A New Lossy Image Compression Technique Using DCT, Round Variable Method & Run Length Encoding

Nitesh Agarwal
Department of Computer Science
Jodhpur Institute of Engineering & Technology
Jodhpur, India

Dr. Sandeep Mathur
Department of Mathematics
Jodhpur Institute of Engineering & Technology
Jodhpur, India

Dr. Anjali Mathur
Department of Mathematics
Jodhpur Institute of Engineering & Technology
Jodhpur, India

Abstract—An image has important roles in human being life, it is used to store our memorable movement, used to represent a thing, used for communication purpose, even in video processing system it has important role because each frame of a video is made of an image. A high-quality image may require 10 to 100 million bits for representation. For example, a nearly photographic image requires approximately 1,280 rows of 800 pixels each, with 24 bits of color information per pixel, that is, a total of 24,576,000 bits, or 3,072,000 bytes. The large data files associated with images thus drive the need for extremely high compression ratios to make storage (particularly of movies) practical. Image compression process use two technique to compress image lossless image compression & lossy image compression. Images that provide numerical, secure & financial information compressed using lossless image compression because we required original data back after decompression process. Lossless image compression use some entropy encoding technique but its compression ratio is low w.r.t lossy image compression. Images that provide numerical, secure & financial information compressed using lossless image compression because we required original data back after decompression process. Lossless image compression use some entropy encoding technique but its compression ratio is low w.r.t lossy image compression. Images that provide numerical, secure & financial information compressed using lossless image compression because we required original data back after decompression process. Lossless image compression use some entropy encoding technique but its compression ratio is low w.r.t lossy image compression. But other images like multimedia images can be compressed using lossy image compression. Lossy image compression require some transformation, quantization & entropy encodings to compress an image. Normally lossy image compression is process as follow

Key Words: RVM, DCT, RLE, MSE, PSNR.

1. INTRODUCTION

Digital devices & computational resources have limited communication & storage capabilities for example without compression, a CD with a storage capacity of approximately 600 million bytes only. Hence we need to compress image data for storing purpose as well as for communication over a network for example if there is a video conference organize by an organization which has a low bandwidth of a network, if image quality high for each frame of a video then it is difficult to deliver idea by video using low bandwidth over network provided by organization, hence organization need some image compression technique by which it compress the video before communication to provide a good communication in synchronize manner. Image compression process use two technique to compress image lossless image compression & lossy image compression. Images that provide numerical, secure & financial information compressed using lossless image compression because we required original data back after decompression process. Lossless image compression use some entropy encoding technique but its compression ratio is low w.r.t lossy image compression. But other images like multimedia images can be compressed using lossy image compression. Lossy image compression require some transformation, quantization & entropy encodings to compress an image. Normally lossy image compression is process as follow
2-D DCT can be defined as N*N. An image is a 2D pixel matrix hence 2D DCT is used to domain by partitioning image pixel matrix into blocks of size table, RVM method & Run Length Encoding (RLE). quantization of transform value using standard quantization transformation DCT (Discrete Cosine Transform), this lossy image compression mostly based on a lossy image compression using RVM method. RLE can compressed an image only if input values repeated at encoding techniques. In this paper RLE is used for encoding & paper introduce a new module named as RVM before entropy this idea this paper introduce a new module named as RVM before entropy encoding & this QDCT block dequantize by following formula & this QDCT block dequantize by following formula

A Quantizer simply reduces the number of bits needed to store the transformed coefficients by reducing the precision of those values. Since this is a many-to-one mapping, it’s a lossy process and is the main source of compression in an encoder. The quantization matrix is designed to provide more resolution to more perceivable frequency components over less perceivable components (usually lower frequencies over high frequencies) in addition to transforming as many components to 0, which can be encoded with greatest efficiency. A DCT block is quantize using following formula

\[
QDCT(i,j)=\text{ROUND}\left(DCT(i,j) \times QT(i,j)\right)
\]

(1)

Where (i,t) define position of input & output value, QDCT is DCT block after quantization, QT is standard quantization matrices & defined as

<table>
<thead>
<tr>
<th>i</th>
<th>16</th>
<th>11</th>
<th>10</th>
<th>16</th>
<th>24</th>
<th>40</th>
<th>51</th>
<th>61</th>
</tr>
</thead>
<tbody>
<tr>
<td>j</td>
<td>12</td>
<td>12</td>
<td>14</td>
<td>19</td>
<td>26</td>
<td>58</td>
<td>60</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>13</td>
<td>16</td>
<td>24</td>
<td>40</td>
<td>57</td>
<td>69</td>
<td>56</td>
</tr>
<tr>
<td></td>
<td>14</td>
<td>17</td>
<td>22</td>
<td>29</td>
<td>51</td>
<td>87</td>
<td>80</td>
<td>62</td>
</tr>
<tr>
<td></td>
<td>18</td>
<td>22</td>
<td>37</td>
<td>56</td>
<td>68</td>
<td>109</td>
<td>103</td>
<td>77</td>
</tr>
<tr>
<td></td>
<td>24</td>
<td>35</td>
<td>55</td>
<td>64</td>
<td>81</td>
<td>104</td>
<td>113</td>
<td>92</td>
</tr>
<tr>
<td></td>
<td>49</td>
<td>64</td>
<td>78</td>
<td>87</td>
<td>103</td>
<td>121</td>
<td>120</td>
<td>101</td>
</tr>
<tr>
<td></td>
<td>72</td>
<td>92</td>
<td>95</td>
<td>98</td>
<td>112</td>
<td>100</td>
<td>103</td>
<td>99</td>
</tr>
</tbody>
</table>

Table 1: Quantization Matrices [9]

In this lossy image compression QDCT provides input for RVM method.

1.3 RVM (Random Variable Method)

RVM take inputs from quantize block & round the input value according to its variable for ex RVM has input as 101,102,103,104,105,106,107,108,109,110 & variable used in RVM is 5 then it round the each input value as 100,100,100,105,105,105,105,110,110,110 the reason behind

\[
\alpha(v) = \begin{cases} 
\frac{1}{\sqrt{N}} & \text{for } v = 0 \\
\frac{2}{\sqrt{N}} & \text{for } v \neq 0 
\end{cases}
\]

(3)
to use this method before encoding is that an entropy encoding technique gives a high compression as the input values have repeated data sequentially. It work as

\[ RVM(i, j) = \text{ROUND} \left( \frac{QDCT(i, j)}{X} \right) \] (7)

& its inverse work as

\[ QDCT(i, j) = \text{ROUND} \left( RVM(i, j) \ast X \right) \] (8)

Where X is variable used in RVM method [10].

1.4 RLE (Run Length Encoding) [12]

This is a very simple compression technique method used for compressing sequential data. Many digital image consist pixel values that are repeats sequentially for such type of image RLE is useful. In proposed method RLE receive sequential data from RVM block & store input value that repeats & no of time that input value repeat sequentially. For ex. RVM block has data as

<table>
<thead>
<tr>
<th>0</th>
<th>0</th>
<th>0</th>
<th>1</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>6</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>8</td>
<td>8</td>
<td>8</td>
</tr>
</tbody>
</table>

Fig 3: Pixel matrix of size 4*4

To store above 4*4 RVM block total 16 values are required to store but after applying RLE only 12 values are required to store such as

<table>
<thead>
<tr>
<th>0</th>
<th>3</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>1</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>8</td>
<td>3</td>
</tr>
</tbody>
</table>

Fig 4: matrix After RLE

2. MAIN RESULTS

2.1 Implementation of Proposed Lossy Image Compression

This paper describe how a b/w (8 bit) image compressed using proposed method

Steps involved in this implementation

1. Create pixel matrix of the image & divided it into blocks of size 8*8
2. Apply FDCT (Forward Discrete Sine Transform) on each 8*8 block of pixel matrix to get equivalent 8*8 DCT blocks using eq (1).
3. Apply eq (5) on each block of DCT to get QDCT block.
4. Apply eq (7) on each block of QDCT to get RVM block.
5. Combine each RVM block & apply RLE on combine block & store this encoded block on secondary storage.
6. To get Original image read RLE block from secondary storage & decode it to get combine RVM block & divide it into 8*8 small RVM block
7. Apply eq (8) on each RVM block to get QDCT blocks.
8. To get DCT blocks apply eq (6) on each QDCT block.
9. Apply eq (2) on each DCT block to get IDCT blocks.
10. Combine all IDCT blocks to get pixel matrix.
11. Using pixel matrix we get required image.
12. Now we Find MSE (Mean Squared Error), PSNR (Peak Signal To Noise Ratio) & CR (Compression Ration) to determine quality of image obtain by proposed method for each X variable used in RVM method. MSEx, PSNRx & CRx calculated by following formulas [11] -

\[ MSE_x = \frac{1}{H \times W} \sum_{x=0}^{H-1} \sum_{y=0}^{W-1} [O(x, y) - M_x(x, y)]^2 \] (9)

\[ PSNR_x = 20 \log_{10} (\text{MAX}) - 10 \log_{10} (\text{MSE}_x) \] (10)

\[ CR_x = \frac{\text{Original Image size}}{\text{Output Image size}} \] (11)

Where H=Height of Image, W=Width of Image, variable MAX shows max value of a pixel for example here image is 8 bit hence MAX=255, MSEx, PSNRx & CRx is MSE, PSNR & CR at variable X Quality of image=RVM method

Proposed method is depend on MSE & PSNR value, if as the MSE value decreases then we get a bad quality of image by proposed method & if as the PSNR value increases we get a better quality image hence on basis of this MSE proposed method gives compressed image with best quality.

2.2 Outputs

2.2.1 Image Compression without RVM method

Fig 5: Uncompressed Image  
Size= 768 KB  

Fig 6: Compressed Image  
Size= 103 KB
2.2.2 Image Compression with proposed method

### Table 2: MSE, PSNR, CR without RVM

<table>
<thead>
<tr>
<th>Image Compression without RVM</th>
<th>MSE</th>
<th>PSNR</th>
<th>CR</th>
</tr>
</thead>
<tbody>
<tr>
<td>X=1</td>
<td>17</td>
<td>36</td>
<td>7</td>
</tr>
<tr>
<td>X=2</td>
<td>41</td>
<td>32</td>
<td>11</td>
</tr>
<tr>
<td>X=3</td>
<td>38</td>
<td>32</td>
<td>14</td>
</tr>
<tr>
<td>X=4</td>
<td>54</td>
<td>31</td>
<td>16</td>
</tr>
<tr>
<td>X=5</td>
<td>61</td>
<td>30</td>
<td>19</td>
</tr>
<tr>
<td>X=6</td>
<td>80</td>
<td>29</td>
<td>24</td>
</tr>
<tr>
<td>X=7</td>
<td>87</td>
<td>29</td>
<td>24</td>
</tr>
<tr>
<td>X=8</td>
<td>107</td>
<td>28</td>
<td>24</td>
</tr>
<tr>
<td>X=9</td>
<td>121</td>
<td>27</td>
<td>32</td>
</tr>
<tr>
<td>X=10</td>
<td>138</td>
<td>27</td>
<td>32</td>
</tr>
<tr>
<td>X=11</td>
<td>153</td>
<td>26</td>
<td>32</td>
</tr>
</tbody>
</table>

### Table 3: MSE\(_X\), PSNR\(_X\), CR\(_X\) on different value of X

<table>
<thead>
<tr>
<th>X</th>
<th>MSE(_X)</th>
<th>PSNR(_X)</th>
<th>CR(_X)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>17</td>
<td>36</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>41</td>
<td>32</td>
<td>11</td>
</tr>
<tr>
<td>3</td>
<td>38</td>
<td>32</td>
<td>14</td>
</tr>
<tr>
<td>4</td>
<td>54</td>
<td>31</td>
<td>16</td>
</tr>
<tr>
<td>5</td>
<td>61</td>
<td>30</td>
<td>19</td>
</tr>
<tr>
<td>6</td>
<td>80</td>
<td>29</td>
<td>24</td>
</tr>
<tr>
<td>7</td>
<td>87</td>
<td>29</td>
<td>24</td>
</tr>
<tr>
<td>8</td>
<td>107</td>
<td>28</td>
<td>24</td>
</tr>
<tr>
<td>9</td>
<td>121</td>
<td>27</td>
<td>32</td>
</tr>
<tr>
<td>10</td>
<td>138</td>
<td>27</td>
<td>32</td>
</tr>
<tr>
<td>11</td>
<td>153</td>
<td>26</td>
<td>32</td>
</tr>
</tbody>
</table>

#### 2.2.2.1 Graphs

1) **RVM variable X vs. CR\(_X\)**

![Fig 8: Variation in CR\(_X\) with different value of RVM variable X](image)

2) **RVM variable X vs. MSE\(_X\)**

![Fig 9: Variation in MSE\(_X\) with different value of RVM variable X](image)

---

Fig 7: Proposed Lossy Image Compression with variation in RVM Variable X

Fig 8: Variation in CR\(_X\) with different value of RVM variable X

Fig 9: Variation in MSE\(_X\) with different value of RVM variable X
3) RVM variable X vs. PSNRx

Fig 10: Variation in PSNRx with different value of RVM variable X

4) RVM variable X vs. PSNRx & MSEx

Fig 11: Variation in PSNRx & MSEx with different value of RVM variable X

5) MSEx vs. PSNRx

Fig 12: Graphical comparison between MSEx & PSNRx

3. CONCLUSION

The result presented in this document shows that

1. The results shows that as the value of variable X increases storage size of image decreases as shown in Fig 7.
2. As the value of X increases CRx also increases as shown in Fig 8.
3. As the value of X increases proposed process add more noises in the image i.e. value of MSEx increases as shown in Fig 9.
4. As the value of X increases PSNRx value decreases as shown in Fig 10.
5. As the MSEx value decreases & PSNRx increases quality of image improves but CRx decreases.
6. Fig 11 show that for value 2 & 3 of variable X we get good quality of compressed image.
7. Fig 11 shows that after the value 3 of variable X difference between MSEx & PSNRx increases as the value of variable X increases i.e more noises is added to the image.
8. Fig 12 shows that as the value of MSEx increases value of PSNRx variable decrease i.e. quality of image decreases.

REFERENCES


