MINIMIZING AND MAXIMIZING OF ENCODED IMAGE BY WAVELET TRANSFORM

Manikandan N K, Manivannan D, Antony Kumar K
Assistant Professor, Department of Computer Science and Engineering,
Veltech Dr. RR& Dr.SR University, Avadi, Chennai, India

ABSTRACT
In this work, compression and decompression of an encoded image with stretchy density percentage. A pseudorandom variation is used to encode an unique image, and the encoded data are powerfully compressed and decompressed using wavelet transform. Decompression is performed by inverse operation of Wavelet Transform. The wavelet transform has emerged as a cutting edge tools within the ground of image compression. Wavelet Transform decomposed the spectrum into set of group limited components are called sub band. Ideally, the sub bands can be assembled back to reconstruct the original spectrum without any error. Wavelet- based coding provides significant improvement in picture class at higher density ratio. DWT yield high compression ratio and better the value of the reconstruct figure.

Key words: Image Encryption; Image Compression; Image Decompression.

http://www.iaeme.com/IJCIET/issues.asp?JType=IJCIET&VType=8&IType=10

1. INTRODUCTION
In past years, compression of encoded data has concerned significant research attention. The customary way of securely and ably transmit unneeded data is to first compress the data to decrease the idleness, and then to encode the compressed data to cover its sense. At the receiver side, the decryption and decompression operation are organized to perform to get well the original data. That is the sender should encode the real data and the system supplier may tend to compress the encoded data with no any awareness of the cryptographic key and the real data. At recipient side, a decoder integrating decompression and decryption functions will be used to restructure the original data.

A number of techniques for compressing/decompressing encrypted data have been developed. It has been made known in [1] that, based on the theory of resource coding with side information at the decoder, the presentation of compressing encrypted data may be as good as that of compressing non encrypted data in theory. Two well-situated approaches to lossless compression of encrypted black and white images and to lossy compression of encrypted
Gaussian sequence also exist in [1]. In the earlier approach, the original binary image is encrypted by adding up a pseudorandom string, and the encrypted data are compressed by ruling the syndromes with respect to low-density parity-check (LDPC) channel code [2]. In the ending one, the unique data is encrypted by adding an id. Gaussian sequence, and the encoded data are checked and compressed as the syndromes of trellis code. The density of encoded data for both memory less sources and sources with hidden Markov correlation using LDPC codes is also considered [3]. By employ LDPC code into a variety of bit-planes and exploit the spatial and cross-plane connection among pixels, a few methods for lossless compression of encoded gray and color images are introduce in [4]. In [5], the encoded image is decomposed and the most important bits in high levels are compressed using rate-compatible punctured turbo codes. The decoder can monitor a low-resolution report of the image, learn local statistics based on it, and use the information to obtain the contented in high levels. Furthermore, by increasing statistical model for source data and extend these models to video, [6] presents some algorithms for compressing encoded data and reveal blind compression of encoded video. In [7], a technique called compressive sensing is established to attain lossy compression of encoded image data, and a basis search algorithm is suitably modified to allow joint decompression and decryption. The signal processing in the encryption domain using homomorphic calculation is also discussed in [8] and [9]. Content from the encoded image with ordinar

2. COMPRESSION AND DECOMPRESSION OF ENCRYPTED IMAGE

In this work, a pseudorandom transformation is used to encode an original image. Then, the encrypted data can be efficiently compressed and decompressed by DWT based SPIHT algorithm. When having the compressed data and the permutation way, with the aid of spatial relationship in normal image, the receiver can recreate the major content of the unique image by IDWT using SPIHT algorithm.

A. Image Encryption

Assume the unique image is in uncompressed design and every pixel with a gray rate falling into [0, 255] is denoted by 8 bits. Represent the numbers of the rows and the columns in the unique image as \( N_1 \) and \( N_2 \), and the number of all pixels as \( N = N_1 \times N_2 \). Then, the sum of bits of the unique image is \( 8N \). For image encoding, the data sender pseudo randomly permutes the pixels and the version way is determined by a secret key. The permuted pixel-sequence is view as the encoded data.

A number of permutation-based image encryption methods can be used here [10], [11], only the pixel location are permuted and the pixel values are not masked in the encoded phase, an invader without information of the secret key can know the original histogram from an encrypted image. However, the number of possible permutation ways is \( N! \) so that it is unpractical to achieve a brute force search when is practically large. That means the invader cannot improve the unique y size and fluctuation.

B. SPIHT

SPIHT is a wavelet-based image compression coder. It first converts the image into its wavelet transform and then transmits information about the wavelet coefficients. The decoder uses established signal to recreate the wavelet and perform an inverse transform to improve the image. We preferred SPIHT because SPIHT and its predecessor, the embedded zero tree wavelet coder, were important breakthrough in still image compression in that they accessible extensively enhanced quality over vector quantization, JPEG, and wavelets combined with quantization, while not requiring training and producing an embedded bit stream. SPIHT displays exceptional characteristics over a number of properties all at once including:
Minimizing and maximizing of Encoded Image by Wavelet Transform

- excellent value of image with a high PSNR
- Fast coding and decoding
- A fully progressive bit-stream
- Can be used for lossless compression
- can be united with error safety
- Ability to code for accurate bit rate or PSNR

The Discrete Wavelet Transform (DWT) runs a high and low-pass filter above the signal in one width. The effect is a new image comprises a high and low-pass sub band. This process is then repetitive in the other measurement resulting four sub bands, three high-pass mechanism and one low pass module. The next wavelet level is composed by repeat the horizontal and vertical transformations on the low-pass sub band from the previous level. The DWT repeats this procedure for though many levels are required. Each method is entirely reversible (within the limit of permanent accuracy) so that the unique image can be formed from the wavelet transformed image.

SPIHT is a method of coding and decoding the wavelet change of an image. By coding and transmitting information about the wavelet coefficients, it is likely for a decoder to perform an inverse transformation on the wavelet and reconstruct the original image. The entire wavelet transform does not need to be transmitting in order to get better the image.

C. Wavelet Architectures

The one-dimensional DWT entails demanding computations, which involve significant hardware resources. Most two-dimensional DWT structure have implement folding to reprocess logic for each measurement, since together the horizontal and vertical passes use the same FIR filters. Fig.1 illustrates how the fold architecture uses a 1 dimensional DWT to realize a 2 dimensional DWT.

Such architecture suffers from high memory bandwidth. For an N x N image there are at least 2N2 read and write cycles for the first wavelet level. Additional wavelet levels require re-reading previously computed coefficients, further reducing efficiency.

![Figure 1 Illustration of folded architecture](image)

In order to deal with these superfluous memory accesses the Partitioned DWT was designed. The Partitioned DWT partitions the image into smaller blocks and computes several scales of the DWT at once for each block. In addition, the algorithm makes use of wavelet lifting to reduce the computational difficulty of the DWT. By partition an image into minor blocks, Fig.2 the amount of on-chip memory storage required is significantly condensed since only the coefficients in the block need to be stored.
For larger images that require several individual wavelet scales, the Generic 2-D Biorthogonal DWT architecture use a more quantity of on chip resources. With SPIHT, a 1024 by 1024 pixel image compute seven undo wavelet scales. The proposed architecture would employ 21 individual high and low pass FIR filters. Since each wavelet scale processes data at unlike rates, a part of clock signal is also needed for each scale. The advantage of the architecture is full utilization of the memory’s bandwidth since each pixel is only read and written once.

3. 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 Fig.3. This process is repeated until the top of the pyramid is reached.

There exists a spatial relationship among the 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:

\[ O(i,j) = \{(2i,2j),(2i,2j+1),(2i+1,2j),(2i+1,2j+1)\} \]  

and each of them recursively maintains a spatial similarity to its corresponding four off-spring. This pyramid structure is commonly known as spatial orientation tree. For example, Fig.4 shows the similarity among sub-bands within levels in the wavelet space. 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 is significant by means of a binary search algorithm.

**Figure 4** Off-spring dependencies in the pyramid structure

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 Fig.5, 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.

**Figure 5** Bit-plane ordering and transmission scheme

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:

$$\max_{(i,j) \in \tau_m} |c_{i,j}| \geq 2^n$$

where $c_{i,j}$ is the wavelet coefficient at the $n^{th}$ bit plane, at location $(i,j)$ of the $\tau_m$ 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).

$$S_n(\tau) = \begin{cases} 1, & \max_{(i,j) \in \tau_m} |c_{i,j}| \geq 2^n \\ 0, & \text{otherwise} \end{cases}$$

Wavelets coefficients which are not significant at the nth bit-plane level may be significant at (n-1) the 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.

The SPIHT algorithm can be summarized as follows

1. **Initialization:** Output \( n = \left\lfloor \log_2(\max_{(i,j)} |C_{i,j}|) \right\rfloor \); set the LSP as empty list and add the coordinates \((i,j) \in H \) to the LIP and only those with descendents also to the LIS, as type A entries.

2. **Sorting Pass**
   1.1. for each entry \((i,j)\) in the LIP do:
   2.1.1 Output \( S_n(i,j) \)
   2.1.2 If \( S_n(i,j) = 1 \) 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 \( S_n(D(i,j)) \);
      * if \( S_n(D(i,j)) = 1 \) then
        * for each \((k,l) \in O(i,j)\) do:
          * Output \( S_n(k,l) \)
          * If \( S_n(k,l) = 1 \) then add to the LSP and output the sign of \( C_{k,l} \)
          * If \( S_n(k,l) = 0 \) then add \((k,l)\) to the end of LIP
      * If \( L(i,j) \neq \emptyset \) then move \((i,j)\) to the end of the LIS as entry of type B, and go to step 2.2.2; otherwise remove \((i,j)\) entry from the LIS:
   2.2.2 if the entry is of type B then
      * Output \( S_n(L(i,j)) \)
      * If \( S_n(L(i,j)) = 1 \) then
        * add each \((k,l) \in O(i,j)\) to the end of the LIS as entry of type A
        * remove \((i,j)\) from the LIS

2. **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 \( |C_{i,j}| \).

3. **Quantization step update:** decrement \( n \) by 1 and go to step 2

Notations used in the algorithm are defined as follows:

- \( O(i,j) \): set of coordinates of the off-spring \((i,j)\)
- \( D(i,j) \): set of coordinates of all descendants \((i,j)\)
- \( H(i,j) \): set of coordinates of all tree roots in the highest level of the pyramid
- \( L(i,j) = D(i,j) - O(i,j) \)

4. **CONCLUSION**

This work proposed a novel idea for compressing and encrypted image and designed a practical scheme made up of image encryption, lossy compression, and decompression. The original image is encrypted by pseudorandom permutation. When having the compressed data and the permutation way, a DWT is used to retrieve the values of coefficients in natural image, and SPIHT algorithm is used to compress and decompress the encrypted image. The compression ratio and the quality of reconstructed image vary with different values of compression.
Minimizing and maximizing of Encoded Image by Wavelet Transform

parameters. In the encryption phase of the proposed system, only the pixel positions are shuffled and the pixel values are not masked.

REFERENCES


