Chapter 5

Compression of MR Images using DWT by Comparing RGB and YCbCr color spaces

This chapter consists of a lossy image compression algorithm dedicated to the medical images by comparison of RGB and YCbCr color space. Several lossy/lossless transform coding techniques are used for medical image compresion. Discrete Wavelet Transform (DWT) is one such widely used technique. After a preprocessing step (remove the mean and RGB to YCbCr transformation), the DWT is applied and followed by bisection method that include thresholding, the quantization, dequantization, the Inverse Discrete Wavelet Transform (IDWT), YCbCr to RGB transform of mean recovering. To obtain the best compression ratio (CR), encoding algorithm is used for compressing the input medical image in to three matrices and forward to DWT block a coresponding (m×n) vector containg the maximum possible run of zeros at its end. In the last step decoding algorithm is used to decompress the image using IDWT to get three matrices of medical image.

5.1 Introduction

The basic objective of image compression is to reduced the size of an image data for transmission or store it in an efficient manner, while maintaining the suitable quality of reconstructed images [25,49,51,56,59,115,116]. The easy and reliable digital transmission
and storage of biomedical images would be a tremendous boon to the medical practices. This can help in instant availability of earlier imaging studies when patients are re-admitted [25-27]. Both Medical and surgical teams indulging on patient care could have simultaneous access to imaging studies on monitors throughout the hospital. This long-term digital archiving or rapid transmission is prohibitive without the use of image compression to reduce the file sizes.

There are two basic types of image compression schemes:

The first is lossless compression scheme, that encodes and decodes the data perfectly and reconstructed image matches exactly with the original image which means there is no loss of data with no degradation. In this scheme the coding techniques are Huffman encoding, entropy encoding, and run length ending.

The second lossy compression scheme is used for the sake of using a minimum storage space. In this scheme there is trade-off between compression and image quality. In lossy compression, the final decompressed image must be visually lossless and consist of removing the redundant information in adjacent pixels to minimize the number of bits [4, 46, 47, 59, 66, 117].

In preprocessing step, many decorrelating transforms like YCbCr, YUV, KLT, YIQ [34-36] are used to reduce the correlation between the R, G and B plane. We can preferably use one of these colors space transforms before the application of wavelet transform. In this paper we are considering the YCbCr transform to reduce the correlation between the R, G and B space [37].
Block-based DWT [28, 117] shown in Fig. 5.1, decomposes a broadband signal into two subbands with smaller bandwidths and slower sample rates. A series of HP and LP FIR filters are used to repeatedly divide the input frequency range. The input signals are in the form of frames having frame size as multiple of $2^n$, where $n$ is the number of levels. Each unit consists of a LP and HP FIR filter pair. The halfband filters with a cutoff frequency of $F_s / 4$ are obtained by decimating the output of each LP and HP filter by a factor of 2.

The aim of this chapter is to evaluate the performance of lossy medical image compression wavelet transforms followed by wavelet encoders experiments which have been performed by using magnetic resonance images (MRI) as test images. The performances of the medical images have been evaluated in terms of peak signal-to-noise ratio (PSNR), bit rate (bpp) and the compression ratio (CR). In last step, decoding algorithm has been used to decompress the image using IDWT which has been applied to get three matrices of medical image.

5.2 Methodology

The block diagram of DWT based medical image compression/decompression model is shown in Fig. 5.2. The method is dedicated to lossy medical image compression DWT based and two phases of compression/decompression. The input to the system is a medical image and

![Fig. 5.1. Block-based DWT](image-url)
the output is the compressed one. The compression technique is built with several steps and each will be explained in details.

5.2.1 RGB to YCbCr transformation [34-36]

RGB is not very efficient when dealing with real-word images. All three RGB components need to be of equal bandwidth to generate any color within the RGB color cube. The result of this is a frame buffer that has the same pixel depth and display resolution for each RGB component. Also, processing an image in the RGB color space is usually not the most efficient method. To modify the intensity or color of a given pixel, the three RGB values must be read from the frame buffer, the intensity or color calculated, the desired modifications performed, and the new RGB values calculated and written back to the frame buffer. If the system had access to an image stored directly in the intensity and color format, some processing steps would be faster. For these and other reasons, many video standards use luma and two color difference signals. The most common are the YUV, YIQ and YCbCr color spaces.
In this chapter the RGB to YCbCr transformation, the mean value of three plane images R, G and B are removed and the almost signal energy of the new transformed YCbCr image is contained in the Y plane. Consequently, we can achieve high compression ratio in the Cb and Cr without losses in quality of compressed image when returned to the original RGB space.

The transformation from RGB to YCbCr is performed respecting to

\[
\begin{bmatrix}
Y \\
Cb \\
Cr
\end{bmatrix} = \begin{bmatrix}
16 \\
128 + \frac{1}{256} \\
128
\end{bmatrix} \begin{bmatrix}
65.738 & 129.057 & 25.064 \\
-37.945 & -74.494 & 112.439 \\
\end{bmatrix} \begin{bmatrix}
R \\
G \\
B
\end{bmatrix} \tag{5.1}
\]

Where R, G and B take the typical values from 0 to 255 (8-bit precision), Y is the same range (0-255), and Cb, Cr components are into the range (16-240).

The inverse transformation is expressed by

\[
\begin{bmatrix}
R \\
G \\
B
\end{bmatrix} = \begin{bmatrix}
1.0 & 0.000 & 1.371 \\
1.0 & -0.336 & -0.698 \\
1.0 & 1.732 & 0.000
\end{bmatrix} \begin{bmatrix}
Y \\
Cb \\
Cr
\end{bmatrix} \tag{5.2}
\]

Fig. 5.3 is a original MRI image and we define the RGB to YCbCr transformation for removing the mean values of R, G, and B plane. Fig. 5.4(a),

![Fig. 5.3 Original MRI Image](image-url)
**Fig. 5.4 Block displays of RGB and YcbCr colors**

Fig. 5.4(b) and Fig. 5.4(c) are the block displays an M-by-N matrix element values to specified range of RGB colors. Fig. 5.4(d), Fig. 5.4(e) and Fig. 5.4(f) are the block displays specified range of YCbCr colors. By Fig. 5.4, we confirm that RGB-YCbCr transformation approach is necessary to get superior performance.

### 5.2.2 Block based DWT transform

For any color image, after the RGB to YCbCr transformation, each one of the new three planes YCbCr are partitioned to blocks and each block is DWT transformed. The DWT blocks perform single-level one-dimensional wavelet decomposition with respect to either a particular wavelet (Daubechies, Coiflets, Symlets, Discrete Meyer, Biorthogonal, and Reverse Biorthogonal) or particular wavelet LP and HP decomposition filters.
As shown in Fig. 5.5 the original signals are firstly decomposed into two subspaces, low-frequency subband and high frequency subband. It is first scanned in a horizontal direction and passed through LP and HP decomposition filters producing low frequency as well as high-frequency data in the horizontal direction. Filtered output data are then scanned in a vertical direction and again these filters are applied separately to generate different frequency subbands. The transform generates subbands LL, LH, HL and HH each with one-fourth the size of the original image. Most of the energy is concentrated in low-frequency subband LL, whereas higher-frequency subbands LH, HL and HH contain detailed information of the image in vertical, horizontal and diagonal directions, respectively. For higher level decomposition, DWT can be applied again to the LL subband recursively in a similar way to further compact energy into fewer low-frequency coefficients. The appropriate choice of filters for the transform is very important to achieve high coding efficiency.

Fig. 5.5 Pyramidal algorithm of one-level forward DWT decomposition.
5-2-3 Quantization and Transform coding

A transform coder [118] decomposes a signal in an orthogonal basis and quantizes the decomposition coefficients. The distortion of the restored signal is minimized by optimizing the quantization, the basis, and the bit allocation. It is desirable to perform quantization by dividing the transformed coefficients by quantization value. For low frequency coefficients are divided by smaller values while the high frequency coefficients are divided by larger values.

In this thesis, we code the data using transform coding scheme of following steps:

5.2.3.1 Source Coding

Source coding is to represent information in bits, with the natural aim of using a small number of bits. When the information can be exactly recovered from the bits, the source coding or compression is called lossless; otherwise, it is called lossy. The transform codes in this article are lossy. However, lossless entropy codes appear as components of transform codes, so both lossless and lossy compression are of present interest. In our discussion, the “information” is denoted by a real column vector $x \in \mathbb{R}^N$ or a sequence of such vectors. A vector

![Fig. 5.6. Transform Coding scheme](image-url)
might be formed from pixel values in an image or by sampling an audio signal; $K.N$ pixels can be arranged as a sequence of $K$ vectors of length $N$. The vector length $N$ is defined such that each vector in a sequence is encoded independently. For the purpose of building a mathematical theory, the source vectors are assumed to be realizations of a random vector $\mathbf{x}$ with a known distribution.

The distribution could be purely empirical. A source code is comprised of two mappings: an encoder and a decoder. The encoder maps any vector $\mathbf{x} \in \mathbb{R}^N$ to a finite string of bits, and the decoder maps any of these strings of bits to an approximation $\hat{x} \in \mathbb{R}^N$. The encoder mapping can always be factored as $\gamma \circ \alpha$, where $\alpha$ is a mapping from $\mathbb{R}^N$ to some discrete set $I$ and $\gamma$ is an invertible mapping from $I$ to strings of bits. The former is called a lossy encoder and the latter a lossless code or an entropy code. The decoder inverts $\gamma$ and then approximates $x$ from the index $\alpha(x) \in I$. This is shown in the top half of Fig. 5.6. It is assumed that communication between the encoder and decoder is perfect.

To assess the quality of a lossy source code, we need numerical measures of approximation accuracy and description length. The measure for description length is simply the expected number of bits output by the encoder divided by $N$; this is called the rate in bits per scalar sample and denoted by $R$. Here we will measure approximation accuracy by squared Euclidean norm divided by the vector length

$$d(x, \hat{x}) = \frac{1}{N} \|x - \hat{x}\|^2 = \frac{1}{N} \sum_{i=1}^{N} (x_i - \hat{x}_i)^2$$  \hspace{1cm} (5.3)
This accuracy measure is conventional and usually leads to the easiest mathematical results, though the theory of source coding has been developed with quite general measures [1]. The expected value of $d(x, \hat{x})$ is called the mean-squared error (MSE) distortion and is denoted by $D = E[d(x, \hat{x})]$. The normalizations by $N$ make it possible to fairly compare source codes with different lengths.

### 5.2.3.2 Constrained Source Coding

Transform codes are the most used source codes because they are easy to apply at any rate and even with very large values of $N$. The essence of transform coding is the modularization shown in the bottom half of Fig. 5.6. The mapping $\alpha$ is implemented in two steps. First, an invertible linear transform of the source vector $x$ is computed, producing $y = Tx$. Each component of $y$ is called a transform coefficient. The $N$ transform coefficients are then quantized independently of each other by $N$ scalar quantizers. This is called scalar quantization since each scalar component of $y$ is treated separately. Finally, the quantizer indexes that correspond to the transform coefficients are compressed with an entropy code to produce the sequence of bits that represent the data.

To reconstruct an approximation of $x$, the decoder essentially reverses the steps of the encoder. The action of the entropy coder can be inverted to recover the quantizer indices. Then the decoders of the scalar quantizers produce a vector $\hat{y}$ of estimates of the transform coefficients. To complete the reconstruction, a linear transform is
applied to \( \mathbf{y} \) to produce the approximation \( \mathbf{x} \). This final step usually uses the transform \( T^{-1} \), but for generality the transform is denoted \( U \).

Most source codes cannot be implemented in the two stages of linear transform and scalar quantization. Thus, a transform code is an example of a constrained source code. Constrained source codes are loosely speaking, source codes that are suboptimal but have low complexity. The simplicity of transform coding allows large values of \( N \) to be practical. Computing the transform \( T \) requires at most \( N^2 \) multiplications and \( N \, (N - 1) \) additions. Specially structured transforms like discrete Fourier, cosine, and wavelet transforms are often used to reduce the complexity of this step, but this is merely icing on the cake. The great reduction from the exponential complexity of a general source code to the (at most) quadratic complexity of a transform code comes from using linear transforms and scalar quantization.

### 5.2.3.3 Entropy Codes

Entropy codes are used for lossless coding of discrete random variables. Consider the discrete random variable \( \mathbf{z} \) with alphabet \( \mathcal{I} \). An entropy code \( \gamma \) assigns a unique binary string, called a codeword, to each \( \mathcal{I} \) (See Fig. 5.6). Since the codewords are unique, an entropy code is always invertible. However, we will place more restrictive conditions on entropy codes so they can be used on sequences of realizations of \( \mathbf{z} \). The extension of \( \gamma \) maps the finite sequence \((z_1, z_2, z_3, ..., z_k)\) to the concatenation of the outputs of \( \gamma \) with each input, \( \gamma(z_1), \gamma(z_2), ... \gamma(z_k) \). A code is called uniquely decodable if its extension is one-to-one. A uniquely decodable code can be applied to message sequences without adding any “punctuation” to show where
one codeword ends and the next begins. In a prefix code, no codeword is the prefix of any other codeword. Prefix codes are guaranteed to be uniquely decodable.

### 5.2.3.4 Quantizers

A quantizer $q$ is a mapping from a source alphabet $\mathbb{R}^N$ to a reproduction code-book $\mathcal{L} = \{\mathcal{L}_i\}_{i \in I} \subset \mathbb{R}^N$, where $I$ is an arbitrary countable index set. Quantization can be decomposed into two operations $q = \beta \alpha$, as shown in Fig. 5.6. The lossy encoder $\alpha: \mathbb{R}^N \rightarrow I$ is specified by a partition of $\mathbb{R}^N$ into partition cells $S_i = \{x \in \mathbb{R}^N | \alpha(x) = i\}, i = I$. The reproduction decoder $\beta: I \rightarrow \mathbb{R}^N$ is specified by the codebook $C$. If $N = 1$, the quantizer is called a scalar quantizer; for $N > 1$, it is a vector quantizer.

The quality of a quantizer is determined by its distortion and rate. The MSE distortion for quantizing random vector $x \in \mathbb{R}^N$ is

$$D = \mathbb{E}[d(x, q(x))] = \frac{1}{N} \mathbb{E}[\|x - q(x)\|^2]$$

(5.4)

The rate can be measured in a few ways. The lossy encoder output $\alpha(x)$ is a discrete random variable that usually should be entropy coded because the output symbols will have unequal probabilities. Associating an entropy code $\gamma$ to the quantizer gives a variable-rate quantizer specified by $(\alpha, \beta, \gamma)$. The rate of the quantizer is the expected code length of $\gamma$ divided by $N$. Not specifying an entropy code (or specifying the use of fixed-rate binary expansion) gives a fixed-rate quantizer with rate $R = N^{-1} \log_2 |I|$. Measuring the rate by the idealized performance of an entropy code gives $R = N^{-1} H(\alpha(x))$; the quantizer in this case is called entropy constrained.
5.2.3.5 Bit Allocation

Coding (quantizing and entropy coding) each transform coefficient separately splits the total number of bits among the transform coefficients in some manner. Whether done with conscious effort or implicitly, this is a bit allocation among the components.

Bit allocation problems can be stated in a single common form: One is given a set of quantizers described by their distortion-rate performances as

\[ D_i = g_i(R_i), R_i \in r_i, i = 1, 2, \ldots, N. \quad (5.5) \]

Each set of available rates \( r_i \) is a subset the nonnegative real numbers and may be discrete or continuous. The problem is to minimize the average distortion \( D = N^{-1} \sum_{i=1}^{N} D_i \) given a maximum average rate \( R = N^{-1} \sum_{i=1}^{N} R_i \).

With uniform quantizers, bit allocation is nothing more than choosing a step size \( \Delta_i \) for each of the \( N \) components. The equal-distortion property of the analytical bit allocation solution gives a simple rule: Make all of the step sizes equal. This will be referred to as “lazy” bit allocation.

5.3 Measurements

Several quality measures can be found in open literature of the field. The mean square error (MSE) and the Peak signal to noise ratio (PSNR) are the most used measures.
Mean square error (MSE) is the some sort of average or sum of the squares of the error between two images. For M×N images u(m,n) and ū(m,n), the least square error (LSE) is,

\[
LSE = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} |u(m,n) - ū(m,n)|^2
\]  \hspace{1cm} (5.6)

and average LSE is called the Mean square error (MSE),

\[
MSE = \frac{1}{MN} \sum_{m=1}^{M} \sum_{n=1}^{N} E[|u(m,n) - ū(m,n)|^2]
\]  \hspace{1cm} (5.7)

Where u(m,n) and ū(m,n) are the original and reconstructed intensities belonging to R, G and B plane. The PSNR is defined in decibels (dB) as,

\[
PSNR = 10 \log_{10} \frac{\sigma^2}{MSE}
\]  \hspace{1cm} (5.8)

Where \(\sigma^2\) is the variance of the original image. For medical image we used the relation given in

\[
PSNR = 10 \log_{10} \left( \frac{\sigma^2 \times 3}{MSE(R) + MSE(G) + MSE(B)} \right)
\]  \hspace{1cm} (5.9)

The size of the compressed image is evaluated with the CR or Bit-rate per pixel (BPP) defined by

\[
CR = \frac{\text{Original image in bytes}}{\text{compressed image in bits}}
\]  \hspace{1cm} (5.10)

The various steps during compression and decompression algorithms are summarized as follows:

**Compression algorithm:**

- Input: Medical image I(RGB)
- Break the input image into three matrices I(R), I(G) and I(B)
- Transformation of the I(R), I(G) and I(B) matrices into I(Y), I(Cb) and I(Cr)
- Perform DWT transform of sub-band I(Y), I(Cb) and I(Cr) separately
Transform coder decomposes and quantizes the decomposition coefficients

Output: Compressed medical image $I(YCbCr)$

**Decompression algorithm:**

- Input: Compressed medical image $I(YCbCr)$
- Inverse sub-band transform and dequantization of reproduction code
- IDWT is applied and get $\hat{I}(Y)$, $\hat{I}(Cb)$ and $\hat{I}(Cr)$
- Transformation of the $\hat{I}(Y)$, $\hat{I}(Cb)$ and $\hat{I}(Cr)$ into $\hat{I}(R)$, $\hat{I}(G)$ and $\hat{I}(B)$
- Convert $\hat{I}(R)$, $\hat{I}(G)$ and $\hat{I}(B)$ to $\hat{I}(RGB)$
- Output: Decompressed medical image $\hat{I}(RGB)$

**5.4 Result and discussion**

Various results have been got hold of and summarized after performing different experiments with YCbCr color space on standard medical MR images (Fig. 5.7). The YCbCr transform applied in RGB color space and the respective reconstructed images are shown in Fig. 5.8. Different size images have been tested on the different color images both in the RGB space and in the YCbCr color space.
Fig. 5.7. Original medical MR images

Fig. 5.8. Reconstructed medical MR images
<table>
<thead>
<tr>
<th>Medical Images</th>
<th>PSNR</th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th>Bpp</th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R</td>
<td>G</td>
<td>B</td>
<td>Y</td>
<td>Cb</td>
<td>Cr</td>
<td>R</td>
<td>G</td>
<td>B</td>
<td>Y</td>
<td>Cb</td>
</tr>
<tr>
<td>MR1</td>
<td>11.83</td>
<td>11.83</td>
<td>11.84</td>
<td>12.91</td>
<td>57.93</td>
<td>58.63</td>
<td>0.279</td>
<td>0.279</td>
<td>0.278</td>
<td>0.301</td>
<td>0.502</td>
</tr>
<tr>
<td>MR2</td>
<td>13.28</td>
<td>13.28</td>
<td>13.25</td>
<td>14.26</td>
<td>50.52</td>
<td>47.54</td>
<td>0.225</td>
<td>0.226</td>
<td>0.205</td>
<td>0.253</td>
<td>0.491</td>
</tr>
<tr>
<td>MR3</td>
<td>14.66</td>
<td>14.61</td>
<td>14.76</td>
<td>15.85</td>
<td>36.02</td>
<td>34.1</td>
<td>0.293</td>
<td>0.289</td>
<td>0.287</td>
<td>0.310</td>
<td>0.500</td>
</tr>
<tr>
<td>MR4</td>
<td>11.66</td>
<td>11.54</td>
<td>11.57</td>
<td>12.49</td>
<td>43.85</td>
<td>41.85</td>
<td>0.226</td>
<td>0.223</td>
<td>0.226</td>
<td>0.255</td>
<td>0.503</td>
</tr>
<tr>
<td>MR5</td>
<td>16.84</td>
<td>16.84</td>
<td>16.84</td>
<td>18.08</td>
<td>42.94</td>
<td>60.78</td>
<td>0.226</td>
<td>0.223</td>
<td>0.226</td>
<td>0.255</td>
<td>0.503</td>
</tr>
<tr>
<td>MR6</td>
<td>17.4</td>
<td>17.29</td>
<td>17.24</td>
<td>18.47</td>
<td>52.93</td>
<td>51.08</td>
<td>0.075</td>
<td>0.082</td>
<td>0.088</td>
<td>0.131</td>
<td>0.505</td>
</tr>
<tr>
<td>MR7</td>
<td>18.37</td>
<td>18.37</td>
<td>18.37</td>
<td>19.61</td>
<td>147</td>
<td>147</td>
<td>0.118</td>
<td>0.118</td>
<td>0.118</td>
<td>0.163</td>
<td>0.502</td>
</tr>
<tr>
<td>MR8</td>
<td>18.38</td>
<td>18.38</td>
<td>18.38</td>
<td>19.61</td>
<td>147</td>
<td>147</td>
<td>0.141</td>
<td>0.141</td>
<td>0.141</td>
<td>0.183</td>
<td>0.502</td>
</tr>
<tr>
<td>MR9</td>
<td>18.62</td>
<td>18.62</td>
<td>18.62</td>
<td>19.85</td>
<td>147</td>
<td>147</td>
<td>0.140</td>
<td>0.140</td>
<td>0.140</td>
<td>0.182</td>
<td>0.502</td>
</tr>
<tr>
<td>MR10</td>
<td>16.57</td>
<td>16.57</td>
<td>16.57</td>
<td>17.81</td>
<td>147</td>
<td>147</td>
<td>0.268</td>
<td>0.268</td>
<td>0.268</td>
<td>0.292</td>
<td>0.502</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td>15.76</td>
<td>15.73</td>
<td>15.74</td>
<td>16.89</td>
<td>87.22</td>
<td>88.2</td>
<td>0.188</td>
<td>0.190</td>
<td>0.189</td>
<td>0.223</td>
<td>0.502</td>
</tr>
</tbody>
</table>
Fig. 5.9 Performance of proposed method MR image with RGB and YCbCr as input color space (a) original image, (b) (c) (d) PSNR vs.bpp plot of RGB color space and (e) (f) (g) PSNR vs.bpp plot of YCbCr space.

Tables 5.1 shows the results demonstrating, the superiority of performance of the proposed technique when working in the YCbCr domain. From the Table, it is observed that with PSNR in YCbCr color space increased the percentage of increased PSNR in Cb and Cr color
space is high. Similarly it is observed that, bpp is high in YCbCr color space as compared to RGB color space.

In Fig. 5.9, is shown the plot between bpp vs PSNR of color space RGB and YCbCr individually. Fig. 5.9(a) shows the original medical image, in Figures 5.9(b), 5.9(c) and 5.9(d) are shown plot of RGB color space and in Fig. 5.9(e), 5.9(f) and 5.9(g) show plot of YCbCr color space. From Fig. 5.9(f) and 5.9(g), PSNR is constant, so the almost signal energy of the new transformed YCbCr image is contained in the Y plane. Consequently, we can achieve high compression ratio in the Cb and Cr without losses in quality of compressed image when returned to the original RGB space.