated in a low frequency range. Hence, locations of pixels are the key factor to determine the size of each block and we use wavelet transform to decompose each block into subband components, which are represented by 3D vectors. The 3D vectors are then classified into 8 groups corresponding to quadrants of spatial coordinates. In addition, we apply Fuzzy C-Means algorithm to classify the member in the magnitudes value of 3D vectors into code vector. Due to the lossy coding process, we propose a method of system error compensation on Vector Quantization (VQ) by using principle component analysis and discrete wavelet transform to performed on the system error and keeping the high-energy coefficient for further inverse wavelet transform to yield system error compensation. The reconstructed image and system error compensate will be combined in order to construct an output image \(X_o\). By applying the proposed method, performance of the method is evaluated as 26.19% of bit rate and 1.50% of PSNR improved.

I. INTRODUCTION

Data compression is the mapping of data set into a bit stream to decrease the number of bits required to represent the data set. The data storage space is saved, and memory access time is fastened in case of applying an efficient image coding. In order to enhance the efficiency of communication system for real-time image transmission, we need to decrease the bit rate, and simultaneously maintain PSNR for image quality. To satisfy the mentioned conditions, many developments and researches are undertaken, and results have been published.

M. Antonini, M. Barlaud, P. Mathieu, and I. Daubechies in [1] propose wavelet transform as a tool to reduce memory sizes, and increase the quality of image by using multi-resolution codebook. They design codebook by using a well known splitting technique. The initial codebook is generated by splitting the centroid (center of gravity) of the training set. This splitting technique classifies data image into code vectors and determines mean squared error (MSE). In the splitting technique, median of group data is analyzed without considering the interpixel information. Hence, the adjacent block with high frequency data is not considered in the process. Furthermore, the process with 1 bpp compensates PSNR by using two different sizes of blocks, 2x2 \((N = 256)\) and 4x4 \((N = 128)\), which results in higher PSNR. Chaur-Heh Hsieh group in [2] applies several fast search algorithms based on TIE with multiple control points and the Haar wavelet transform. TIE is constructed with multiple control vectors. A systematic way for determining the control vector is developed. A codebook with 256 code vectors is generated by the fast fuzzy C-mean algorithm with block size of 4x4. L. Chin-Yuan and C. Chin-Hsing in [3], apply wavelet transform to reduce bit rate using in data transmission and data storage. The image quality is improved by using multi resolution codebook designed by the genetic algorithm and the grey-base designed competitive learning network. Both of these techniques provide a better quality of image. However, designing the codebook with this technique considers only the center of each group which calculated from individual pixel in the same group. The codebook is composed of two different block sizes (e.g., 2x2 with 512 code vectors and 4x4 with 64 code vectors). The results shown that the bit rate is 0.84 bpp.

In this paper, the authors aim to find VQ-based coding method with low bit rate and high PSNR. We proposed a approach technique to that utilizes frequency components to determine size of blocks. One of the main advantages of the proposed method is that the size of each block is adjustable to minimize coding errors.

This paper is organized as follows: Section 2 describes problem statement in an image coding. Sections 3 and 4 are a coding system and system error compensate. Section 5 presents experiments and results. Finally, conclusion is in sections 6.

II. PROBLEM STATEMENT IN AN IMAGE CODING

One of research issues in image coding is to find coding methods with low bit rate and high PSNR; the conventional
method employs average of intensity in the 2x2 blocks for
coding. Normally, if we enlarge a block for coding, bit rate is
decreased. On the other hand, since coded data is generated
from intensity average in the block, PSNR becomes worse. In
this paper, we raise “To decrease bit rate while maintaining
PSNR” as theme, and declare it as problem statement. The
small-block coding will absorb distortions between intensity
values and their averages. That means PSNR is decreased
according to block size. However, block size should be
enlarged to cover area consisting of similar intensity values
in order to decrease bit rate maintaining PSNR. Therefore, we
should determine block location and size depending on
adjacent pixels that have similar intensity values.

In frequency domain as shown in Fig. 1, spectrum
representing block that contains similar-intensity pixels is
located in low frequency range. On the other hand, a block
with different - intensity pixels are located in a high frequency
range. Obviously, we can use frequency components to
determine the block location and size in spatial domain.
Separating the low frequency components and high frequency
components can be done in many ways.

One way is the decomposition of the image using a Discrete
Wavelet Transform (DWT) [4] which are used in several
applications.

III. CODING SYSTEM

In this paper image data coding consists of encoding stage and
decoding stage demonstrated in Fig. 2.

A. Encode/Decode

The aim of encoding process is to produce the image with
high PSNR. The process consists of two steps. The first step,
the reconstructed image \( X_i' \) is reconstructed image from
the relationship between the codebook and the nearest
neighbor rule. In the second step, the image is compressed in
a lossy style. To reduce the system error \( e_i \), the principle
component analysis (PCA) [5]-[6] and wavelet transform
(WT) to the system error in frequency domain. The system
error and system error compensate \( e_2 \) can be calculated, is
defined as (1) and (5).

\[
e_1 = X_i - X_i'
\]
\[
e_{11} = \text{PCA Encode}(e_i)
\]
\[
e_{33} = DWT(e_{11})
\]
\[
e_{22} = IDWT(e_{33})
\]
\[
e_2 = \text{PCA Decode}(e_{22})
\]
\[
X_O = X_i' + e_2; \quad e_2 \neq e_i
\]

Where PCA Encode is principle component analysis
encode, IDWT is inverse wavelet transform, PCA Decode is
principle component analysis decode, \( X_i \) is input image, \( X_i' \) is
reconstructed image and \( X_O \) is output image. Therefore, if
system error compensation approaches system error as close
as possible the output image will approach input image, high
PSNR, is define as (6).

In order to have the image reconstructed, the decoding
process needs to have the reverse process of the encoding.
Furthermore, the system error will be inverse transformed.
Then the construction of output image \( (X_O) \) is generated from
the merging of reconstruction image and system error. The
closed loop control algorithm (CLC) that is demonstrated in
Fig. 2.

B. Codebook Design

In the past decade, wavelet transform is a popular research
topic applying for signal processing such as image and sound
signals. It is an effective display technique for analyzing time
and frequency [7]. Moreover, this technique is widely
accepted. In the paper, wavelet transform is applied to
determine a block location and size which depend on adjacent
pixels that have similar intensity values. A group of vector
code from a wavelets’s theory is also implemented to
experiment assumptions. And the mother function of wavelets,
\( \Psi(x) \) can be any function if it satisfies the following condition
\[
\int_{-\infty}^{\infty} |\Psi(x)|^2 \, dx < \infty \tag{7}
\]

Basically, a mother wavelet function is derived from its corresponding mother scaling function \( \Phi(x) \) which satisfies the scaling equation
\[
\Phi(x) = \sum_{i \in \mathbb{Z}} c_i \Phi(2x - i) \tag{8}
\]
For any integer \( j \), the vector space \( V^j \) is defined as
\[
V^j = \text{span}\{\Phi^j(x), \forall i \in \mathbb{Z}\} \tag{9}
\]
\[
\Phi^j(x) = \Phi(2^j x - i), i, j \in \mathbb{Z} \tag{10}
\]

If \( V^j \) satisfies the following four multi-resolution analysis conditions

Condition 1: \( f(x) \in V^j \iff f(2^{-j} x) \in V^0, \forall j \in \mathbb{Z} \)
Condition 2: \( \bigcap_{i \in \mathbb{Z}} V^i = \{0\} \)
Condition 3: \( \bigcup_{i \in \mathbb{Z}} V^i = L^2(\mathbb{R}) \)
Condition 4: \( \ldots V^{-2} \subset V^{-1} \subset V^0 \subset V^1 \subset V^2 \subset \ldots \)

That the mother wavelet is given by
\[
\Psi(x) = \sum_{i \in \mathbb{Z}} (-1)^i c_{-i} \Phi(2x - i) \tag{11}
\]

One of the most important characteristics of the wavelet transform is multi-resolution. The wavelet transform converts the pixel values of an image to its wavelet representation, without losing any information, in the spatial domain. In this paper, the Interpixel codebook is designed by applying the wavelet transform theory. The Interpixel codebook design consists of the following steps:

Step 1: The input image size 512x512 pixels are separated into 16,384 square sub-blocks. Each sub-block has 4x4 pixels and nominated as \( E_i \) to \( E_{16,384} \) as shown in Fig. 3.

Step 2: Analyze the characteristics of each sub-block to define address of sub-block associated with frequency of input image.
- Each sub-block is decomposed into 2 levels. The two levels using the Haar basis wavelet transform is performed on image to generate the corresponding transformed coefficients. As shown in Fig. 3.
- 3D vector \( \{ \hat{B}_1, \hat{B}_2, \hat{B}_{16,384} \} \) is constructed by using the coefficient values of LH2, HL2, and HH2 sub-bands to determine the unit vector \( i, j, k \).
- Find magnitudes \( B_1, B_2, B_{16,384} \) of 3D vectors and phase \( (\theta_n) \) is discarded, which can be described are:
  \[
  B_1 = \| \hat{B}_1 \| \angle \theta_1, \quad B_2 = \| \hat{B}_2 \| \angle \theta_2, \\
  B_1 = \| \hat{B}_1 \| \angle \theta_1, \ldots, B_{16,384} = \| \hat{B}_{16,384} \| \angle \theta_{16,384}
  \]

Step 3: Divide sub-blocks into eight groups (Total group) according to 3D vector signs as show in Fig. 4 and Small group less than 256 blocks, as shown in (11)-(12)
\[
\text{Total group} = \begin{cases} 
  \{ \text{Big group}(m) \} | i, j, k | m = 1, \\
  \{ \text{Big group}(m) | i, j, k | m = 2, \\
  \ldots, \text{Big group}(m) | i, j, k | m = 8 \\
  \{ \text{Small group}(1), \text{Small group}(2), \ldots, \text{Small group}(n) \}
\end{cases}
\tag{13}
\]

Step 4: Find amount code vector by applying the Fuzzy C-Means algorithm [8]. Fuzzy C-Means algorithm classifies \( d \) dimensional data into groups, says \( k \) groups. Unlike the classical \( K \)-means scheme where the data is solely classified into one of the \( k \) groups, Fuzzy C-Means algorithm computes the membership probabilities and classifies the data based upon them. The membership probability to classified \( i^{th} \) data into \( j^{th} \) group, \( u_{ij} \), is derived by minimizing the object function, \( J_m \), which is defined as
\[
J_m = \sum_{i=1}^{N} \sum_{j=1}^{K} u_{ij}^m \| x_i - c_j \|^2, \quad 1 \leq m \leq \infty \tag{14}
\]
where \( m \) is the arbitrary real number, \( x_i \) is the \( d \) dimensional \( i^{th} \) data, \( c_j \) is the group centroid and \( \| \cdot \| \) is the similarity measure among groups. The membership probability, \( u_{ij} \), and the group centroid are defined as
\[
u_{ij} = \frac{1}{\sum_{k=1}^{N} u_{ij}^m \| x_i - c_k \|^{m-1}} \cdot \frac{1}{N} \sum_{i=1}^{N} u_{ij}^m x_i \tag{15}
\]
The Fuzzy C-Means algorithm is proceeded as followed:

(i) Initialize the matrix \( U^{(0)} = \{ u_{ij} \} \)
(ii) For \( k^{th} \) loop, compute the group centroid \( C^{(k)} = \{ c_j \} \) using \( U^{(k)} \) where
\[
C^{(k)}_{mj} = \frac{1}{N} \sum_{i=1}^{N} u_{ij}^m x_i
\]
(iii) Update \( U^{(k)} \) to \( U^{(k+1)} \) where
\[
u_{ij} = \frac{1}{\sum_{k=1}^{N} \| x_i - c_k \|^{m-1}} \cdot \frac{1}{N} \sum_{i=1}^{N} u_{ij}^m x_i \tag{16}
\]
(iv) Stop if \( \| U^{(k+1)} - U^{(k)} \| < \varepsilon \) else go to step 2

450
In this paper, we apply Fuzzy C-Means algorithm to further classify the member in the Big group into $K$ Small group. The 2 dimensional data is formed by using the magnitude and the phase of the vector. In each Small group, means of magnitude values is then calculated to determine address of sub-blocks.

**Step 5:** Find Interpixel codebook from address of sub-blocks size 4x4 pixels, as shown in Table. I and Fig. 5 (Address of sub-blocks is defined as address of $E_1$ to $E_{16,384}$).

**TABLE I**

A CLASSIFICATION SAMPLE OF 3D VECTOR TO 8 TYPES

<table>
<thead>
<tr>
<th>Quadrant ($Q_s$)</th>
<th>3D Vector ($B_n$)</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s = 1, 2, ..., 8$</td>
<td>$n = 1, 2, ..., 8$</td>
</tr>
<tr>
<td>$Q_1$</td>
<td>$-i, -j, -k$</td>
</tr>
<tr>
<td>$Q_2$</td>
<td>$-i, -j, +k$</td>
</tr>
<tr>
<td>$Q_3$</td>
<td>$-i, +j, -k$</td>
</tr>
<tr>
<td>$Q_4$</td>
<td>$-i, +j, +k$</td>
</tr>
<tr>
<td>$Q_5$</td>
<td>$+i, -j, -k$</td>
</tr>
<tr>
<td>$Q_6$</td>
<td>$+i, -j, +k$</td>
</tr>
<tr>
<td>$Q_7$</td>
<td>$+i, +j, -k$</td>
</tr>
<tr>
<td>$Q_8$</td>
<td>$+i, +j, +k$</td>
</tr>
</tbody>
</table>

where Small group is integer value in the range of \([1, 2, ..., 256]\). From Fig. 5 find Interpixel codebook by applying the following equation.

$$C_{\text{Small group}(n)}(i, j) = \frac{1}{m} \sum_{i=1}^{m} (X_{i,j}); j = 1, 2, ..., 16 \quad (16)$$

where $m$ is member in each column of Small group(n).

256. Fig. 5 demonstrates the designing of Interpixel codebook by using numbers of Small group as shown in Fig. 5. Each Small group consists of magnitude values which can determine positions of Interpixel codebook related to frequency as shown in Fig. 1.

**IV. SYSTEM ERROR COMPENSATION (SEC)**

Compression in the VQ pattern basically causes information loss. The information loss then causes reconstruction containing loss information. Moreover, system error non-compensation will cause blocky effect to occur. Therefore,

**Fig. 3. A sample of wavelet transform input image to find 3D vectors.**

Therefore, if a test uses numbers of bit equal to 8, the result will yield numbers of code vector that is equal to

**Fig. 4. A grouping sample of magnitude value.**

**Fig. 5. Interpixel codebook design.**
the VQ compression needs improvement to contain system error compensation. The algorithm system error compensation consists of two main processes. In the first process, algorithm system error \( E \) is calculated from input image \( \left( X_{ij} \right) \) of size 512x512 pixels and the reconstruction image \( \left( \hat{X}_{ij} \right) \). In the second part, system error is encoded of using principle component analysis (PCA). PCA is widely used in data processing and dimensionality reduction. PCA is a useful statistical technique that has found application in fields such as face recognition and image compression, and is a common technique for finding patterns in data of high dimension. In this paper, we applied PCA to system error is reduced dimension (4x16,384 pixels) of system error, we called system error1 \( e_{i,j} \). System error1 is decomposed into 2 sub-bands \([LL_1 (1x8,192), HH_1 (1x8,192)]\). Perform single-level discrete wavelet transform (DWT) of system error by wavelet transform name “haar”. However, only sub-band \( LL_1 \) is used.

\[ \text{Step 4:} \quad \text{Choosing components and forming a feature vector} \]

\[ \text{Step 5:} \quad \text{Deriving the new data set} \]

Upon decoding, the reconstructed image and system error is then combined to construct an output image. The resulting images were evaluated by the peak signal to noise ratio (PSNR) and the mean squared error (MSE) is defined (for images of size N x N) as

\[ \text{PSNR} = 10 \log_{10} \frac{255^2}{\text{MSE}} \]

\[ \text{MSE} = \frac{1}{N^2} \sum_{i=1}^{N} \sum_{j=1}^{N} (X_{ij} - \hat{X}_{ij})^2 \]

where \( X_{ij} \) and \( \hat{X}_{ij} \) are the pixel gray levels from the original and reconstructed image respectively as shown in Table II - Table VI, and 255 is the peak gray level. The comparison between CLC algorithm and the LBG is shown in Table II. The LBG, most popular algorithm, is used in VQ [1]. The bit rate of \([(512*512*8)/(4*4))] = 0.50 bpp for CLC, \([(512*512*8)/(4*4)]+(8*(1024+512+4))] / [(512*512)+(1024+512+4)]=0.54 bpp for CLC+SEC respectively. The comparison between CLC and CLC+SEC systems is shown in Table III - Table IV and V which shows the compression ratio (CR) of \([512*512*8] / (512*512*8/(4*4))] = 16.00 for CLC, and \((512*512*8) / (8*(1024+512+4))] = 14.63 for CLC+SEC respectively. And Table V the average PSNR produced by the proposed CLC+SEC (0.80 bpp) is 16.50% higher than that by the conventional method (WT+GGCLN, 0.84 bpp)[3].

V. EXPERIMENTS AND RESULTS

The experiment condition is setup Pentium(R) 4 CPU 3.01 GHz and block size 4x4 pixels. The four standard images of size 512x512 are Airplane, Girl, Lena, and Pepper images. The tests of CLC and CLC+SEC algorithm using the four standard images (block size 4x4 pixels) are demonstrated in Fig. 6.

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Images</th>
<th>0.38 bpp PSNR</th>
<th>MSE</th>
<th>0.44 bpp PSNR</th>
<th>MSE</th>
<th>0.50 bpp PSNR</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLC</td>
<td>Airplane</td>
<td>26.82</td>
<td>153.25</td>
<td>24.11</td>
<td>252.35</td>
<td>28.16</td>
<td>99.33</td>
</tr>
<tr>
<td>CLC</td>
<td>Girl</td>
<td>28.24</td>
<td>97.52</td>
<td>27.68</td>
<td>109.62</td>
<td>28.87</td>
<td>84.35</td>
</tr>
<tr>
<td>CLC</td>
<td>Lena</td>
<td>28.39</td>
<td>94.21</td>
<td>25.26</td>
<td>146.89</td>
<td>28.97</td>
<td>82.43</td>
</tr>
<tr>
<td>CLC</td>
<td>Pepper</td>
<td>26.99</td>
<td>130.04</td>
<td>24.82</td>
<td>214.33</td>
<td>27.39</td>
<td>118.60</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Algorithms</th>
<th>Images</th>
<th>0.13 PSNR</th>
<th>MSE</th>
<th>0.19 PSNR</th>
<th>MSE</th>
<th>0.25 PSNR</th>
<th>MSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLC+SEC</td>
<td>Airplane</td>
<td>21.87</td>
<td>422.75</td>
<td>21.90</td>
<td>419.84</td>
<td>23.26</td>
<td>306.96</td>
</tr>
<tr>
<td>CLC+SEC</td>
<td>Girl</td>
<td>25.87</td>
<td>168.30</td>
<td>23.89</td>
<td>265.51</td>
<td>25.95</td>
<td>165.23</td>
</tr>
<tr>
<td>CLC+SEC</td>
<td>Lena</td>
<td>24.27</td>
<td>243.27</td>
<td>23.19</td>
<td>311.95</td>
<td>25.11</td>
<td>200.48</td>
</tr>
<tr>
<td>CLC+SEC</td>
<td>Pepper</td>
<td>22.72</td>
<td>347.60</td>
<td>21.61</td>
<td>448.83</td>
<td>23.58</td>
<td>285.16</td>
</tr>
</tbody>
</table>

Table II: Margin specifications PSNR and MSE of the images reconstructed from bit rate of various sizes designed by the LBG and the proposed CLC algorithm.

Table III: PSNR and MSE of the images reconstructed from bit rate of various size designed by the CLC and the proposed CLC+SEC algorithm.
Table IV: PSNR and MSE of the images reconstructed from various compression ratios (CR) designed by the CLC+SEC and CLC algorithms.

<table>
<thead>
<tr>
<th>Images</th>
<th>64.00 (0.50 bpp)</th>
<th>16.00 (0.62 bpp)</th>
<th>64.00 (0.84 bpp)</th>
<th>16.00 (0.62 bpp)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Airplane</td>
<td>155.98</td>
<td>155.98</td>
<td>155.98</td>
<td>155.98</td>
</tr>
<tr>
<td>PSNR</td>
<td>26.20</td>
<td>26.20</td>
<td>26.20</td>
<td>26.20</td>
</tr>
<tr>
<td>MSE</td>
<td>223.40</td>
<td>29.40</td>
<td>419.84</td>
<td>419.84</td>
</tr>
<tr>
<td>Lena</td>
<td>90.17</td>
<td>90.17</td>
<td>90.17</td>
<td>90.17</td>
</tr>
<tr>
<td>PSNR</td>
<td>28.58</td>
<td>28.58</td>
<td>28.58</td>
<td>28.58</td>
</tr>
<tr>
<td>MSE</td>
<td>71.96</td>
<td>71.96</td>
<td>71.96</td>
<td>71.96</td>
</tr>
<tr>
<td>Pepper</td>
<td>90.17</td>
<td>90.17</td>
<td>90.17</td>
<td>90.17</td>
</tr>
<tr>
<td>MSE</td>
<td>72.79</td>
<td>72.79</td>
<td>72.79</td>
<td>72.79</td>
</tr>
<tr>
<td>Pepper</td>
<td>90.17</td>
<td>90.17</td>
<td>90.17</td>
<td>90.17</td>
</tr>
<tr>
<td>PSNR</td>
<td>27.57</td>
<td>27.57</td>
<td>27.57</td>
<td>27.57</td>
</tr>
<tr>
<td>MSE</td>
<td>113.78</td>
<td>113.78</td>
<td>113.78</td>
<td>113.78</td>
</tr>
<tr>
<td>Pepper</td>
<td>90.17</td>
<td>90.17</td>
<td>90.17</td>
<td>90.17</td>
</tr>
<tr>
<td>PSNR</td>
<td>25.08</td>
<td>25.08</td>
<td>25.08</td>
<td>25.08</td>
</tr>
<tr>
<td>MSE</td>
<td>201.87</td>
<td>201.87</td>
<td>201.87</td>
<td>201.87</td>
</tr>
</tbody>
</table>

Table V: PSNR and MSE of the images reconstructed from various compression ratios designed by the CLC, CLC+SEC, LBG+SEC, and WT+GGCLN algorithm.

| Algorithms | Compression Ratios (CR) | Images       | CLC+SEC     | CLC         |
|------------|-------------------------|--------------|-------------|
|            | 16.00 (0.50 bpp)        | Airplane     | 155.98      | 155.98      |
|            |                         | PSNR         | 26.20       | 26.20       |
|            |                         | MSE          | 223.40      | 223.40      |
|            |                         | Girl         | 90.17       | 90.17       |
|            |                         | PSNR         | 28.58       | 28.58       |
|            |                         | MSE          | 71.96       | 71.96       |
|            |                         | Lena         | 90.17       | 90.17       |
|            |                         | PSNR         | 28.55       | 28.55       |
|            |                         | MSE          | 72.79       | 72.79       |
|            |                         | Pepper       | 90.17       | 90.17       |
|            |                         | PSNR         | 27.57       | 27.57       |
|            |                         | MSE          | 113.78      | 113.78      |
|            |                         | Pepper       | 90.17       | 90.17       |
|            |                         | PSNR         | 25.08       | 25.08       |
|            |                         | MSE          | 201.87      | 201.87      |

Table VI: Comparison of PSNR with conventional method.

<table>
<thead>
<tr>
<th>Original Images</th>
<th>WT+GGCLN</th>
<th>CLC+SEC</th>
</tr>
</thead>
<tbody>
<tr>
<td>Airplane</td>
<td>27.66 dB</td>
<td>22.49 dB</td>
</tr>
<tr>
<td>Lena</td>
<td>29.52 dB</td>
<td>24.34 dB</td>
</tr>
<tr>
<td>Girl</td>
<td>31.90 dB</td>
<td>26.76 dB</td>
</tr>
<tr>
<td>Milkdrop</td>
<td>N/A</td>
<td>32.56 dB</td>
</tr>
</tbody>
</table>

VI. CONCLUSIONS

An approach method for image coding is proposed in this paper which consists of three contributions. Firstly, we exploit wavelet transformation in the coding process by classifying pixel value into 8 groups based upon frequency components. The block location and size depending on adjacent pixels with similar intensity values are then determined yielding a decreased bit rate while maintaining PSNR. Secondly, fuzzy C mean is used to further improved the bit rate and PSNR by further classifying member in the each groups. Thirdly, we propose a method of system error compensation on Vector Quantization (VQ) by using principle component analysis and discrete wavelet transform to performed on the system error. The experiments on the proposed method have been performed with four standard images. The experimental results reveal the proposed method is evaluated as 0.62 bpp, and average 30.14 dB for bit rate and PSNR respectively, that is improved by 26.19% and 1.50% for bit rate and PSNR respectively comparing with the conventional method as cited in Table V (WT+GGCLN).

REFERENCES