Lossless Medical Image Compression using Predictive Coding and Integer Wavelet Transform based on Minimum Entropy Criteria

Komal Gupta, Ram Lautan Verma, Md. Sanawer Alam

Abstract

Lossless image compression has one of its important application in the field of medical images. Enormous amount of data is created by the information present in the medical images either in multidimensional or multiresolution form. Efficient storage, transmission, management and retrieval of the voluminous data produced by the medical images has nowadays become increasingly complex. Solution to the complex problem lies in the lossless compression of the medical data. Medical data is compressed in such a way so that the diagnostics capabilities are not compromised or no medical information is lost. This paper proposes a hybrid technique for lossless medical image compression that combines integer wavelet transforms and predictive coding to enhance the performance of lossless compression. Here we will first apply the integer wavelet transform and then predictive coding to each subband of the image obtained as an output to lifting scheme. Measures such as entropy, scaled entropy and compression ratio are used to evaluate the performance of the proposed technique. In this paper, for lossless compression of the grey-scale medical images a hybrid scheme is proposed. The scheme combines the integer wavelet transforms (IWT) and predictive coding. Here we will apply first the predictive coding and then integer wavelet transform to obtain subbands of the image as an output of the lifting scheme. This thesis presents an adaptive lifting scheme, which performs integer-to-integer wavelet transform, for lossless image compression. Our compression method is proposed to offer higher compression ratio and minimum entropy when applied to different test images.

Keywords- lossless compression, integer wavelet transform, subband coding, predictive coding, medical image, entropy.

1. INTRODUCTION

Compression offers a means to reduce the cost of storage and increase the speed of transmission, thus medical images have attained lot of attention towards compression. These images are very large in size and require lot of storage space. Image compression can be lossless and lossy, depending on whether all the information is retained or some of it is discarded during the compression process. In lossless compression, the recovered data is identical to the original, whereas in the case of lossy compression the recovered data is a close replica of the original with minimal loss of data. Lossy compression is used for signals like speech, natural images, etc., where as the lossless compression can be used for text and medical type images.

There has been a lot of research going on in lossless data compression. The most common lossless compression algorithms are run-length encoding, LZW, DEFLATE, JPEG, JPEG 2000, JPEG-LS, LOCO-I etc. Lempel–Ziv–Welch is a lossless data compression[18]

In order to provide a reliable and efficient means for storing and managing medical data computer based archiving systems such as Picture Archiving and Communication Systems (PACS) and Digital-Imaging and Communications in Medicine (DICOM) standards were developed. Health Level Seven (HL7) standards are widely used for exchange of textual information in healthcare information systems. With the explosion in the number of images acquired for diagnostic purposes, the importance of compression has become invaluable in developing standards for maintaining and protecting medical images and health records.

There has been a lot of research going on in lossless data compression. The most common lossless compression algorithms are run-length encoding, LZW, DEFLATE, JPEG, JPEG 2000, JPEG-LS, LOCO-I etc. Lempel–Ziv–Welch is a lossless data compression algorithm which can be used to compress images. The performance of LZW can be enhanced by introducing three methods. The first two methods eliminate the frequent flushing of dictionary, thus lowering processing time and the third method improves the compression ratio by reducing number of bits transferred over the communication channel. JPEG is most commonly used lossy compression technique for photographic images which can be converted into lossless by performing integer reversible transform. Lossless compression in JPEG [7] is achieved by performing integer reversible DCT (RDCT) instead of the floating point DCT used in original JPEG on each block of the image later using lossless quantization. Lossless JPEG does not allow flexibility of the code stream, to overcome this JPEG 2000[1-2] has been proposed. This technique performs lossless compression based on an integer wavelet filter called biorthogonal 3/5. JPEG 2000”s lossless mode runs really slow and often has less compression ratios on artificial
and compound images. To overcome this drawback JPEG-LS [6] has been proposed. This is a simple and efficient baseline algorithm containing two distinct stages called modeling and encoding. This technique is a standard evolved after successive refinements as shown in articles [3], [4], and [5]. JPEG-LS algorithm is more scalable than JPEG and JPEG 2000.

2. LOSSLESS IMAGE COMPRESSION

Many image compression algorithms use some form of transform coding. Figure 5 shows a block diagram of encoder and decoder using transform coding. The first step is to obtain a mathematical transformation to the image pixels in order to reduce the correlation between the pixels. The result of the transform is known as the transform coefficients. After this step, in lossy compression, an explicit quantizer may be used, or an implicit quantizer such as the truncation of the bitstream may be used. The source of the data loss in image compression is the quantizer. Thus, in the lossless compression case, the quantizer is not used. The third step is coefficient coding, which means that the transform coefficients are reorganized in order to exploit properties of the transform coefficients and obtain new symbols to be encoded at the fourth step. For example, the transform coefficients can be considered as a collection of quad-trees or zerotrees [8] [9] and or treated in a bit plane fashion, so as to provide scalability to the compressed bitstream. The symbols from the coefficient coding are losslessly compressed at the entropy coding step. Entropy coding can be any method capable of compressing a sequence of symbols, such as Huffman coding [10], arithmetic coding [11] and Golomb coding [12].

3. LOSSLESS COMPRESSION CODING TECHNIQUES

In this section different coding techniques used to achieve lossless compression are discussed. The primary encoding algorithms used to produce bit sequences are entropy coding techniques of which the most efficient are Huffman coding (also used by DEFLATE) and arithmetic coding. We also go over lossless predictive coding technique.

3.1 Entropy coding

Entropy measures the amount of information present in the data or the degree of randomness of the data. After the data has been quantized into a finite set of values it can be encoded using an entropy coder to achieve additional compression using probabilities of occurrence of data. This technique reduces the statistical redundancy. The entropy coder encodes the given set of symbols with the minimum number of bits required to represent them. It is a variable length coding which means that it assigns different number of bits to different gray levels. If the probability of occurrence is more, then fewer bits/sample will be assigned.

\[ H = - \sum_{k=1}^{M} P_k \log_2 P_k = \sum_{k=1}^{M} P_k \log_2 \left( \frac{1}{P_k} \right) \]

In the least random case it takes only one value where

\[ H = 0 \]

Most random case:

\[ H = \log_2 M \]

The average number of bits per pixel needed with Huffman coding is given by

\[ R = \sum_{k=1}^{M} P_k N_k \]

Where \( P_k \) represent the probabilities of the symbols and \( N_k \) represent the number of bits per the code generated. Coding efficiency (\( \eta \)) can also be calculated using \( H \) and \( R \) generated earlier

\[ \eta = \frac{H}{R} \times 100 \]
3.2. Huffman Coding

Huffman coding is an entropy coding algorithm which is used in lossless compression. In this technique the two smallest probabilities are combined or added to form a new set of probabilities. This uses a variable length code table which is based on the estimated probability of occurrence for each possible value of the source symbol. This is developed by David A. Huffman. In Huffman coding each symbol is represented in a specific method which expresses the most common characters with fewer strings than used for any other character. It is not optimal when the symbol-by-symbol restriction is dropped, or when the probability mass functions are unknown, not identically distributed, or not independent. The basic technique involves creating a binary tree of nodes which can be finally stored as an array. This size depends on the number of symbols which have given probabilities. Now the lowest two probabilities will be added and one probability will be represented by „0” and the other probability which is added will be assigned a „1”. This process is repeated until all the additions are completed leaving a sum of one. The simplest construction algorithm uses a priority queue where the node with lowest probability is given highest priority. The performance of the method is calculated using entropy.

4. INTEGER WAVELET TRANSFORM

The wavelet transform generally produces floating-point coefficients. Although the original pixels can be reconstructed by perfect reconstruction filters without any loss in principle, the use of finite-precision arithmetic and quantization prevents perfect reconstruction. The reversible IWT (Integer Wavelet Transform), which maps integer pixels to integer coefficients and can reconstruct the original pixels without any loss, can be used for lossless compression [13] [14] [15] [16]. One approach used to construct the IWT is the use of the lifting scheme (LS) described by Calderbank et al. The IWT construction using lifting is done in the spatial domain, contrary to the frequency domain implementation of a traditional wavelet transform [16] [17].

Wavelet transforms have proven extremely effective for transform-based image compression. Since many of the wavelet transform coefficients for a typical image tend to be very small or zero, these coefficients can be easily coded. Thus, wavelet transforms are a useful tool for image compression.

The main advantage of wavelet transforms over other more traditional decomposition methods (like the DFT and DCT) is that the basis functions associated with a wavelet decomposition typically have both long and short support. The basis functions with long support are effective for representing slow variations in an image while the basis functions with short support can efficiently represent sharp transitions (i.e., edges).

5. LIFTING SCHEME

The simplest lifting scheme is the lazy wavelet transform, where the input signal is first split into even and odd indexed samples.

\[ (\text{odd}_{j-1}, \text{even}_{j-1}) = \text{Split}(s_j) \]

The samples are correlated, so it is possible to predict odd samples from even samples which in the case of Haar transform are even values themselves. The difference between the actual odd samples and the prediction becomes the wavelet coefficients. The operation of obtaining the differences from the prediction is called the lifting step. The update step follows the prediction step, where the even values are updated from the input even samples and the updated odd samples. They become the scaling coefficients which will be passed on to the next stage of transform. This is the second lifting step.

\[ d_{j-1} = \text{odd}_{j-1} - P(\text{even}_{j-1}) \]
\[ s_{j-1} = \text{even}_{j-1} + U(d_{j-1}) \]

Finally the odd elements are replaced by the difference and the even elements by the averages. The computations in the lifting scheme are done in place which saves lot of memory and computation time. The lifting scheme provides integer coefficients and so it is exactly reversible. The total number of coefficients before and after the transform remains the same.

![Fig. 2: Forward Lifting Scheme](image1)

![Fig. 3: Inverse Lifting Scheme](image2)
The inverse transform gets back the original signal by exactly reversing the operations of the forward transform with a merge operation in place of a split operation. The number of samples in the input signal must be a power of two, and these samples are reduced by half in each succeeding step until the last step which produces one sample.

\[
\begin{align*}
\text{Even}_{j-1} &= s_{j-1} - \text{U}(d_{j-1}) \\
\text{Odd}_{j-1} &= d_{j-1} + \text{P} (\text{Even}_{j-1}) \\
\text{Finally, } s_j &= \text{Merge} (\text{Even}_{j-1}, \text{Odd}_{j-1})
\end{align*}
\]

The Haar wavelet transform uses predict and update operations of order one. Using different predict and update operations of higher order, many other wavelet transforms can be built using the lifting scheme.

\[
\begin{align*}
\text{Even}_{j-1} &= s_{j-1} - \text{U}(d_{j-1}) \\
\text{Odd}_{j-1} &= d_{j-1} + \text{P} (\text{Even}_{j-1}) \\
\text{Finally, } s_j &= \text{Merge} (\text{Even}_{j-1}, \text{Odd}_{j-1})
\end{align*}
\]

Basic steps involved in the decomposition are illustrated in Fig 4[18]. Firstly the image/signal is sent through a low pass and band pass filter simultaneously (predict and update in case of lifting) and down sampled by a factor of 2. The process is repeated and the final four outputs are combined to from the transformed image as shown in Fig 3.8.

The transformed image shows different sub bands of which the first sub band is called LL which represents the low resolution version of the image, the second sub band is called LH which represents the horizontal fluctuations, the third band is called the HL which represents the vertical fluctuations, and the fourth sub band is called the HH which represents the diagonal fluctuations.

**6. Introduction to Predictive Coding**

The prediction technique computes the weighted differences between neighboring pixel values to estimate the predicted pixel value. The prediction error is decomposed by a one-level integer wavelet transform to improve the prediction. Let \( f(n) \) be the original sample then the difference \( d(n) \) will be given by

\[ d(n) = f(n) - f(n-1). \]

Fig 6 shows that it is easier to encode the difference rather than encoding the original sample because of less dynamic range.

**Fig. 4: Steps for Decomposition Using Lifting**

**Fig. 5: Input and Outputs of Lifting Scheme**

**Fig. 6: Original Histogram**

**Fig. 7: Histogram of the difference**
Generally, the second order predictor is used which is also called Finite Impulse Response (FIR) filter. The simplest predictor is the previous value, in this experiment the predicted value is sum of the previous two values with alpha and beta being the predictor coefficients.

\[ f(n) = \langle f(n-1) \rangle \]

In the process of predictive coding input image is passed through a predictor where it is predicted with its two previous values.

\[ \hat{f}(n) = \alpha \cdot f(n-1) + \beta \cdot f(n-2) \]

\( \hat{f}(n) \) is the rounded output of the predictor, \( f(n-1) \) and \( f(n-2) \) are the previous values, \( \alpha \) and \( \beta \) are the coefficients of the second order predictor ranging from 0 to 1. The output of the predictor is rounded and is subtracted from the original input. This difference is given by

\[ d(n) = f(n) - \hat{f}(n) \]

Now this difference is given as an input to the decoder part of the predictive coding technique. In the decoding part the difference is added with the \( f^*(n) \) to give the original data

\[ f(n) = d(n) + \hat{f}(n) \]

### 7. IMPLEMENTATION AND EXPERIMENTAL RESULTS

In this report the Integer Wavelet Transform (IWT) and the Predictive Coding Techniques are used to perform lossless image compression. The performance of the proposed techniques is calculated by finding the Entropy and scaled entropy of the compressed image. The performance is also measured using compression ratio which is given by the ratio of the bits in the original uncompressed data to the number of bits in the compressed data.

#### 7.1. Procedure

The procedure of the implementation involves two methods of performing compression on the test images. In the first method the predictive coding technique is applied first followed by the integer wavelet transform. The second method involves reduction of the filter coefficients by a factor of 3/2 and performing predictive coding followed by integer wavelet transform. All these methods use Haar filter in the lifting scheme and the filter coefficients are given by

\[ h_1 = [-1 \ 9 \ 9 \ 1]/16; \]
\[ h_2 = [0 \ 0 \ 1 \ 1]/(-4); \]

Where \( h_1 \) are the prediction filter coefficients and \( h_2 \) are the update filter coefficients in the lifting scheme.

The reduced filter coefficients are given by

\[ h_1 = [-1 \ 9 \ 9 \ 1]/(16*1.5); \]
\[ h_2 = [0 \ 0 \ 1 \ 1]/(-4*1.5); \]

The implementation of all the two methods for nasal fracture image is also shown in the results.

---

**Fig. 8:** Predictive Encoder

**Fig. 9:** Predictive Decoder

---

**Fig. 10:** Block Diagram for Predictive Coding Followed by IWT
In this method predictive coding is applied on image first, this converts the image into $f^*(n)$. Now on this output integer wavelet transform is applied which divides the image into four subbands $ss$, $sd$, $ds$, $dd$. The reconstruction process involves applying the inverse integer wavelet transform on the transformed image followed by applying predictive decoding on the output of the inverse transform $F(n)$. The reconstructed image is represented by $z$.

### 7.1.1. Implementation using method 1

![Original image of nasal fracture](image1)

**Fig 11:** Original image of nasal fracture [19]

![Predictive Encoding](image2)

**Fig 12:** Predictive Encoding

![Image obtained after subband coding](image3)

**Fig 13:** Image obtained after subband coding

![Image obtained after inverse IWT](image4)

**Fig 14:** Image obtained after inverse IWT

![Predictive Decoding](image5)

**Fig 15:** Predictive Decoding

![Difference of Original and reconstructed image](image6)

**Fig 16:** Difference of Original and reconstructed image

### 7.1.2. Implementation using method 2

![Original image of nasal fracture](image7)

**Fig 17:** Original image of nasal fracture

![Predictive Encoding](image8)

**Fig 18:** Predictive Encoding

![Image obtained after subband coding](image9)

**Fig 19:** Image obtained after subband coding

![Image obtained after inverse IWT](image10)

**Fig 20:** Image obtained after inverse IWT

![Predictive Decoding](image11)

**Fig 21:** Predictive Decoding

![Difference of Original and reconstructed image](image12)

**Fig 22:** Difference of Original and reconstructed image

The other filters used are Daubechies 2 and Daubechies 3 in the lifting scheme. Daubechies 2 is an orthogonal wavelet with two vanishing moments and Daubechies 3 filter is also an orthogonal wavelet with 3 vanishing moments.
7.1.3. Using Daubechies 2 filter

The coefficients of Daubechies 2 are given by

\[ h_1 = \left[ \frac{1 + \sqrt{3}}{4\sqrt{2}}, \frac{3 + \sqrt{3}}{4\sqrt{2}}, \frac{3 - \sqrt{3}}{4\sqrt{2}}, \frac{1 - \sqrt{3}}{4\sqrt{2}} \right] \]
\[ h_2 = \left[ -\frac{1 - \sqrt{3}}{4\sqrt{2}}, \frac{3 - \sqrt{3}}{4\sqrt{2}}, \frac{3 + \sqrt{3}}{4\sqrt{2}}, \frac{1 + \sqrt{3}}{4\sqrt{2}} \right] \]

7.1.4. Using Daubechies 3 filter

The coefficients of Daubechies 3 are given by

\[ h_1 = [0.3327, 0.8069, 0.4600, -0.1350, -0.085, 0.0352] \]
\[ h_2 = [-0.0352, -0.085, 0.1350, 0.4600, -0.8069, 0.3327] \]

Now different tables are drawn based on the above mentioned four methods.

**Table 1:** Scaled Entropy of different test images

<table>
<thead>
<tr>
<th>IMAGES</th>
<th>ORIGINAL ENTROPY</th>
<th>METHOD 1 SCALED ENTROPY</th>
<th>METHOD 2 SCALED ENTROPY</th>
<th>DAUBECHIES 2 SCALED ENTROPY</th>
<th>DAUBECHIES 3 SCALED ENTROPY</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iris of Eye</td>
<td>0.9989</td>
<td>0.7639</td>
<td>0.7346</td>
<td>0.9533</td>
<td>0.7549</td>
</tr>
<tr>
<td>Mri of Ankle</td>
<td>0.9327</td>
<td>0.6132</td>
<td>0.8439</td>
<td>0.8040</td>
<td>0.7665</td>
</tr>
<tr>
<td>Mri of Brain</td>
<td>0.8939</td>
<td>0.8141</td>
<td>0.9889</td>
<td>0.8668</td>
<td>0.8576</td>
</tr>
<tr>
<td>Nasal fracture</td>
<td>0.8395</td>
<td>0.6892</td>
<td>0.9934</td>
<td>0.7452</td>
<td>0.7687</td>
</tr>
</tbody>
</table>

**Table 2:** Compression ratio of different test images

<table>
<thead>
<tr>
<th>IMAGES</th>
<th>COMPRESSION RATIO</th>
</tr>
</thead>
<tbody>
<tr>
<td>Iris of Eye</td>
<td>1.508</td>
</tr>
<tr>
<td>Mri of Ankle</td>
<td>1.097</td>
</tr>
<tr>
<td>Mri of Brain</td>
<td>1.061</td>
</tr>
<tr>
<td>Nasal fracture</td>
<td>1.098</td>
</tr>
</tbody>
</table>

**Table 3:** Scaled entropy at different combinations of alpha and beta

<table>
<thead>
<tr>
<th>S. No.</th>
<th>ALPHA</th>
<th>BETA</th>
<th>ORIGINAL ENTROPY</th>
<th>METHOD 1 SCALED ENTROPY</th>
<th>METHOD 2 SCALED ENTROPY</th>
<th>DAUBECHIES 2 SCALED ENTROPY</th>
<th>DAUBECHIES 3 SCALED ENTROPY</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.1</td>
<td>0.1</td>
<td>0.8395</td>
<td>0.6572</td>
<td>0.9892</td>
<td>0.7088</td>
<td>0.7356</td>
</tr>
<tr>
<td>2</td>
<td>0.3</td>
<td>0.3</td>
<td>0.8395</td>
<td>0.6892</td>
<td>0.9934</td>
<td>0.7452</td>
<td>0.7687</td>
</tr>
<tr>
<td>3</td>
<td>0.5</td>
<td>0.5</td>
<td>0.8395</td>
<td>0.8162</td>
<td>0.7797</td>
<td>0.8371</td>
<td>0.8805</td>
</tr>
<tr>
<td>4</td>
<td>0.7</td>
<td>0.7</td>
<td>0.8395</td>
<td>0.7433</td>
<td>0.4170</td>
<td>0.7832</td>
<td>0.7918</td>
</tr>
<tr>
<td>5</td>
<td>0.9</td>
<td>0.9</td>
<td>0.8395</td>
<td>0.7158</td>
<td>0.3690</td>
<td>0.7494</td>
<td>0.7664</td>
</tr>
<tr>
<td>6</td>
<td>0.1</td>
<td>0.01</td>
<td>0.8395</td>
<td>0.6508</td>
<td>0.9881</td>
<td>0.7034</td>
<td>0.7290</td>
</tr>
<tr>
<td>7</td>
<td>0.01</td>
<td>0.1</td>
<td>0.8395</td>
<td>0.6551</td>
<td>0.9886</td>
<td>0.7068</td>
<td>0.7330</td>
</tr>
<tr>
<td>8</td>
<td>0.01</td>
<td>0.01</td>
<td>0.8395</td>
<td>0.6478</td>
<td>0.9878</td>
<td>0.7025</td>
<td>0.7278</td>
</tr>
</tbody>
</table>
Fig 23: Graphs showing Scaled Entropy at different combinations of alpha and beta

Fig 24: Bargraphs showing Scaled Entropy of different test images

8. Conclusion
In this paper, a lossless image compression method is presented first using predictive coding and then Integer Wavelet Transform for different medical test images. The compression ratio is calculated for different test images such as iris of eye [19], mri of ankle [19], mri of brain [19], nasal fracture [19]. Then scaled entropy of different test images are calculated for different test images using method 1, method 2, Daubechies 2, Daubechies 3 filter coefficients. The first method employs predictive coding technique on the image first followed by the integer wavelet transform giving a transformed output, which is then passed through the reverse techniques to get back the original image. The second method performed on the image involve reduction of the Haar filter coefficients by a factor of 3/2 and both methods are implemented on all the images. The other filters used are Daubechies 2 and Daubechies 3 in the lifting scheme. Daubechies 2 is an orthogonal wavelet with two vanishing moments and Daubechies 3 filter is also an orthogonal wavelet with 3 vanishing moments. The high compression efficiency is seen to be achieved by not only a combination PREDICTIVE CODING and IWT, but also by optimisation in the design of the predictor and the choice of the transform. Using 4 levels of IWT, performance of these four methods is purely lossless. Entropy and scaled entropy are used to calculate the performance of the system, which calculates the number of bits per pixel. A lower entropy and scaled entropy indicate higher performance of the system. The analysis of experimental results has given some conclusions. Predictor coefficients alpha and beta value can lie between 0 and 1 and its choice is critical, so different combinations of these coefficients are tested. The best combination is (0.01, 0.01) as seen by method 1 technique and Daubechies 3 filter that has been highlighted in the table 3 and it gives the least entropy.

Acknowledgement
The author wants to thanks her guide Mr. R.L. Verma sir (Asstt. Prof.) of A.I.E.T, Mr. Md. Sanawer Alam sir (HOD) of A.I.E.T. and Krishna Mohan Pandey sir for their collaboraton that has made this work possible.

References


