ISSN No. 2278-3091

Volume 1, No.3, July – August 2012
International Journal of Advanced Trends in Computer Science and Engineering
Available Online at http://warse.org/pdfs/ijatcse02132012.pdf

High Throughput Image Compression Using SPIHT With Huffman Encoding

[1]S.Vijayadheeswar Reddy , [2]I.V.G.Manohar , [3]Dr.Rajeyyagari Sivaram
[1&2] Nalanda Institute of Engg. & Tech. Guntur, [3]Amara Institute of Engg. & Tech.
dr.r.sivaram@gmail.com

ABSTRACT

Digital Imaging and Communications in Medicine (DICOM) is a standard used for distribution and viewing of medical images from different modalities. In this paper, we propose a method for compressing DICOM images based on the well-known Set Partitioning in Hierarchical Trees (SPIHT) which possess the progressive transmission capabilities useful for telemedicine applications. This is in sharp contrast to Joint Photographic Experts Group (JPEG) where the images have to be first compressed to the required level and only then can be transmitted. In our method, the header and image data is separated from the DICOM image where the header is transmitted first with necessary modifications. The image data is then compressed using SPIHT and transmitted. It is found from the experiment that our method gives better results in terms of compression and time.

Keywords: DICOM, Image Compression, JPEG, SPIHT, Wavelets, *Encoding, DWT, SPIHT, Huffman*

1. INTRODUCTION

In recent years, wavelet transform [1] [2] as a branch of mathematics developed Rapidly, which has a good localization property[3] in the time domain and frequency domain, can analyze the details of any scale and frequency. So, it superior to Fourier and DCT. It has been widely applied and developed in image processing and compression.

EZW stands for ‘Embedded Zero tree Wavelet’, which is abbreviated from the title of Jerome Shapiro’s 1993 article[4],”Embedded Image Coding Using Zero trees of Wavelet Coefficients”. EZW is a simple and effective image compression algorithm, its output bit-stream ordered by importance. Encoding was able to end at any location, so it allowed achieving accurate rate or distortion. This algorithm does not need to train and require pre-stored codebook. In a word, it does not require any prior knowledge of original image. More improvements over EZW are achieved by SPIHT, by Amir Said and William Pearlman, in 1996 article, "Set Partitioning In Hierarchical Trees" [5]. In this method, more (wide-sense) zero trees are efficiently found and represented by separating the tree root from the tree, so, making compression more efficient.

Experiments are shown that the image through the wavelet transform, the wavelet coefficients’ value in high frequency region are generally small [6], so it will appear seriate "0" situation in quantify. SPIHT does not adopt a special method to treat with it, but direct output. In this paper, focus on this point, propose a simple and effective method combined with Huffman encode for further compression. A large number of experimental results are shown that this method saves a lot of bits in transmission, further enhanced the compression performance.

2. THE SPIHT ALGORITHM

One of the most efficient algorithms in the area of image compression is the Set Partitioning in Hierarchical Trees (SPIHT). In essence it uses a sub-band coder, to produce a pyramid structure where an image is decomposed sequentially by applying power complementary low pass and high pass filters and then decimating the resulting images. These are one-dimensional filters that are applied in cascade (row then column) to an image whereby creating a four-way decomposition: LL (low-pass then another low pass), LH (low pass then high pass), HL (high and low pass) and finally HH (high pass then another high pass). The resulting LL version is again four-way decomposed, as shown in Figure 1. This process is repeated until the top of the pyramid is reached.

\

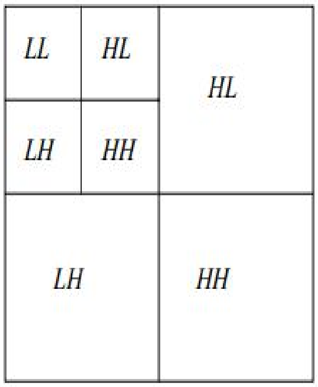


Figure 1: Image decomposition using wavelets

There exists a spatial relationship among the 1-3,6 coefficients at different levels and frequency sub-bands in the pyramid structure. A wavelet coefficient at location (i,j) in the pyramid representation has four direct descendants (off-springs) at locations:

$$O(i,j)=\{(2i,2j),(2i,2j+1),(2i+1,2j),(2i+1,2j+1)\} \qquad (1)$$

and each of them recursively maintains a spatial similarity to its corresponding four offspring. This pyramid structure is commonly known as spatial orientation tree. For example, Figure 2 shows the similarity among sub-bands within levels in the wavelet space 6 . If a given coefficient at location (i,j) is significant in magnitude then some of its descendants will also probably be significant in magnitude. The SPIHT algorithm takes advantage of the spatial similarity present in the wavelet space to optimally find the location of the wavelet coefficient that are significant by means of a binary search algorithm.

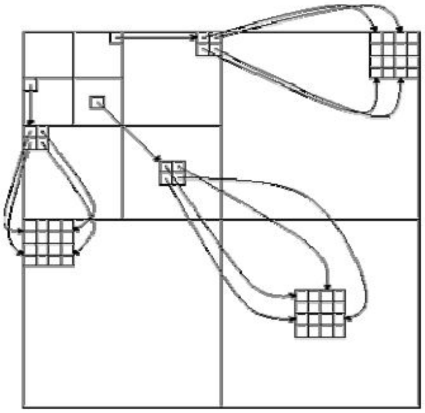
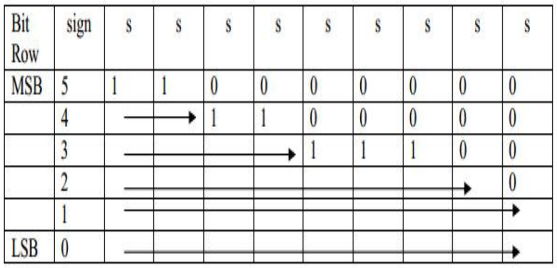


Figure 2: Off-spring dependencies in the pyramid structure[6]

The SPIHT algorithm sends the top coefficients in the pyramid structure using a progressive transmission scheme. This scheme is a method that allows obtaining a high quality version of the original image from the minimal amount of transmitted data. As illustrated in Figure 3, the pyramid wavelet coefficients are ordered by magnitude and then the most significant bits are transmitted first, followed by the next bit plane and so on until the lowest bit plane is reached. It has been shown that progressive transmission can significantly reduced the Mean Square Error (MSE) distortion for every bit-plane sent To take advantage of the spatial relationship among the coefficients at different levels and frequency bands, the SPIHT coder algorithm orders the wavelets coefficient according to the significance test defined as:

$$\max_{(i,j)\in\tau_m} \left|C_{i,j}\right| \ge 2^n \qquad (2)$$

where C is the wavelet coefficient at the nth bit plane, at location (i,j)of the i,j τ m subset of pixels, representing a parent node and its descendants. If the result of the significance test is yes an S flag is set to 1 indicating that a particular test is significant. If the answer is no, then the S flag is set to 0, indicating that the particular coefficient is insignificant. This is represented by equation (3).



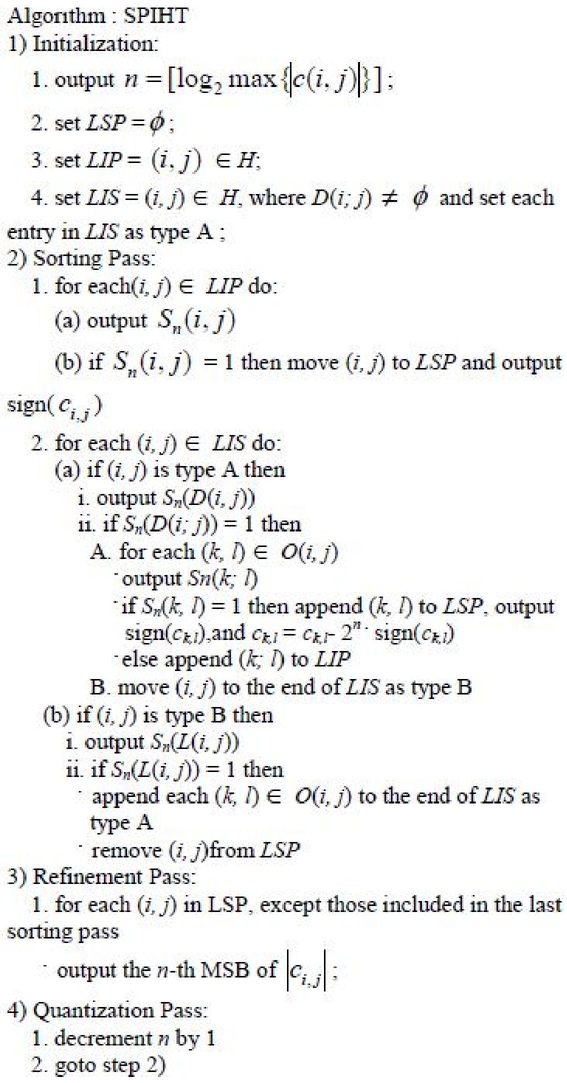
Bit Row	sign	s	s	s	s	s	s	s	s	s
MSB	5	1	1	0	0	0	0	0	0	0
	4	→	→	1	1	0	0	0	0	0
	3	→	→	→	→	1	1	1	0	0
	2	→	→	→	→	→	→	→	→	0
	1	→	→	→	→	→	→	→	→	→
LSB	0	→	→	→	→	→	→	→	→	→

Figure 3: Bit-plane ordering and transmission scheme

$$S_n(\tau)=\begin{cases}1, & \max\limits_{(i,j)\in\tau}\left|C_{i,j}\right|\ge 2^n\\ 0, & \text{otherwise}\end{cases} \qquad (3)$$

Wavelets coefficients which are not significant at the nth bit-plane level may be significant at (n-1)th bit-plane or lower. This information is arranged, according to its significance, in three separate lists: list of insignificant sets (LIS), the list of insignificant

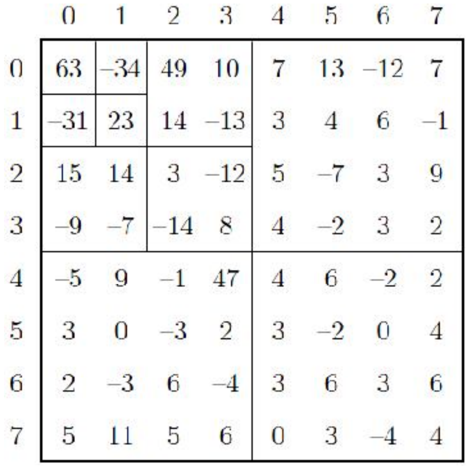
Pixels (LIP) and the list of significant pixels (LSP). In the decoder, the SPIHT algorithm replicates the same number of lists. It uses the basic principle that if the execution path of any algorithm is defined by the results on its branching points, and if the encoder and decoder have the same sorting algorithm then the decoder can recover the ordering information easily



```
Algorithm : SPIHT
1) Initialization:
   1. output $n = [\log_2 \max\{|c(i,j)|\}]$;
   2. set $LSP = \phi$;
   3. set $LIP = (i,j) \in H$;
   4. set $LIS = (i,j) \in H$, where $D(i;j) \neq \phi$ and set each
entry in $LIS$ as type A ;
2) Sorting Pass:
   1. for each$(i,j) \in LIP$ do:
      (a) output $S_n(i,j)$
      (b) if $S_n(i,j) = 1$ then move $(i,j)$ to $LSP$ and output
sign($c_{i,j}$)
   2. for each $(i,j) \in LIS$ do:
      (a) if $(i,j)$ is type A then
         i. output $S_n(D(i,j))$
         ii. if $S_n(D(i;j)) = 1$ then
           A. for each $(k,l) \in O(i,j)$
              · output $Sn(k;l)$
              · if $S_n(k,l) = 1$ then append $(k,l)$ to $LSP$, output
                sign($c_{k,l}$),and $c_{k,l} = c_{k,l} - 2^n \cdot$ sign($c_{k,l}$)
              · else append $(k;l)$ to $LIP$
           B. move $(i,j)$ to the end of $LIS$ as type B
     (b) if $(i,j)$ is type B then
        i. output $S_n(L(i,j))$
        ii. if $S_n(L(i,j)) = 1$ then
          · append each $(k,l) \in O(i,j)$ to the end of $LIS$ as
          type A
          · remove $(i,j)$from $LSP$
3) Refinement Pass:
   1. for each $(i,j)$ in LSP, except those included in the last
sorting pass
      · output the $n$-th MSB of $|c_{i,j}|$;
4) Quantization Pass:
   1. decrement $n$ by 1
   2. goto step 2)
```

3. ANALYSE OF SPIHT ALGORITHM

Here a concrete example to analyze the output binary stream of SPIHT encoding is given. The following is 3-level wavelet decomposition coefficients of SPIHT encoding:



	0	1	2	3	4	5	6	7
0	63	–34	49	10	7	13	–12	7
1	–31	23	14	–13	3	4	6	–1
2	15	14	3	–12	5	–7	3	9
3	–9	–7	–14	8	4	–2	3	2
4	–5	9	–1	47	4	6	–2	2
5	3	0	–3	2	3	–2	0	4
6	2	–3	6	–4	3	6	3	6
7	5	11	5	6	0	3	–4	4

$n = [\log_2 \max\{|c(i,j)|\}] = 5,$ so, The initial threshold value:T0=25, for T0, the output binary stream: 1110001110 0010000001010110000, 29 bits in all. By the SPIHT encoding results, we can see that the output bit stream with a large number of seriate "0" situation, and along with the gradual deepening of quantification, the situation will become much more severity, so there will have a great of redundancy when we direct output

4. MODIFIED SPIHT ALGORITHM

For the output bit stream of SPIHT encoding with a large number of seriate "0" situation ,we obtain a conclusion by a lot of statistical analysis: ‘000’ appears with the greatest probability value , usually will be about 1/4.Therefore, divide the binary output stream of SPIHT every 3 bits as a group, every group recorded as a symbol, a total of eight kinds of symbols, statistical probability that they appear, and then encoded using variable-length encoding naturally reached the further compressed, in this paper, variable length encoding is Huffman encoding. Using the output bit stream of above example to introduce the new encoding method process.

1) First, divide the binary output stream every 3 bits as a group: 111 000 111 000 100 000 010 101 100 00. In this process, there will be remain 0,1,2 bits cannot participate. So, in order to unity, in the head of the output bit stream of Huffman encoding cost two bits to record the number of bits that do not participate in group and those remainder bits direct output in end. Figure 2 is shown the output bit stream structure of Huffman encoding

2) The emergence of statistical probability of each symbol grouping results are as follows:

P(‘000’)= 0.3333 P(‘001’)= 0

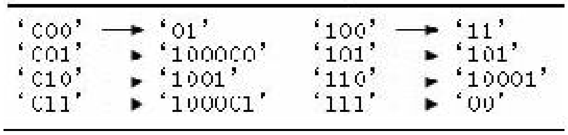
P(‘010’)= 0.1111 P(‘011’)= 0

P('100')= 0.2222 P('101')= 0.1111

P('110')= 0 P('111')= 0.2222

3) According to the probability of the above results, using Huffman encoding , then obtain code word book, as follow table1

Table 1 Code word comparison table



'000'	→	'01'	'100'	→	'11'
'001'	►	'100000'	'101'	►	'101'
'010'	►	'1001'	'110'	►	'10001'
'011'	►	'100001'	'111'	►	'00'

Through the above code book we can get the corresponding output stream: 10 00 01 00 01 11 01 1001 101 11 00, a total of 25 bits, The '10' in the head is binary of remainder bits'number.The last two bits '00' are the result of directly outputting remainder bits. Compared with the original bit stream save four bits. Decoding is inverse process of the above-mentioned process.

5. RESULTS

The SPIHT algorithm was implemented using MATLAB. Our preliminary results are as follows, Figure 4 shows the original Lena image, Figure 5 illustrates the wavelet pyramid tree Treated by using the bi-orthogonal wavelet bior4.4 in MATLAB [2]) and Figure 5 shows the recovered image using up to 5th bit plane (from the maximum of 13 bit-planes). The recovered image is visually very close to the original mage. If all the bit planes are used then the original image is recovered completely (up to rounding errors). Although the MATLAB version of the SPIHT runs slow no attempt was done to optimize the code as for instance in8,9. The authors' aim was to have an implementation ready for students to experiment with. Students' familiarity with MATLAB and accessible tools in it will allow them to easily modify the code and reduce development time. For instance one method is already explored in10 by modifying the SPIHT algorithm using Lossy/Lossless region of interest (ROI) coding. In a similar fashion our students can modify the algorithm to include different methods of ROI coding.

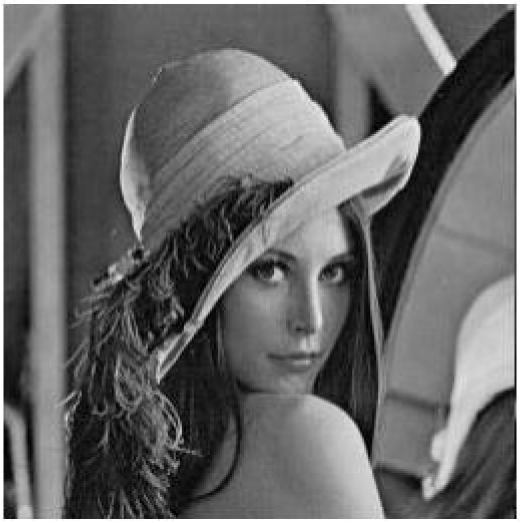


Figure 4. Original Image

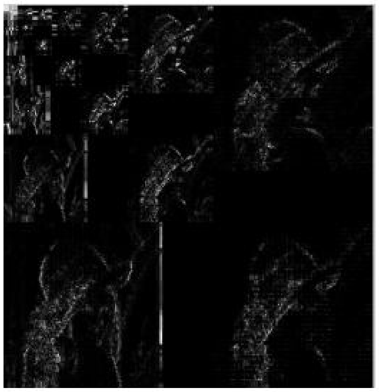


Figure 5: Pyramid tree generated by the four-way decomposition

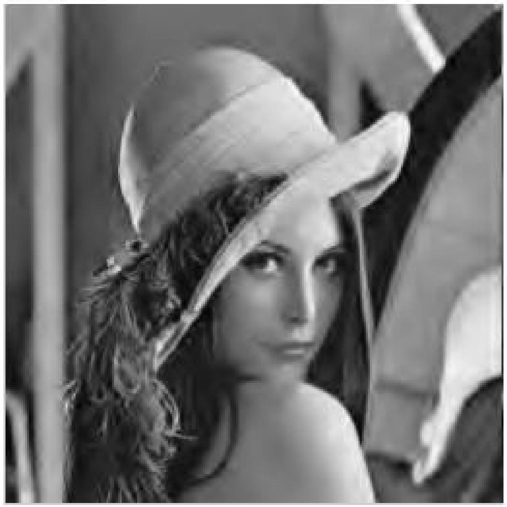


Figure 6: Recovered image at 5th level

6. CONCLUSION

Proposing a simple and effective method combined with Huffman encoding for further compression in this paper that saves a lot og bits in the image data transmission. There are very wide range of practical value for today that have large number of image data's to be transmitted.. We have implemented the SPIHT algorithm using MATLAB and its wavelet toolbox.

Students' familiarity with this platform is what guided us through this development. Further research includes the optimization of the code as in8, improvements for region of interest coding, and real-time implementation of the algorithm. Although the program runs slower than its C counterpart, it gives the flexibility of using stable and robust set of tools such as MATLAB's.

REFERENCES

1. A. Said and W. Pearlman. **A New, fast and Efficient Image Code Based on Set Partitioning in Hierarchical Trees**, IEEE Transactions on Circuits and Systems for Video technology, Vol. 6, No. 3, June 1996.

2. J.M. Shapiro. **Embedded image coding using zero-trees of wavelet coefficients**, IEEE Transactions Signal Processing, vol. 41, Dec. 1993.

3. D. Taubman. **High Performance scalable image compression with EBCOT**, IEEE Transactions on Image Processing, Vol. 9, July 2000.

4. I. Hontsch and L. Karan. **Locally adaptive perceptual image coding**, IEEE Transactions on Image Processing, Vol. 9, September 2000.

5. Misiti, Y, Misiti, G. Oppenheim and J.M. Poggi. **Wavelet Toolbox, For use with MATLAB**, The Mathworks Inc., Natick, MA.

6. V. Ruiz. **Bit-Plane Compression Using SPIHT**, http://www.ace.ual.es/~vruiz/IASTED-2000/html/node3.html

7. D. Hanselman and B. Littlefield. **Mastering MATLAB: A Comprehensive Tutorial and Reference**, Prentice Hall, Upper Saddle River, New Jersey, 2001.

8. Y. Sun, H. Zhang and G. Hu. **Real-Time Implementation of a New Low-Memory SPIHT Image Coding Algorithm Using DSP Chip**, IEEE Transactions on Image Processing, Vol. 11, No. 9, September 2002.

9 Rafael C. Gonzalez Richard and E. Woods. **Digital image processing**: second ed [M]. Beijing Publishing House of Electronics Industry 2002

10. Marc Antonni Michel Barlaud Pierre Mathieu et al. **Image coding using wavelet transform** [J]. IEEE Trans. Image Processing 1992 1(2), pp. 205-220.

11 Cheng Li-chi, Wang Hong-xia and Luo Yong. **Wavelet Theory and Applications**, Beijing: Science Press, 2004(Chinese).

12 J. M. Shapiro. **Embedded Image Coding using Zero Tree of Wavelets Coefficients** [J]. IEEE Trans. Signal Processing 1993 41(12), pp. 345-346.

13. Amir Said and William A.Pearlman. **A new fast and efficient image codec based on set partitioning in hierarchical trees** [J]. IEEE Transactions On Circuits and Systems for VideoTechnology 1996 6(3), pp.243-250.

14. Fan Qi-bin. **Wavelet analysis**, Wuhan: Wuhan University Press, 2008.