An Spiht Algorithm With Huffman Encoder For Image Compression And Quality Improvement Using Retinex Algorithm

A. Mallaiah, S. K. Shabbir, T. Subhashini

Abstract- Traditional image coding technology mainly uses the statistical redundancy between pixels to reach the goal of compressing. The research on wavelet transform image coding technology has made a rapid progress. Because of its high speed, low memory requirements and complete reversibility, digital wavelet transform (IWT) has been adopted by new image coding standard, JPEG 2000. The embedded zero tree wavelet (EZW) algorithms have obtained not bad effect in low bit-rate image compression. Set Partitioning in Hierarchical Trees (SPIHT) is an improved version of EZW and has become the general standard of EZW So, In this paper we are proposing DWT and SPIHT Algorithm with Huffman encoder for further compression and Retinex Algorithm to get enhanced quality improved image.

I. INTRODUCTION

SPIHT is computationally very fast and among the best image compression algorithms known today. According to statistic analysis of the output binary stream of SPIHT encoding, propose a simple and effective method combined with Huffman encode for further compression. Wavelet transform as a branch of mathematics developed rapidly, which has a good localization property 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'. "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. SPIHT stands for "Set Partitioning In Hierarchical Trees". 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. The image through the wavelet transform, the wavelet coefficients 'value in high frequency region are generally small, 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. Mallaiah, Associate Professor in ECE Department, Gudlavalleru Engineering College, Gudlavalleru, A.P

S. K. Shabbir, P.G. Student in ECE Department, Gudlavalleru Engineering College, Gudlavalleru, A.P.

T. Subhashini, Asst. Professor in ECE Department, Sri Vasavi Institute of Engineering & Technology, Pedana, A.P.

E-mail: malli797@gmail.com , sks20.12@gmail.com , subhashini.anagani@gmail.com

A large number of experimental results are shown that this method saves a lot of bits in transmission, further enhanced the compression performance and image quality improved at the time of image retrieval using Retinex Algorithm and for clearly visible.

II. SPIHT ALGORITHM

A. Description of the algorithm

Image data through the wavelet decomposition, the coefficient of the distribution is change into a tree. According to this feature, defining a data structure: spatial orientation tree. Four-level wavelet decomposition of the spatial orientation trees structure are shown in Figure1. Here we can see that each coefficient has four children except the 'red' marked coefficients in the LL sub band and the coefficients in the highest sub bands (LH1, HL1, and HH1). The following set of coordinates of coefficients is used to represent set partitioning method in SPIHT algorithm. The location of coefficient is noted by (i,j), where i and j indicate row and column indices, respectively.

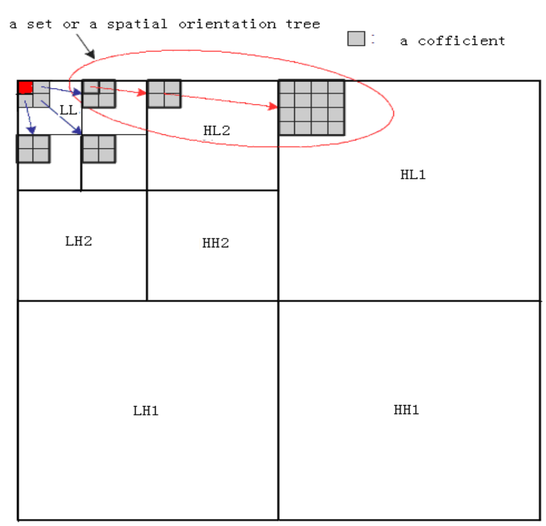

Figure1. Parent-child relationship in SPIHT

H: Roots of the all spatial orientation trees
O(i,j): Set of offspring of the coefficient (i,j),

O(i,j)={(2i,2j)(2i,2j+1)(2i+1,2j)(2i+1,2j+1)}, except (i,j) is in LL. When (i,j) is in LL sub band, O(i,j) is defined as:O(i,j)={(i,j+w_{LL}),(i+h_{LL},j),(i+$h_{LL,}$ j+w_{LL})}, where w_{LL} and h_{LL} are the width and height of the LL sub band, respectively. D(i,j): Set of all descendants of the coefficient (i,j),

L(i,j): D(i,j)-O(i,j)

A significance function $S_n(r')$ which decides the significance of the set of coordinates, r' , with respect to threshold 2^n is defined by:

$$S_n(r')=\begin{cases} 1, \text{ if } \max_{(i,j)\in r'} \{ |C_{i,j}| \} \geq 2^n \\ 0, \text{ else where} \end{cases}$$

Where $C_{i,j}$ is the wavelet coefficients. In the algorithm, three ordered lists are used to store the significance information during set partitioning. List of insignificant sets (LIS), list of insignificant pixels (LSP) are those three lists. Note that the term 'pixel' is actually indicating wavelet coefficient if the set partitioning algorithm is applied to a wavelet transformed image.

Algorithm: SPIHT

1) Initialization:

1. Output n=[$\log_2$max{│ C(i,j)│}];
2. Set LSP= Ø ;
3. Set LIP= (i,j) ∈ H;
4. Set LIS=(i,j) ∈ H, where D(i,j) ≠ Ø and set each entry in LIS as type A;

2)Sorting Pass:

2.1) for each entry (i, j) in the LIP do:

output Sn(i, j),

if Sn(i, j)=1 then move (i, j) to the LSP, and output the sign of $c_{i,j.}$

2.2) for each entry (i, j) in the LIS do:

2.2.1) if the entry is of type A then output Sn(D(i, j)),

if Sn(D(i, j)) = 1 then

* for each $(k,l) \in O(i,j)$ do:

output Sn(k, l),

if Sn(k, l) = 1 then

add (k, l) to the LSP, output the sign of $c_{k,l}$, if Sn(k, l)=0 then

add (k, l) to the end of the LIP,

*if $L(i,j) \neq 0$ then

move (i, j) to the end of the LIS, as an entry of type B, go to Step 2.2.2).

Otherwise remove entry (i, j) from the LIS,

2.2.2) if the entry is of type B

output Sn(L(i, j)),

if Sn(L(i, j)) = 1 then

*add each h_m to the end of the LIS as an entry of type A,

*remove (i,j) from the LSP,

3) Refinement Pass:

For each entry (i, j) in the LSP, except those included in the last sorting pass (i.e., with the same n), output the *n*th most significant bit of | $c_{i,j}$ |;

4) Quantization Pass:

Decrement *n* by 1 and go to Step 2.

B. Analyze of SPIHT algorithm

Here a concrete example to analyze the output binary stream of SPIHT encoding. 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: $T_0=2^5$, for T_0 , the binary stream: 11100011100010000001010110000, 29 bits in all. By the SPIHT encoding results, we can see the output bit stream with a large number of seriate "0" situation , and with the gradual deepening of quantification, the situation will become much more severity, so there we have great output of redundancy when we direct output.

C. 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 ¼. Therefore, divide the binary output stream of SPIHT every 3 bits as a group recorded as a symbol, a total of 8 types of symbols, statistical probability that they appear and then encoded using variable length encoding naturally achieve the further compression , in this paper variable length encoder is Huffman encoder. In above example the output bit stream introduce a new encoding method process.

1) First divide the every output binary stream in to 3 bits as a group; <u>111</u> <u>000</u> <u>111</u> <u>000</u> <u>100</u> <u>000</u> <u>010</u> <u>101</u> <u>100</u> 00. In their process, there will have remaining 0, 1, 2 bits can not participate. So in the head of the output bit stream of Huffman encoding has two bits to record the number of bits which do not participate in group and that remainder bits are direct output in end. Figure 2 is shows the bit stream structure of Huffman encoding.

Number of remain bits	Bits stream	Remain bits

Figure2. the bit stream structure of Huffman encoding

2) The emergence of statistical probability of

each symbol grouping results 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 above probability results, by applying Huffman encoding obtain the code word book as following table 1.

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 <u>00</u> <u>01</u> <u>00</u> <u>01</u> <u>11</u> <u>01</u> <u>1001</u> <u>101</u> <u>11</u> 00, a total of 25 bits. The '10' in the first is binary of remainder bits numbers. The last two bits 'oo' are the result of directly output remainder bits. Compare with original bit stream save 4 bits. Decoding is reverse process of the above mentioned process.

III. MODIFIED RETINEX ALGORITHM

Based on wavelet transform, we propose a modified RETINEX algorithm. First, the image is processed by the wavelet transform. Second the horizontal and vertical low frequency component LL obtained by the wavelet transform is processed by the RETINEX Algorithm. Here in this after the Decoder process given to RETINEX Algorithm. And then enhanced image is obtained by inverse wavelet transform specific steps are follows,

a) The gray scale image after processing by the wavelet transform or Decoder. And then four components are obtained including horizontal and vertical low frequency component LL, horizontal high frequency and vertical low frequency HL, horizontal low frequency and vertical high frequency LH, and horizontal and vertical component HH.

b) The horizontal and vertical low frequency component LL is processed by Gaussian low pass filter, only the low frequency component LL is processed by Gaussian low pass filter can overcome the shortcoming of traditional RETINEX Algorithm that some high frequency components losed by filtering.

c) The logarithm of the reflectance of LL can be obtained,

$$\log R_{LL} = \log S_{LL} - \log L_{LL}$$

d) Then the reflectance of LL can be obtained

$$R_{LL} = \exp(\log S_{LL} - \log L_{LL})$$

e) R_{LL} which has been processed by Retinex Algorithm and HL,LH,HH are processed by inverse wavelet transform. Then enhanced image is obtained. Here by connecting all these components for this operation is shown in below Fig 3 block diagram. By this block diagram we will see how the image will be processed to functional blocks for compression and reconstruction.

IV. ANALYSIS OF EXPERIMENTAL RESULTS

In order to verify the validity of this algorithm, images usually using all are analyzed, we use 5-level pyramids constructed with the 9/7 tap filters. Table 2 is shown the experiment results of two standard 512*512 grayscale image Lena, Goldhill at different rate. Average code length which is calculated as follows,

$$L=\sum_{i=0}^{8} P(i)Li$$

Where p is the probability of symbols appeared, Li is the length of word code. From the experimental results, we see that values of L are less than 3, so we can achieve the compression effect. For each image in the same rate always the probability of each symbol appear flat and small fluctuations only, so saving the no of bits are also same thing. With the rate increase word code length in average (L) will be an increasing trend, but after the rate greater than 0.3bpp the trending will become very slow and more value of rate more bits will be save. Here we are use the modified Retinex Algorithm because of the traditional Retinex Algorithm has poor visibility and little contrast.

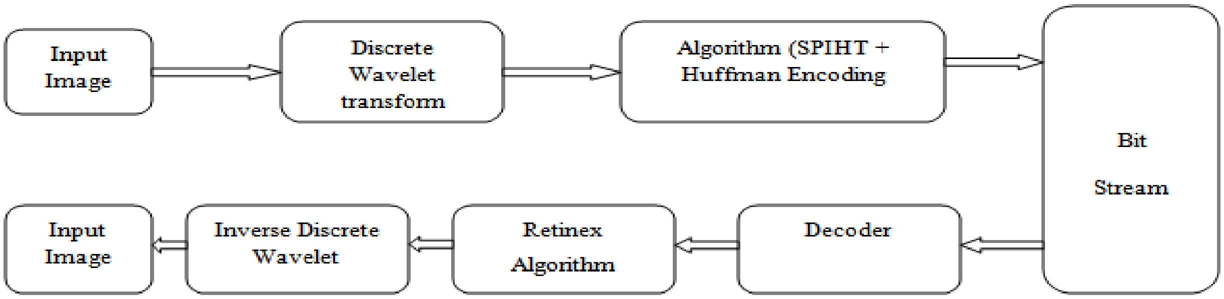

Fig 3. Block Diagram

Table 2 statistical results at different rate

Image	Lena			Goldhill		
Rate (bpp)	0.1	0.3	0.5	0.1	0.3	0.5
P('000')	0.2636	0.2362	0.2352	0.2521	0.2457	0.2205
P('001')	0.1364	0.1393	0.1383	0.1468	0.1441	0.1454
P('010')	0.1201	0.1229	0.1212	0.1271	0.1276	0.1282
P('011')	0.1001	0.1022	0.1028	0.0974	0.0974	0.1008
P('100')	0.1362	0.1397	0.1407	0.1403	0.1419	0.1461
P('101')	0.0759	0.0840	0.0832	0.0746	0.0767	0.0816
P('110')	0.0961	0.1021	0.1024	0.0938	0.0954	0.0992
P('111')	0.0716	0.0738	0.0762	0.0680	0.0713	0.0782
Average code length	2.8839	2.9216	2.9242	2.8905	2.9023	2.9393
The no of output bits by SPIHT encoding	26225	78650	131080	26225	78647	131080
The no of output bits by the new algorithm encoding	25430	76802	127986	25478	76287	128642
The no of saving bits	795	1848	3094	747	2360	2438

Fig 4. Original image

Fig 5. Image processed by traditional Retinex Algorithm

Fig 4 is original image and Fig 5 is the resultant processed by traditional Retinex Algorithm Fig 6 is the output image which has been processed by modified Retinex Algorithm, the image processed by modified Retinex Algorithm is very clearer in visible and uniform illumination.

Fig 6. Image processed by modified Retinex Algorithm

V. CONCLUSION

Proposing a simple and effective method combined with Huffman encoding for further compression. In this paper a lot of bits are saves for the data transmission and storage purpose. In this decoding process uses the Retinex Algorithm for clear vision and quality improvement. So by using this large no. of image data's to be transmitted.

VI. REFERENCES

[1] Wei Li, Zhen Peng Pang "SPIHT Algorithm with Huffman Encoding" Intelligent Information Technology and Security Informatics (IITSI), 2010 Third International Symposium on. 22 april 2010.

[2] Ming Hao, Xingbo Sun "A modified Retinex Algorithm based on Wavelet Transformation", 2010 Second International Conference on Multi Media and Information Technology.

[3] FAN Qi-bin. "Wavelet analysis". Wuhan: Wuhan University Press, 2008.

[4] Cheng Li-chi, Wang Hong-xia, Luo Yong. "Wavelet theory and applications" Beijing: Science Press, 2004(Chinese).

[5] H.Wang, S.J Li, Y.Wang, "Face Recognition under Varying Lighting Conditions Using Self Quotient Image.Proc". IEEE Intl Conf. Automatic Face and Gesture Recognition, 2004.

[6] Rafael C. GONZALEZ Richard E. WOODS. "Digital image processing": second ed [M]. Beijing Publishing House of Electronics Industry 2002.

[7] Amir SAID 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 Video Technology 1996.

[8] J. M. SHAPIRO. "Embedded image coding using zero tree of wavelets coefficients [J]". IEEE Trans. Signal Processing 1993.

VII. About The Authors

A.Mallaiah received the M.E degree in Electronic Instrumentation from Andhra University, Visakhapatnam in 2004, B.Tech degree in Electronics and communication Engineering from R.V.R & J.C College of Engineering, Guntur in 2002. He is Associate Professor, Department of Electronics and Communication Engineering at Gudlavalleru Engineering College, Gudlavalleru. He has a total teaching experience (UG and PG) of 7 years. He has guided and co-guided 6 P.G students. His research areas include Embedded Systems, Molecular Electronics, Digital Signal Processing and Digital Image Processing.

SK.Shabbir is pursuing his M.Tech degree in Digital Electronics & Communication Systems from Gudlavalleru Engineering College, Gudlavalleru, B.Tech from Electronics & Communication Engineering from QIS College of Engineering and Technology, Ongole in 2009. His research interest includes Digital Image processing and Embedded Systems.

T.Subhashini is working as an Asst.Professor in the Department of Electronics & Communication Engineering at Sri Vasavi Institute of Engineering & Technology, Pedana, She has 2 international journal publications to her credit. She has Completed M.Tech degree in Digital Electronics & Communication Systems from Gudlavalleru Engineering College, Gudlavalleru in 2010, B.Tech from Electronics &Communication Engineering from V.R Siddhartha Engineering College, Vijayawada in 2005. Her main research interest includes Digital Signal Processing, Digital Image Processing and Embedded Systems.