Efficiency Spiht In Compression And Quality Of Image

Asst. Lecturer **Enas M. Jamel***

Abstract

Image compression is an important tool to reduce the bandwidth and storage requirements of practical image systems. To reduce the increasing demand of storage space and transmission time compression techniques are the need of the day. Discrete time wavelet transforms based image codec using Set Partitioning In Hierarchical Trees (SPIHT) is implemented in this paper. Mean Square Error (MSE), Peak Signal to Noise Ratio (PSNR) and Maximum Difference (MD) are used to measure the picture quality of reconstructed image. MSE and PSNR are the most common picture quality measures. Different kinds of test images are assessed in this work with different compression ratios. The results show the high efficiency of SPIHT algorithm in image compression.

Keywords: Image compression, Lossy compression, SPIHT, Image Quality Measures

1-Introduction

Image compression can be divided into two main categories, lossless and lossy compression. Lossless and lossy compressions are terms that describe whether or not all original images can be recovered when the compressed one is uncompressed. With lossless compression, every single bit of data that was originally in the image remains after the image uncompressed (The uncompressed image called reconstructed image), and the information completely restored. On the other hand, lossy compression reduces an image by permanently eliminating certain information, especially redundant information. When the image is uncompressed, only a part of the original information is still there (although the user may not notice it) [1].

Since first attempts, the discrete cosine transform (DCT) domain has been used. DCT is used in the popular JPEG file format, and most video compression methods are generally based on that method [2]. JPEG is very efficient coding method but the performance of block-based DCT scheme degrades at high compression ratio [3].

In recent time, much of research activities in image coding have been focused on the Discrete Wavelet Transform (DWT). DWT offers adaptive spatial-frequency resolution (better spatial resolution at high frequencies and better frequency resolution at low frequencies) that is well suited to the properties of human visual system. It can provide better image quality than DCT, especially at higher compression ratio. Set Partitioning In Hierarchical Trees (SPIHT) coding algorithm introduced by Said and Pearlman [8] is a very efficient technique for wavelet image compression. SPIHT is improved of Embedded Zero tree Wavelet (EZW) coding algorithm introduced by J. M. Shapiro [3]. It works on discrete wavelet transform coefficients of an image. It is very effective and computationally simple technique for image compression. SPIHT is refinement to EZW and uses its principles of operation and has even better performance than EZW in image compression [4]. SPIHT coding operates by exploiting the relationships among the wavelet coefficients across the different scales at the same spatial location in the wavelet sub bands and most efficient when using level 5 and higher [7,2].

* Dept. of Computer Science- College of Education for Women- Baghdad University

This paper implements SPIHT coding using the MATLAB platform. It displays the image quality that is measured objectively using Peak Signal to Noise Ratio (PSNR) and Mean Square Error (MSE).

2-Wavelet Transform

Over the past several years, the wavelet transform has gained widespread acceptance in signal processing in general and in image compression research in particular. In many applications wavelet-based schemes outperform other coding schemes like the one based on DCT [6].

Lifting approach uses to implement the computation of the discrete-time wavelet transform. Lifting scheme is derived from a polyphase matrix representation of the wavelet filters, a representation that is distinguished between even and odd samples by using the algorithm of filter factoring to split the original filter into a series of shorter filters. Those filters are designed as lifting steps; each step one group of coefficients are lifted (altered) with the help of the other one (classical dyadic transform always leads to two groups of coefficients, low-pass and high-pass) [2].

In the case of the most widely used image processing wavelet filters – the Cohen-Daubechies-Feauveau 9/7-tap filters (CDF 9/7) – it is an easy task since the most efficient scheme has already been proposed and used (for example in the JPEG 2000 codec). Computational savings of the scheme are gained from the length of the filters (convolution with a 9-tap filter is slower than with a series of two-tap filters) and due to minimum dependency between the coefficients [2].

Image can be viewed in the frequency domain as partitioning into a set of subbands, where each partitioning step is obtained by applying the 2D wavelet transform. One level of 2D wavelet transform results in four sets of data (wavelet coefficients), that correspond to four 2D frequency sub bands. For these four subbands, if the original image data is on the zero decomposition level (scale), use the following notation on kth decomposition level:

HH_K, HL_K, LH_K, LL_K …….. (partitioning step)

HH_K (high–high or diagonal details), HL_K (high–low or horizontal details), LH_K (low–high or vertical details), and LL_K (low–low or approximation). LL_K sub band is also called image approximation as it represents image on a lower scale, while to other subbands refer as to image details. Wavelet decomposition is dyadic in a case when only the LL_K sub band is further transformed. It results in a new set of subbands: HH_{K+1}, HL_{K+1}, LH_{K+1}, and LL_{K+1}. Dyadic decomposition used in image compression will thus generate hierarchical pyramidal structure, as shown in Figure1.If the dyadic decomposition of N levels is performed (N times transforming the low–low subband) the result will be 3N+1 subbands [5].

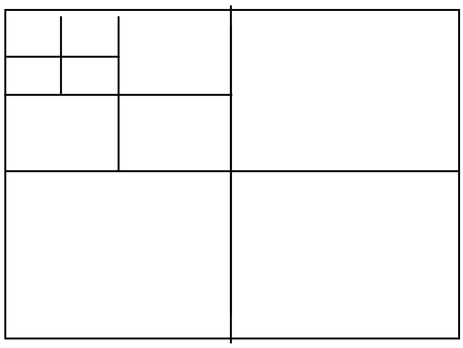


Figure1. Pyramidal structure of 3-level wavelet decomposition

Wavelet-based coding is more robust under transmission and decoding errors, and also facilitates progressive transmission of images, because of their inherent multiresolution nature. Wavelet coding schemes are especially suitable for applications where scalability and tolerable degradation are important [6].

3- SPIHT Coding Scheme

The SPIHT algorithm operates on a wavelet-transformed image with equal length and width of an integer power of 2. It encodes the wavelet coefficients in a way that uses a hierarchical organization of the coefficients [6]. The image is first converted into its wavelet transform and then the wavelet coefficients are fed to the encoder. The primary reason behind the use of wavelet transformation is that, the transform coefficient $c_{(i,j)}$ has a greater significance than that of the pixel $p_{(i,j)}$ in image compression. Let p be the original image where (i, j) is the pixel coordinate, then its coefficients c obtained with a unitary hierarchical subband transformation [7].

SPIHT codes the wavelet coefficients in a way that uses a hierarchical organization of the coefficients [6]. SPIHT coding involves the coding of the position of significant wavelet coefficients and the coding of the position of zero trees in the wavelet subbands. The SPIHT coder exploits the following image characteristics: 1) the majority of an image's energy is concentrated in the low frequency components and a decrease in variance is observed as it moves from the highest to the lowest levels of the subband pyramid and 2) it has been observed that there is a spatial self-similarity among the subbands, and the coefficients are likely to be better magnitude-ordered if it moves downward in the pyramid along the same spatial orientation [7].
The following sets of coordinates are used to represent the coding method [7-8]:

O (i,j): set of coordinates of all offspring of node (i,j).

D (i,j): set of coordinates of all descendants of node (i,j).

H (i,j): set of all tree roots (nodes in the highest pyramid level).

L (i,j): D (i,j) – O(i,j) (all descendents except the offspring [5].

SPIHT makes use of three lists:
List of insignificant sets (LIS): contains the set of wavelet coefficients defined by tree structures, and found to have magnitude smaller than a threshold (are insignificant). The sets exclude the coefficient corresponding to the tree or all sub tree roots, and have at least four elements. The entries in LIS are sets of the type $D(i, j)$ (type A) or type $L(i, j)$ (type B).
List of insignificant pixels (LIP): contains the individual coefficients that have magnitude smaller than the threshold.
List of significant pixels (LSP): contains the pixels that are found to have magnitude larger than the threshold (are significant).
During the sorting pass, the pixels in the LIP that were insignificant in the previous pass are tested, and those that emerge significant are moved to the LSP. Then, the sets are sequentially assessed along the LIS order, and when a set is found significant it is removed from the list and partitioned. The new sets with more than one element are added back to LIS, while the one element sets are added to the end of LIP or LSP, according to their being significant. The significance function is defined as follows:

$$Sn(T)=\begin{cases} 1, & \max\limits_{(i,j)\in T}\left\{\left|c_{i,j}\right|\right\}\geq 2^{n} \\ 0, & otherwise \end{cases} \qquad (1)$$

3-1 The SPIHT Algorithm

1. Initialization.
2. Sorting Pass.
3. Refinement Pass.
4. Quantization-step updates [6-8]

3.1.1 Initialization

It initializes the value of n for testing significance of pixels and constructing significance map. The LSP is set as an empty list. The LIS is initialized to maintain all pixels in the low pass subband that have descendents and hence act as roots of spatial trees. All these pixels are assigned to be of type A. LIP is initialized to contain all pixels in low pass pixels, as following:

1. n = $\left|\log_2\left(\max\left\{c_{(i,j)}\right\}\right)\right|$ where $c_{(i,j)}$ is the coefficient at position (i, j) in the image.
2. LIP = All elements in H.
3. LSP = Empty.
4. LIS = D's of Roots.

3.1.2 Sorting Pass

The purpose of the sorting pass is to manipulate the three lists (i.e. LIS, LIP and LSP) so that they are correct with respect to the current value of the magnitude threshold N. In this pass, elements of the LIP may be moved to the LSP. Elements of the LIS are decomposed as necessary: a set in the LIS may be broken into type A or type B subsets and its roots may be moved into the LIP or LSP as appropriate. Each entry of the LIP is tested for significance with respect to n. if significant, a 1 is transmitted, a sign bit representing sign of that pixel is transmitted and pixel coordinates are moved to LSP. If not, then 0 is transmitted.

1. Process LIP.

a) For each coefficient (i,j) in LIP, $Sn_{(i,j)}$ is output where $Sn_{(i,j)}$=1 when max $\left|c_{(i,j)}\right|\geq 2^{n}$ or $Sn_{(i,j)}$ = 0 for other.

b) If $Sn_{(i,j)}$=1, sign of coefficient (i,j): 0/1 is output and (i,j) is moved to the LSP.

2. Process LIS.

a) For each entry (i,j) in LIS and if the entry is of type D then output $Sn\left(D_{(I,J)}\right)$.

i) If $Sn(D_{(I,J)})$ = 1 then for each (k,l) $\in O_{(i,j)}$ output $Sn_{(k,l)}$.

ii) If $Sn_{(k,l)}$ = 1, then add (k,l) to the LSP and output sign of coefficient: 0/1 .

iii) If $Sn_{(k,l)}$ =0, then add (k,l) to the end of the LIP.

b) If type L then output $Sn(L_{(i,j)})$.

i) If $Sn(L_{(i,j)})$ =1 then add each (k,l) $\in O_{(i,j)}$ to the end of the LIS as an entry of type D and remove (i,j) from the LIS.

3.1.3 Refinement pass

The LSP contains the coordinates of the pixels that are visited in the refinement pass. For each element (i,j) in LSP except those included in the last sorting pass (i.e., with same n), output the n^{th} most significant bit of $|c_{i,j}|$.

3.1.4 Quantization Step Update

The quantization-step updates simply decrements N. That is, the magnitude threshold is decreased. The algorithm then returns to the sorting pass and continues. The algorithm can be halted if one wishes at any time; such as if the compressed data stream has reached desired size. N is decremented by 1 and the procedure is repeated from step 2 on wards as shown here in the following steps:

1. Decrement n by 1.
2. Then go back to the Significance Map Encoding Step (Sorting Pass).

3-2 Decoding

The decoder recovers the ordering from the execution path. It is easy with this scheme coding and decoding have the same computational complexity. An additional task done by decoder is to update the reconstructed image. For the value of n when a coordinate is moved to the LSP, it is known that

$$2^n \le |C_{i,j}| < 2^{n+1} \qquad (2)$$

So, the decoder uses that information, plus the sign bit that is input just after the insertion in the LSP, to set

$$C^{\wedge}_{i,j} = \pm 1.5 * 2^n \qquad (3)$$

(Where $C^{\wedge}$ represents set the reconstruction vector)

Similarly, during the refinement pass the decoder adds or subtracts 2^{n-1} to $C^{\wedge}_{i,j}$ When it inputs the bits of the binary representation of $|C_{i,j}|$. In this manner the distortion gradually decreases during the sorting and refinement passes [8].

4-Objective Quality Measures

Image quality measures (IQM) are figures of merit used for evaluation of imaging systems or of coding / processing techniques. Let $x_{(m,n)}$ denotes the samples of original image, and $x^T_{(m,n)}$ denotes the samples of compressed image. M and N are number of pixels in row and column directions respectively [3].

Mean Square Error is given by:

$$\text{MSE} = \frac{1}{MN}\sum_{m=1}^{M}\sum_{n=1}^{N}\left(x_{(m,n)} - x_{(m,n)}^{T}\right)^{2} \quad (4)$$

Peak Signal To Noise Ratio is given as:

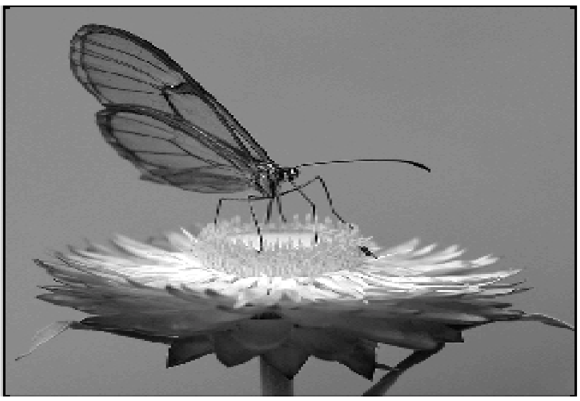
$$\text{PSNR} = 10\log_{10}\left[\frac{(255)^2}{MSE}\right] \quad (5)$$

The Maximum Difference (MD) parameter gives the maximum difference in pixel values of two images as:

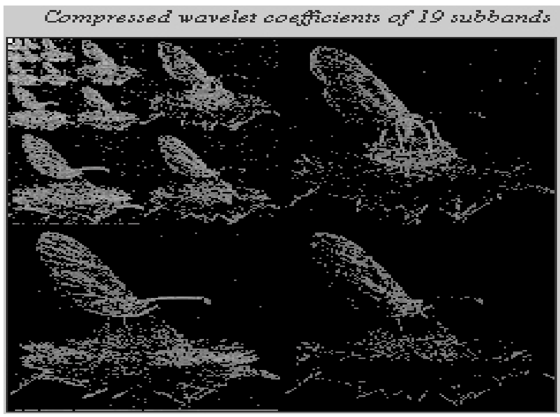
$$\text{MD} = Max\left(\left|x_{(m,n)} - x_{(m,n)}^{T}\right|\right) \quad (6)$$

5-Experimental Results

In this section, the results obtained from the experimentation to illustrate the implementation of the DWT-SPIHT coding scheme in image compression. The scheme is implemented in MATLAB platform. The DWT-SPIHT coding scheme is evaluated on natural dyadic square (512 x 512) grayscale images. The filter bank CDF 9/7 was used for decomposition image of depth 6 according to partitioning step. Figure 2 Shows Wavelet coefficients at six-level image decomposition. The wavelet coefficients are encoded with SPIHT code scheme according to SPIHT algorithm.



(a)



(b)

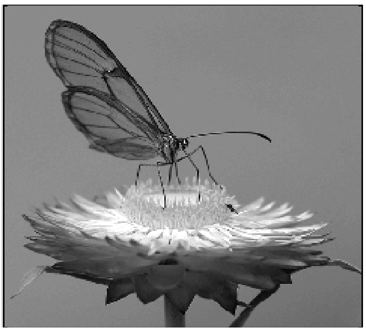
Figure 2: (a) Original image, (b) The Wavelet coefficients in the form of a decomposition image at six-level.

The experiments used six test images, code with the SPIHT image compression coder for each test image with different compression ratio (CR). Figure 3 shows the reconstructed image with CR=0.08bpp and CR=0.15bpp respectively (where bpp is bit per pixel). The reconstructed images are very good even through compression ratio is set noticeably high. The execution time is measured for encode and decode of the images. Table 1 shows the execution time for encode and decode of SPIHT algorithm. Testing was performed on Lena image 256×256 with different (CR). From the result it is obvious the encoding times are noticeably higher than decoding since a descendant checking scheme is applied. The execution time is decreased for encode and decode with compression ratio increase. That confirms SPIHT efficiency in image compression.

(a) Original Lena

(b) DWT-SPIHT (1:100 / 0.08bpp)

(c) DWT-SPIHT (1:53 / 0.15bpp)



(a) Original Butterfly

(b) DWT-SPIHT (1:100 / 0.08bpp)

(c) DWT-SPIHT (1:53 / 0.15bpp)

(a) Original Baby

(b) DWT-SPIHT (1:100 / 0.08bpp)

(c) DWT-SPIHT (1:53 / 0.15bpp)

(a) Original Cat

(b) DWT-SPIHT (1:100 / 0.08bpp)

(c) DWT-SPIHT (1:53 / 0.15bpp)

(a) Original Herkal (b) DWT-SPIHT (1:100 / 0.08bpp) (c) DWT-SPIHT (1:53 / 0.15bpp)

(a) Original Home (b) DWT-SPIHT (1:100 / 0.08bpp) (c) DWT-SPIHT (1:53 / 0.15bpp)

Figure 3: (a) Original image, (b) The reconstructed image with DWT-SPIHT, CR= 0.08bpp, (c) The reconstructed image with DWT-SPIHT, CR=0.15bpp

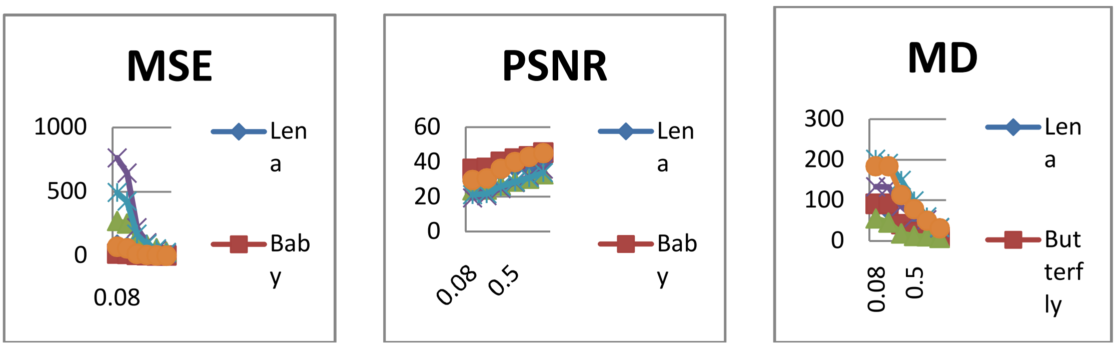
Table 1:The execution time (time in seconds) for encode and decode of SPIHT algorithm

CR	ENCODE	DECODE
100:1	1.000	0.500
80:1	1.063	0.531
40:1	1.922	1.031
20:1	3.344	1.984
10:1	6.125	3.875
8:1	6.203	3.969

The quality of the reconstructed image is measured in terms of MSE, PSNR and MD according to eq(4), eq(5) and eq(6) . Table 2: shows the quality measures for six test images with varying bit per pixel (bpp). From the experiments it is obvious the image quality different for test images at the same bpp depending on the type of image. Figure 4 shows the graphical of measures MSE, PSNR and MD are different for test images at same compression ratio.

Table 2:The quality measured for six test images with varying bpp

Quality Measures	Bit rate (bpp)	Lena	Butterfly	Baby	Cat	Herkal	Home
MSE	0.08	86.012	70.572	16.634	271.354	763.031	491.711
	0.1	67.843	56.975	14.159	252.486	644.838	425.397
	0.3	23.352	15.97	6.579	142.59	219.73	167.338
	0.5	13.51	6.488	4.271	88.194	100.183	91.19
	0.7	9.876	3.33	3.111	59.032	44.612	52.426
	1	6.576	2.005	1.9	33.216	17.109	26.642
PSNR	0.08	28.785	29.645	35.921	23.795	19.305	21.214
	0.1	29.816	30.574	36.62	24.108	20.636	21.843
	0.3	34.448	36.098	39.949	26.59	24.712	25.895
	0.5	36.824	40.01	41.825	28.676	28.123	28.531
	0.7	38.185	42.902	43.201	30.42	31.636	30.935
	1	39.951	45.11	45.344	32.917	35.799	33.875
MD	0.08	81	91	55	133	201	184
	0.1	75	91	45	132	191	184
	0.3	37	41	18	85	149	113
	0.5	28	22	12	57	98	78
	0.7	23	13	11	52	59	50
	1	14	9	7	32	33	31



(a) (b) (c)

Figure 4: (a) Graphical comparison MSE for test image at same CR.
(b) Graphical comparison PSNR for test image at same CR.
(c) Graphical comparison MD for test image at same CR.

The image quality is decrease with compression ratio increase. Example, the Baby image has low-detail, one can increase the compression ratio to 0.03bpp and the image quality is kept good. But the Home image has high –detail, the reconstructed image quality with compression ratio 0.5bpp would achieved even desirable quality.

6-Conclusions

The result showed the high efficiency of DWT-SPIHT in image compression. SPIHT is a very computationally simple algorithm and is easy to implement in comparison with other coding methods. SPIHT is fast coding and decoding, the execution time is decreased with compression ratio increased. The image quality is depending on the image type and compression ratio. The DWT-SPITH has achieved good image quality with high PSNR. High compression ratio produce low image quality and according to the image type. It is also concluded that MD is good quality measure in SPIHT compression, the MD is increased with compression ratio increased and vice versa.

References

[1] Zhai Liang , Tang Xinming, Zhang Guo, Wu Xiaoliang, "Effects of JPEG2000 and SPIHT Compression on Image Classification", The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Vol. XXXVII, Part B7, Beijing 2008.

[2] Jan Maly, Pavel Rajmic, "DWT-SPIHT Image Codec Implementation", Department of Telecommunications, Brno University of Technology, Brno, Czech Republic, 2007.

[3] Bhawna1 Rani, R K Bansal , Dr Savina Bansal," Comparison of JPEG and SPIHT Image Compression Algorithms using Objective Quality Measures", Multimedia, Signal Processing and Communication Technologies, International Publication, pp.90-93, 2009.

[4] Mohammad Pooyan, Ali Taheri, Morteza Moazami-Goudarzi, Iman Saboori,"Wavelet Compression of ECG Signals Using SPIHT Algorithm", World Academy of Science, Engineering and Technology 2, 2005.

[5] Nikola Sprljan, Sonja Grgic, Mislav Grgic**,"** Modified SPIHT algorithm for wavelet packet image coding", Real-Time Imaging 11, 378–388, 2005.

[6] Ms.Mansi Kambli and Ms.Shalini Bhatia,"Comparison of different Fingerprint Compression Techniques", Signal & Image Processing, International Journal (SIPIJ) Vol.1, No.1, September 2010.

[7] Chandan Singh D. Rawat, Sukadev Meher**,"** A Hybrid Coding Scheme Combining SPIHT and SOFM Based Vector Quantization for Effectual Image Compression", European Journal of Scientific Research ISSN 1450-216X Vol.38 No.3, pp. 425-440, 2009.

[8] Amir Said, William A. Pearlman, "A New, Fast, and Efficient Image Codec Based on Set Partitioning in Hierarchical Trees", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 6, No. 3, pp. 243-249, June 1996.

كفاءة تقسيم مجموعة في الاشجار المرتبية (SPIHT) في ضغط وجودة الصورة

م.م.ايناس مظفر جميل
قسم علوم الحاسبات- كلية التربية للبنات – جامعة بغداد

المستخلص:

ان ضغط الصورة اداة / وسيلة مهمة لتقليل عرض الحزمة و حجم الخزن المطلوبين لانظمة الصور العملية. ان الحاجة اليومية الماسة لتقليل حجم الخزن وزمن الارسال أدى الى استخدام تقنيات الضغط للصور.

هذا البحث يقدم طريقة تقسيم مجموعة في الاشجار المرتبية SPIHT لصور تعتمد على التحويلات المويجة المتقطع.

ان قياس نوعية الصورة بعد عملية الضغط تقاس باستخدام متوسط مربع الخطأ (MSE) ، و كذلك باستخدام نسبة قمة الاشارة الى الضوضاء (PSNR) والاختلاف الاقصىMD). وتعدُّ (MSE) و(PSNR) من مقاييس نوعية الصورة الأكثر استخداماً.

استخدمت في هذا البحث صور متنوعة و مختلفة ، و اجريت عليها الفحص و التقييم ، و باستخدام أنساب ضغط مختلفة. بينت نتائج البحث ان خوارزمية SPIHT كانت ذا كفاية و مقدرة عالية في ضغط الصور.