OPEN ACCESS
International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-3 Issue-3, August 2013

A new Set Partitioning in Hierarchical (SPIHT) Algorithm and Analysis with Wavelet Filters

Vidhi Dubey, Rahul Dubey

Abstract- Spiht-Set Partitioning in Hierarchical Trees algorithm is widely used as a compression and encoding algorithm for satellite image compression and transmission. Though it provides efficient lossless compression with high PSNR the associated complexity of algorithm is very high which makes it unfeasible for many practical hardware implementations. Based on the SPIHT algorithms, we define two modifications to develop a simpler image coding method. The first concept is obtained from the relationship between the bit-planes and the target bit- rate. The second concept is obtained by applying different wavelet filters. Based on the above mentioned concepts, we can discard the refinement pass and improve the image quality at different target bit-rates. The project implements image codec's based on both the algorithms and compares their performance on the basis of PSNR values. The images used are square grayscale images. The programming is done in java platform

Keywords—wavelet filter, compression, encoding, PSNR

I. INTRODUCTION

Internet Communication is not very secure. Virtually all business, government and academic organizations transfer data through the Internet. Some of the data may be confidential which must be protected from a third person, but due to the lack of security an intruder in the network may be able to view this information. So various security measures need to be taken to protect data that are sent through the Internet. The main objective of this project is to secure the image, which will be transmitted. The image can be any type such as bmp, jpeg, gif, icon, etc. Cryptography is an effective method of protecting data that are passed through the network. Various algorithms are available to carry out cryptography. Internet security consists of measures to detect and prevent security attacks and to correct security violation. The SPIHT (Set Partitioning in Hierarchical Trees) algorithm is a fast and efficient technique for image compression and encryption. SPIHT generally operates on an entire image at once. The whole image is loaded and transformed and then the algorithm requires repeated access to all coefficient values. After the wavelet transform, we use this algorithm to encode the wavelet coefficients.

In recent years, much research work has been done to improve [1] , [2], [3], ex-tend SPIHT [4] ,[5] suggested a new coding technique to improve SPIHT. However, mentioned research developments increase the coding complexity to achieve better coding performance. Sun et al. (2002) bits" proposed a method that suggests that the "number of error should describe in detail how to adaptively determine the threshold at different target bit-rates or bandwidths. However, recently mobile devices and information appliances with limited computation resources and battery power have acquired lower-complexity multimedia coders instead. In this paper, we suggest two modifications to develop a simpler image coding algorithm. The first concept is based on the relation-ship between the bit-planes and the target bit-rate. The second concept is obtained from the application of different wavelet filters in The SPIHT. Based on these two concepts, we can implement a new SPIHT with the features of wavelet filters . An acceptable peak signal to noise ratio (PSNR) loss of new SPIHT is obtained as compared to the original SPIHT.

In Section II, we first introduce Image compression fundamentals, In Section III we explain about wavelet transform, In section IV we first introduce the original SPIHT algorithm. In Section V, we present the coding procedure for the simplified SPIHT algorithm. The simulation results and conclusions are presented in Sections VI and VII, respectively.

Manuscript received on August, 2013.
Vidhi Dubey,
Rahul Dubey

II. IMAGE COMPRESSION

A common characteristic of most images is that the neighboring pixels are correlated and therefore contain redundant information. The foremost task then is to find less correlated representation of the image. Two fundamental components of compression are redundancy and irrelevancy reduction. Redundancy reduction aims at removing duplication from the signal source (image/video). Irrelevancy reduction omits parts of the signal that will not be noticed by the signal receiver, namely the Human Visual System (HVS). In general, three types of redundancy can be identified:

Spatial Redundancy or correlation between neighboring pixel values, Spectral Redundancy or correlation between different color planes or spectral bands, Temporal Redundancy or correlation between adjacent frames in a sequence of images (in video applications).Image compression research aims at reducing the number of bits needed to represent an image by removing the spatial and spectral redundancies as much as possible. Since we will focus only on still image compression, we will not worry about temporal redundancy.

Two ways of classifying compression techniques are mentioned here. Lossless vs. Lossy compression: In lossless compression schemes, the reconstructed image, after compression, is numerically identical to the original image. However lossless compression can only a achieve a modest amount of compression. An image reconstructed following lossy compression contains degradation relative to the original.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

Often this is because the compression scheme completely discards redundant information. Predictive vs. Transform coding: In predictive coding, information already sent or available is used to predict future values, and the difference is coded. Since this is done in the image or spatial domain, it is relatively simple to implement and is readily adapted to local image characteristics. Differential Pulse Code Modulation (DPCM) is one particular example of predictive coding. Transform coding, on the other hand, first transforms the image from its spatial domain representation to a different type of representation using some well-known transform and then codes the transformed values (coefficients). This method provides greater data compression compared to predictive methods, although at the expense of greater computation.

III. WAVELET TRANSFORM

Wavelet analysis is an exciting new method for solving difficult problem in image processing, pattern recognition, computer graphics. Wavelets allow complex information such as images and patterns to be decomposed into elementary forms at different positions and scales and subsequently reconstructed with high precision. The Fourier transform is an useful tool to analyze the frequency components of the signal. However, if we take the Fourier transform over the whole time axis, we cannot tell at what instant a particular frequency rises. Short-time Fourier transform (STFT) uses a sliding window to find spectrogram, which gives the information of both time and frequency. But still another problem exists: The length of window limits the resolution in frequency. Wavelet transform seems to be a solution to the problem above. Wavelet transforms are based on small wavelets with limited duration. The translated-version wavelets locate where we concern. Whereas the scaled-version wavelets allow us to analyze the signal in different scale. wavelets as a family of functions constructed from translations and dilations of a single function called the "mother wavelet" ψ(t). They are defined by

$$\psi_{(a,b)}(t) = \frac{1}{\sqrt{|a|}}\psi\left(\frac{t-b}{a}\right); \quad a,b \in R, a \neq 0 \quad \dots(1)$$

The parameter a is the scaling parameter or scale, and it measures the degree of compression. The parameter b is the translation parameter which determines the time location of the wavelet. If $|a| < 1$, then the wavelet in (1) is the compressed version (smaller support in time- domain) of the mother wavelet and corresponds mainly to higher frequencies. On the other hand, when $|a| > 1$, then $\psi_{a,b}(t)$ has a larger time-width than ψ and corresponds to lower frequencies. Thus, wavelets have time-widths adapted to their frequencies. This is the main reason for the success of the Morlet wavelets in signal processing and time-frequency signal analysis. The DWT of a signal x is calculated by passing it through a series of filters. First the samples are passed through a low pass filter with impulse response resulting in a convolution of the two. The signal is also decomposed simultaneously using a high-pass filter h. The outputs giving the detail coefficients (from the high-pass filter) and approximation coefficients (from the low-pass). It is important that the two filters are related to each other and they are known as a quadrature mirror filter. However, since half the frequencies of the signal have now been removed, half the samples can be discarded according to Nyquist's rule. The filter outputs are then sub sampled. This decomposition has halved the time resolution since only half of each filter output characterizes the signal. However, each output has half the frequency band of the input so the frequency resolution has been double.

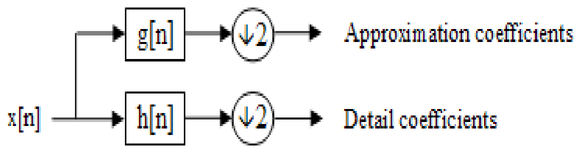


Fig 1 wavelet decomposition one level

This decomposition is repeated to further increase the frequency resolution and the approximation coefficients decomposed with high and low pass filters and then down-sampled. This is represented as a binary tree with nodes representing a sub-space with different time-frequency localization. The tree is known as a filter. At each level in the above diagram the signal is decomposed into low and high frequencies.

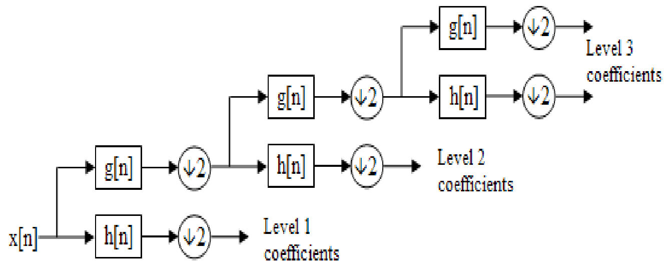


Fig 2 wavelet decomposition two level

Due to the decomposition process the input signal must be a multiple of 2^n where n is the number of levels.

IV. SPIHT ALGORITHM

The SPIHT (Set Partitioning in Hierarchical Trees) algorithm can perform better at higher compression ratios for a wide

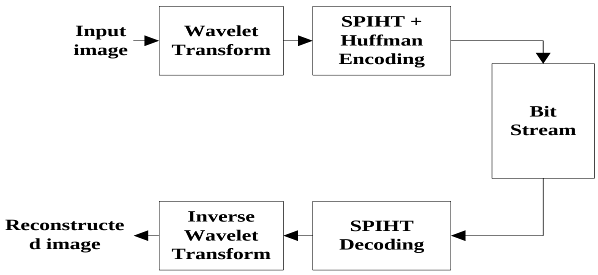


Fig 3. SPIHT algorithm Block diagram

variety of images. The algorithm uses a partitioning of the trees in a manner that tends to keep insignificant coefficients together in larger subsets. Before the algorithm is explained, certain notations are to be made familiar. The wavelet coefficients are again divided into trees originating from the lowest resolution band (band I). The coefficients are grouped into 2 x 2 arrays that, except for the coefficients in band I, are off springs of a coefficient of a lower resolution band.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

The coefficients in the lowest resolution band are also divided into 2 x 2 arrays. The coefficient in the top-left comer of the array does not have any off spring.
The trees are further partitioned into four types of sets, which are sets of coordinates of the coefficients: O (i, j) This is the set of coordinates of the off springs of the wavelet coefficient at location (i, j). D (i,j) This is the set of all descendants of the coefficient at location (i,j) H is set of all root nodes. L (i,j) This is the set of coordinates of all the descendants of the coefficient at location except for the immediate off springs of the coefficient at location (i,j). So,

$$L(i, j) = D(i, j) - O(i, j) \qquad (2)$$

A set D (i, j) or L (i,j) is said to be significant if any coefficient in the set has a magnitude greater than the threshold. The algorithm makes use of three lists: the list of insignificant pixels (LIP), the list of significant pixels (LSP), and the list of insignificant sets (LIS). The LSP and LIS lists contain the coordinates of coefficients, while the LIS contains the coordinates of the roots of sets of type D or L .The initial value of the threshold is given as:

$$n = \log_2 C_{max} \qquad (3)$$

Where, Cmax is the maximum magnitude of the coefficients to be encoded. The LIP list is initialized with the set H. Those elements of H that have descendants are also placed in LIS as type D entries. The LSP list is initially empty. In each pass, the members of the LIP are first processed, then the members of LIS. This is essentially the significance map encoding step. In the refinement step the elements of LSP are processed. Each coordinate contained in LIP is examined first. If the coefficient at that coordinate is significant (i.e., it is greater than 2n), a 1 is transmitted, followed by a bit representing the sign of the coefficient (1 for positive,0 for negative). Then that coefficient is moved to the LSP list. If the coefficient at that coordinate is not significant, a 0 is transmitted. After examining each coordinate in LIP, the sets in LIS are examined. If the set at coordinate (i, j) is not significant, a 0 is transmitted. If the set is significant, a 1 is transmitted. If the set is of type D, each of the off springs of the coefficient at that coordinate is checked. For each coefficient that is significant, a 1 is transmitted, the sign of the coefficient, and then the coefficient is moved to the LSP. For the rest a 0 is transmitted and their coordinates are added to the LIP. If this set is not empty, it is moved to the end of the LIS and marked as type D. This new entry into the LIS has to
be examined during this pass. If the set is empty, the coordinate (i, j) is removed from the list. If the set is of type D, each coordinate in D(I ,j) is added to the end of the LIS as the root of a set of type D. Again, note that these new entries in the LIS have to be examined during this pass. Then (i, j) is removed from the LIS. Once each of the sets in the LIS (including the newly formed ones) is processed, a refinement step is started. In the refinement step each coefficient that was in the LSP prior to the current pass is examined and output the nth most significant bit of | ci,j|. The coefficients that have been added to the list in this pass are ignored because, by declaring them significant at this particular level, the decoder has already been informed of the value of the nth most significant bit. This completes one pass. Depending on the availability of more bits or external factors, n is decremented by one and the process continues.

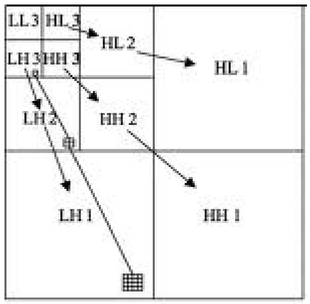


Fig 4. spatial orientation trees in SPIHT

V. 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. Divide the every output binary stream in to 3 bits as a group; 111 000 111 000 100 000 010 101100 00. In their process, there will have remaining 0, 1, 2 bits cannot 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.

VI. RESULT AND DISCUSSION

The following results were obtained with monochrome 512×512 images at different bit rate. A Max grey level = 255 has been used (as there are 0 to 255 grey levels represented with 8 bits in the BMP format images). Here we used bit rate from .25 to 1 bpp. It is important to observe that the bit rates are not entropy estimates- they were calculated from the actual size of the compressed file. Furthermore, by using the progressive transmission ability, the sets of distortion are obtained from the same file, that is, the decoder read the first bytes of the file (up to the desire rate), calculated the inverse sub band transformation , and then compare the recovered image with the original image. The distortion is measured by the peak signal to noise ratio:

$$PSNR = 10 \mathrm{Log}_{10}\left(\frac{255^2}{MSE}\right) \mathrm{db} \qquad (4)$$

(a)

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

(c)

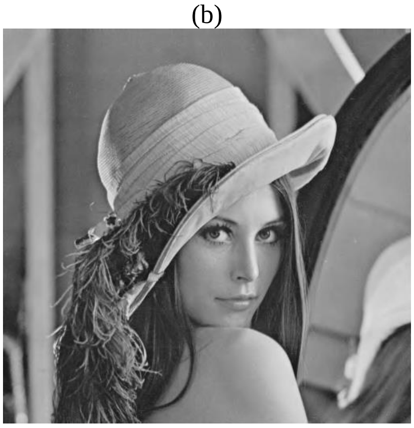
Fig 5. Images obtained with the modified SPIHT method (d) rate = 1

where,

$$MSE = \frac{1}{M \times N}\sum_{i=1}^{M}\sum_{j=1}^{N}[I(i,j) - I_{reconstruct}(i,j)]^2 \quad (5)$$

where M and N are rows and column of image . I is the original image and $I_{reconstruct}(i,j)$ is the image obtained after reconstruction. Analysis of different wavelet filter on this coder also taken. Used wavelet filters are Daubechies, Bior, Coiflet . Table 1 shows the comparison of two algorithm image compression quality, it can be seen from Table 1, new algorithm compared with classical SPIHT, PSNR has decreased at lower bit rate, but PSNR has improved at higher bit rate. Our algorithm (Modified SPIHT) gives an improved result as compare to existing SPIHT at all bpp . Analysis of different wavelet filters as previously explain have also done. PSNR value of Lena image at different bpp and at different wavelet filter have shown in Table II. All PSNR values give an improved result. But bior 6.8 has given a good result as compare to other two wavelet filters.

(b)



(d)

(a) original Lena image (b) rate = .25 (c) rate = .50

Lena	PSNR/dB(Lena)	
/bpp	Modified SPIHT	SPIHT
0.25	33.23	33.12
0.5	36.59	36.2
0.75	38.3	37.8
1	39.85	39.38

Table I

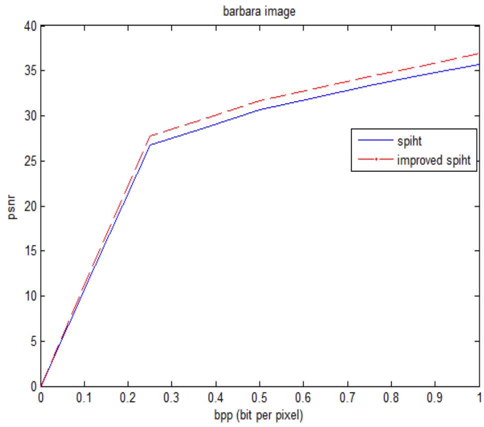


Fig 6

bpp	Wavelet Filter(PSNR in dB)		
	bior 6.8	dB 10	coif 5
0.25	33.38	32.77	33.23
0.5	36.68	36.22	36.55
0.75	38.41	38.03	38.25
1	39.92	39.66	39.85

Table II

VII. CONCLUSION

The modified SPIHT algorithm given higher PSNR at higher bpp and it given an average improvement of .36 dB at all bpp. Analysis on different wavelet filter also given a good PSNR value at different bpp.

REFERENCES

1. Zhu, J., and Lawson, S., 2001, “Improvements to SPIHT for Lossy Image Coding,” IEEE International Conference on Electronics, Circuits and Systems, Vol. 3, pp. 1363-1366
2. Weng, S. J., and Areekul, V., 2003, “Adaptive Directional Zero-Tree Image Coding,” Proceedings of IEEE International Symposium on Circuits and Systems, Bangkok, Thailand, Vol. 2, pp. 424-427.
3. Sun, Y., Zhang, H., and Hu, G., 2002, “Real-Time Implementation of a New Low-Memory SPIHT Image Coding Algorithm Using DSP Chip,” IEEE Trans-actions on Image Processing, Vol. 11, No. 9, pp.1112-1116..
4. Pearlman, A. W., Islam, A., Nagaraj, N., and Said, A., 2004, “Efficient, Low-Complexity Image Coding with a Set-Partitioning Embedded Block Coder,” IEEE Transactions on Circuits and Systems for Video Technology, Vol. 14, No. 11, pp. 1219-1235.
5. Zhu, J., and Lawson, S., 2001, “Improvements to SPIHT for Lossy Image Coding,” IEEE International Conference on Electronics, Circuits and Systems, Vol. 3, pp. 1363-1366.

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)