

DCSPIHT: Image Compression Algorithm

***Manik Groach**

***Dr. Amit Garg**

*Department of ECE

BGSB University

Rajouri, J&K, (India)

**Department of ECE

Maharishi Markandeshwar University

Mauilana, Haryana, (India)

Abstract

In this Research paper, we present an efficient hybrid image compression method by combining the features of two different techniques DCT and SPIHT. First the image is compressed using DCT to compress low frequency component. Subsequently, compressed image is decomposed using biorthogonal wavelet transform. The decomposed output is further compressed using SPIHT encoding for high frequency component. The reconstruction of the original image involves the linear combination of its corresponding processes as in image encoding. The reconstructed image quality achieved after decoding is of desirable quality. Experimental results show that the DCSPHIT gives better or close quality to the SPIHT.

Keywords- DCT; SPIHT; DWT; DCSPHIT.

1 . INTRODUCTION

Recently, with the rapid proliferation of multimedia contents and the advent of World Wide Web (WWW), data compression techniques have gained immense significance. Data compression has emerged as a significant technology for effectual storage and transmission of digital data [1]. Lately, digital images have been extensively made use of in many diverse fields. The enormous volume of data essential to describe these digital images makes transmission and storage sluggish and unfeasible.

Digital images like any other data necessitate compression techniques so as to conserve disk space needed for storage and time incurred for transmission [2]. The objective of image compression techniques is to eliminate redundant data present in image such that it allows for satisfactory image reconstruction [3]. Generally in image compression, an original image mapped onto a bit stream appropriate for communication over or storage in a digital medium. Obviously, the number of bits needed to represent the coded image should be lesser than that required for the original image [4]. Ever since its introduction, image compression has been one of the most exigent fields of image processing research and is taking an active part in the development of a variety of multimedia computer services and telecommunication applications like teleconferencing; digital broadcast codec, video technology and more [5].

Image compression techniques can be broadly classified into two namely: lossless and lossy. Lossless techniques warranty a perfect reconstruction of every pixel, but they are very limited in term of compression rate. Lossy techniques permit better compression rates to be attained, whilst introducing some distortion in reconstructed images. To date, so many image compression techniques have been developed: techniques based on predictive coding, transform coding, vector quantization and more. Among these, transform coding and vector quantization based techniques have received a great deal of attention. Over the past decades, the use of wavelet transforms for image compression methods has gained wide popularity. In image compression, wavelets can be defined as mathematical functions that offer very good quality compression at very high ratios owing to their capability to decompose signals into different scales or resolutions. In the literature, state of the art algorithms like JPEG and MPEG have successfully made use of traditional transforms such as the Discrete Cosine Transform (DCT).

Image compression applications benefit from the following desirable properties of wavelet transforms namely, 1) orthogonality, 2) compact support, 3) linear phase and high approximation/vanishing moments of the basis function, 4) efficient multi-resolution representation, 5) scalability, and 6) embedded coding with progressive transmission. The general procedure involved in wavelet transform-based image compression techniques is, first the image data is decorrelated by applying a wavelet transform, then the

resulting transform coefficients are quantized and the quantized values are coded. Many wavelet-based image coders such as Embedded Zero-tree wavelets (EZW) [8], Set Partitioning In Hierarchical Trees (SPIHT) [9], Morphological Representation of Wavelet Data (MRWD), Group Testing for Wavelets (GTW) and Modulated Wavelet Subband Image Coding (MWSIC) have been proposed and are of a great success. Of most algorithms developed, SPIHT algorithm ever since its introduction in 1996 for image compression has received a lot of attention. For still image compression, SPIHT achieves considerably better quality when compared to vector quantization, JPEG, and wavelets combined with quantization techniques, while not requiring training and producing an embedded bit stream.

Rani, B *et al.* [11] have conducted a comparative study of JPEG and SPIHT compression algorithms. A set of objective picture quality measures such as 1) peak signal to noise ratio (PSNR), 2) maximum difference (MD), 3) least mean square error (LMSE), 4) structural similarity index (SSIM) and 5) picture quality scale (PQS) were utilized to measure the picture quality. The comparison was carried out on the basis of the results of these quality measures. SPIHT based compression accomplishes superior results when compared to JPEG for all types of compressions.

In the aspect of improving the code quality, Discrete Cosine Transform has enormous natural advantages. Images compressed by Discrete Cosine Transform, compression ratio can reach tens of thousands to one, and decoding is almost real-time operation, when decoding we can be arbitrary scale the size, and maintain the fine structure of the edge. In this paper, while given full consideration to the characteristics of human vision, we combine the two algorithms, that is Discrete Cosine Transform and SPIHT algorithm.

2. CODING AND ALGORITHM ANALYSIS.

2(a) Discrete Cosine Transform.

A Discrete Cosine transform expresses a sequence of finitely many data points in terms of a sum of cosine functions oscillating at different frequencies. DCT's are important to numerous applications in science and engineering, from lossy compression of audio and images. The use of cosine rather than sine functions is critical in these applications, for compression it turns out that cosine functions are much more efficient.

The 1D DCT is defined as

$$D_x\{f\}(\omega) = \alpha(\omega) \sum_{x=0}^{N-1} f(x) \cos\left[\frac{(2x+1)\omega\pi}{2N}\right]; \quad \alpha(\omega) \begin{cases} \sqrt{\frac{1}{N}} & \omega = 0 \\ \sqrt{\frac{2}{N}} & \text{else} \end{cases} \quad (\dots 1)$$

which is similar to the DFT:

$$F(\omega) = \frac{1}{N} \sum_{x=0}^{N-1} f(x) e^{-\frac{j2\pi x\omega}{N}} \quad (\dots 2)$$

2D DCT is defined using the separability property as 1D transform on the rows and on the columns, applied separately

$$F(u,v) = D_y\{D_x\{f(x,y)\}\} \quad (\dots 3)$$

One of the advantages of DCT is the fact that it is a real transform, whereas DFT is complex. This implies lower computational complexity, which is sometimes important for real-time applications.

2(b) Discrete wavelet Transform.

A wavelet, in the sense of the Discrete Wavelet Transform (or DWT), is an orthogonal function which can be applied to a finite group of data. Functionally, it is very much like the Discrete Fourier Transform, in that the transforming function is orthogonal, a signal passed twice through the transformation is unchanged, and the input signal is assumed to be a set of discrete-time samples. Both transforms are convolutions.

Whereas the basis function of the Fourier transform is a sinusoid, the wavelet basis is a set of functions which are defined by a recursive difference equation

$$\phi(x) = \sum_{k=0}^{M-1} c_k \phi(2x - k), \quad (\dots 4)$$

Where the range of the summation is determined by the specified number of nonzero coefficients M. The number of nonzero coefficients is arbitrary, and will be referred to as the order of the wavelet. The value of the coefficients is, of course, not arbitrary, but is determined by constraints of orthogonality and normalization. Generally, the area under the wavelet "curve" over all space should be unity, which requires that

$$\sum c_k = 2 \quad (\dots 5)$$

Equation (4) is orthogonal to its translations; i.e., $\int \phi(x)\phi(x;k)dx = 0$. It is also desired that the equation is orthogonal to its dilations, or scales; i.e., $\int \phi(x)\phi(2x ; k)dx = 0$.

Such a function ϕ does exist, and is given by

$$\phi(x) = \sum (-1)^k c_{1-k} \phi(2x - k) \quad (\dots 6)$$

It is dependent upon the solution of $\phi(x)$. Normalization requires that,

$$\sum c_k c_{k-2m} = 2\delta_{0m} \quad (\dots 7)$$

It means that the above sum is zero for all m not equal to zero, and that the sum of the squares of all coefficient is two. Another important equation which can be derived from the above conditions and equations is

$$\sum (-1)^k c_{1-k} c_{k-2m} = 0 \quad (\dots 8)$$

A good way to solve for values of equation (4) is to construct a matrix of coefficient values. This is a square MxM matrix where M is the number of nonzero coefficient. The matrix is designated L, with entries $L_{ij} = c_{2i-j}$. This matrix always has an Eigen value equal to 1, and its corresponding (normalized) eigenvector contains, as its components, the value of the ϕ function at integer values of x. Once these values are known, all other values of the function $\phi(x)$ can be generated by applying the recursion equation to get values at half-integer x, quarter-integer x, and so on down to the desired dilation. This effectively determines the accuracy of the function approximation.

This class of wavelet functions is constrained, by definition, to be zero outside of a small interval. This is what makes the wavelet transform able to operate on a finite set of data, a property which is formally called "compact support." Most wavelet functions, when plotted, appear to be extremely irregular. This is due to the fact that the recursion equation assures that a wavelet ϕ function is non differentiable everywhere. The functions which are normally used for performing transforms consist of a few sets of well-chosen coefficient resulting in a function which has a discernible shape. Two of these functions are shown in Figure 1 the first is the Haar basis function, chosen because of its simplicity for the following discussion of wavelets, and the second is the Daubechies-4 wavelet, chosen for its usefulness in data compression.

Table 1: Coefficients for three named wavelet functions

Wavelet	c_0	c_1	c_2	c_3	c_4	c_5
Haar	1.0	1.0				
Daubechies-4	$\frac{1}{4}(1 + \sqrt{3})$	$\frac{1}{4}(3 + \sqrt{3})$	$\frac{1}{4}(3 - \sqrt{3})$	$\frac{1}{4}(1 - \sqrt{3})$		
Daubechies-6	0.332671	0.806891	0.459877	-0.135011	-0.085441	0.035226

They are named for pioneers in wavelet theory [6] the nonzero coefficients c_k which determine these functions are summarized in Table 1. Coefficients for the Daubechies-6 wavelet, one used in the discussion of the wavelet transformer hardware implementation, are also given in Table 1.

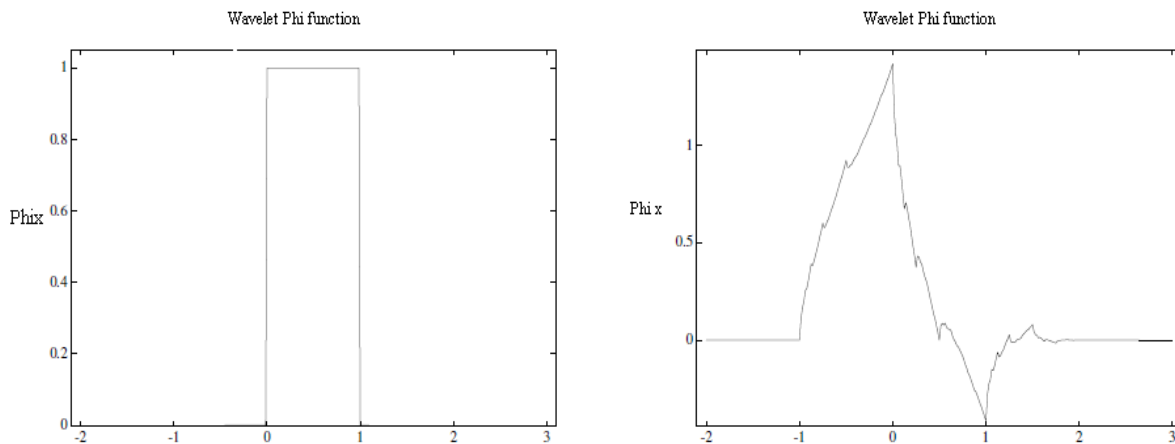


Figure 1: The Haar and Daubechies-4 wavelet basis functions

The Mallat "pyramid" algorithm [7] is a computationally efficient method of implementing the wavelet transform, and this is the one used as the basis of the hardware implementation described in Section 3. The lattice filter is equivalent to the pyramid algorithm except that a different approach is taken for the convolution, resulting in a different set of coefficients, related to the usual wavelet coefficients c_k by a set of transformations.

2(c) SPIHT Encoding

SPIHT is one of the most advanced schemes available that outperforms even the state-of-the-art JPEG 2000 in some situations. The Set-Partitioning in Hierarchical Trees (SPIHT) [9] coding operates by exploiting the relationships among the wavelet coefficients across the different scales at the same spatial location in the wavelet sub bands. In general, SPIHT coding involves the coding of the position of significant wavelet coefficients and the coding of the position of zero trees in the wavelet sub bands.

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 we move from the highest to the lowest levels of the sub band pyramid
- 2) It has been observed that there is a spatial self-similarity among the sub bands, and the coefficients are likely to be better magnitude-ordered if we move downward in the pyramid along the same spatial orientation.

A tree structure, termed spatial orientation tree, clearly describes the spatial relationship on the hierarchical pyramid. Fig. 2 shows how the spatial orientation tree is defined in a pyramid constructed with recursive four-sub band splitting. Every pixel in the image signifies a node in the tree and is determined by its corresponding pixel coordinate. Its direct descendants (offspring) symbolize the pixels of the same spatial orientation in the next finer level of the pyramid. The tree is defined in such a manner that each node has either no offspring (the leaves) or four offspring's, which at all times form a group of 2×2 adjacent pixels. In Fig. 2, the arrows are directed from the parent node to its four offspring's. The pixels in the highest level of the pyramid are the tree roots and are also grouped in 2×2 adjacent pixels. Nevertheless, their offspring branching rule is different, and in each group, one of them has no descendants.

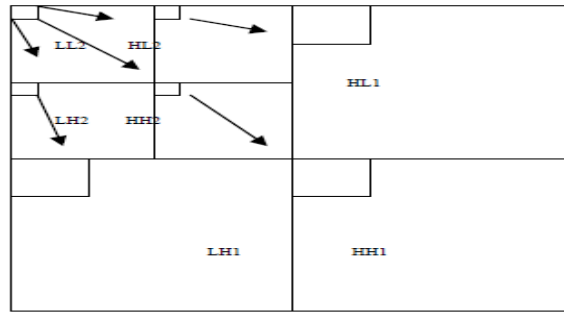


Figure 2: Tree structure used in SPIHT

The following are the sets of coordinates used to represent the coding method:

$O(i, j)$ in the tree structures is the set of offspring (direct descendants) of a tree node defined by pixel location (i, j) .

$D(i, j)$ is the set of descendants of node defined by pixel location (i, j) .

$L(i, j)$ is the set defined by $L(i, j) = D(i, j) - O(i, j)$.

Except for the highest and lowest pyramid levels, the set partitioning trees has,

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

The rules for splitting the set (e.g. when found significant),

- 1) The initial partition is formed with the sets (i, j) and $D(i, j)$, for all $(i, j) \in CH$.
- 2) If $D(i, j)$ is significant, then it is partitioned into $L(i, j)$ plus the four single-element sets with $(k, l) \in O(i, j)$.
- 3) If $L(i, j)$ is significant, then it is partitioned into the four sets $D(k, l)$ with $(k, l) \in O(i, j)$. The significant values of the wavelet coefficients modeled in the spatial orientation tree are stored in three ordered lists namely,
 - 4) **LIS, List of insignificant sets:** 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).
 - 5) **LIP, List of insignificant pixels:** contains the individual coefficients that have magnitude smaller than the threshold.
 - 6) **LSP, List of significant pixels:** 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:

$$S_n(T) = \begin{cases} 1, & \max\{c_{i,j}\} \geq 2^n \\ & (i,j) \in T \\ 0, & \text{otherwise} \end{cases} \quad (\dots 9)$$

Algorithm

1). **Initialization:** output $n = \lceil \log_2(\max_{(i,j)} \{|c_{i,j}|\}) \rceil$

Set the LSP as an empty list, and add the coordinates $(i, j) \in H$ to the LIP, and only those with descendants also to the LIS, as type A entries.

2). **Sorting Pass:**

2.1.) 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 this 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 (k, l) 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 $L(i, j) \neq 0$ then move (i, j) to the end of the LIS, as an entry of type B, and go to Step 2.2.2); otherwise, remove entry (i, j) 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 an entry of type A;
 - remove (i, j) from the LIS.

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

4). **Quantization-Step Update:** decrement n by 1 and go to Step 2.

Some of the advantages of SPIHT encoding include:

- (i) Allows a variable bit rate and rat distortion control as well as progressive transmission
- (ii) An intensive progressive capability – we can interrupt the decoding (or coding) at any time and a result of maximum possible detail can be reconstructed with one-bit precision.
- (iii) Very compact output bit stream with large bit variability no additional entropy coding or scrambling has to be applied.

3. DCSPIHT ALGORITHM FOR IMAGE COMPRESSION

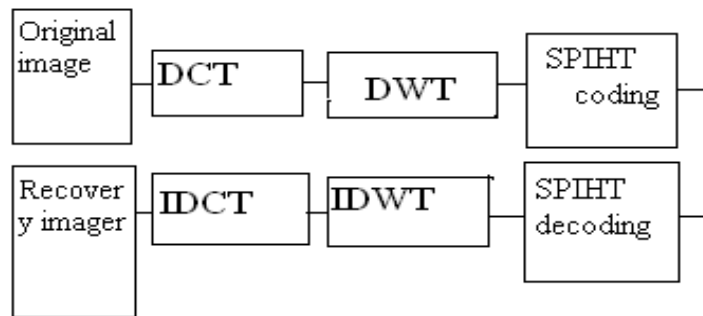


Figure 3: DCSPIHT algorithm coding /decoding diagram

Human eye sensitivity to different frequencies is different, it is not sensitive to gray-scale error, but the human eye is particularly sensitive to the image edge features such as, through the SPIHT algorithm to improve transformation process; increase the edge threshold, the human visual characteristics and SPIHT algorithm pay more attention to image edge information. At the same time, the DCT coding and SPIHT algorithm are combined to achieve hybrid DCT and SPIHT coding.

This DCSPIHT algorithm combines two different techniques DCT and SPIHT to achieve better image compression because every image consists of low frequency and high frequency component. As we know DCT is the technique which is more efficient for low frequency component and SPIHT gives better result for high frequency component.

As see in figure 3, first of all original image is pass through the DCT coding after that we create the wavelet transformation of DCT output, this dwt output then encoded with SPIHT technique, now overall coded data is to be transmitted. In the receiver side received data is to be decoded.

4 . EXPERIMENTAL RESULTS

This section, present the results obtained from the experimentation to illustrate the effectiveness of the proposed hybrid coding scheme in image compression. The presented scheme is implemented in MATLAB (Matlab 7.8). The proposed hybrid coding scheme is evaluated on natural dyadic square (512 x 512) grayscale images. The results obtained are compared to the SPIHT. The test images used in the experiments include: Lena. The quality of the reconstructed images is measured in terms of PSNR value and the efficiency of the presented hybrid scheme is measured in terms of its compression ratio. The PSNR in decibels (dB) is defined as:

$$PSNR = 20 \log_{10} \frac{MAX}{RMSE} = 20 \log_{10} \frac{255}{\sqrt{\frac{1}{NM} \sum_{x=1}^N \sum_{y=1}^M [f(x,y) - f'(x,y)]^2}}$$

Where, MAX is the image depth. N and M are the width and height of the image in pixels. f is the original image and f' is the reconstructed image.

For Lena test image, the original image and the reconstructed image obtained using the proposed hybrid coding scheme are given in Figure 4. The comparison of performance measures (compression rate measured in PSNR) of the proposed hybrid coding scheme against results of SPIHT coding are presented in Table II.

Table II: Comparison of performance measures of (a) DCSPIHT compression scheme (b) SPIHT

Image	Bit rate	PSNR (dB)	
		DCSPIHT	SPIHT
LENA	0.25	31.40	31.805
	0.50	35.16	35.045
	0.65	36.60	36.456



Figure 4: (a) Original image (b) Reconstructed image

5. CONCLUSION

In this research paper, we have proposed an efficient hybrid image compression scheme combining Discrete Cosine Transform and Set Partitioning in Hierarchical Trees (SPIHT) coding for effectual compression of images. Initially, the input image has been subjected to DCT coding then the output is decomposed using biorthogonal wavelet transform. This decomposed output further compressed using SPIHT encoding. The experimental results have portrayed the effectiveness of the proposed hybrid scheme in image compression.

References

- [1] Laha. A., Pal.N.R. and Chanda, B. Oct. 2004. "Design of Vector Quantizer for image compression using Self Organizing Feature Map and Surface Fitting", IEEE Transactions on Image Processing, Vol. 13, No. 10, pp. (1291-1303).
- [2] Yao Z. and Baozong Y., March 1996. "A hybrid image compression scheme combining block based fractal coding and DCT," Signal Processing: Image Communication, Vol. 8, No. 2, and pp. (73- 78).
- [3] Curtis, S.E. and C.E. Martin, 2005. "Functional fractal image compression," Proceedings of 6th Symposium on Trends in Functional Programming, TFP 2005, September 23-24, Tallinn, Estonia, pp. (383-398).
- [4] Cosman, P.C. Gray, R.M. and Vetterli, M., Feb 1996. "Vector quantization of image subbands: a survey", IEEE Transactions on Image Processing, Vol. 5, No. 2, pp. (202-225).
- [5] N.A. Koli and M.S. Ali, 2008."A Survey on Fractal Image Compression Key Issues", Journal of Information Technology, Vol. 7, No. 8, pp. (1085-1095).
- [6] Daubechies, "Orthonormal Bases of Compactly Supported Wavelets," Comm. Pure Applied Mathematics, vol. 41, pp. (909-996), 1988
- [7] Mallat, S. "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 11, pp. (674-693), 1989.
- [8] Shapiro, J. M.: Embedded image coding using zero trees of wavelet coefficients. IEEE Transactions on Signal Processing, 1993, Vol. 41(12) pp. (3445-3462).
- [9] Said, A., and Pearlman, W.A., 1996. "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. (243-250).
- [10] S. Esakkirajan, T. Veerakumar, V. Senthil Murugan and P. Navaneethan, 2008. "Image Compression using Multiwavelet and Multi-stage Vector Quantization," International Journal of Signal Processing (IJSP), Vol. 4, No.4, pp. (246 – 253).
- [11] Rani, B. Bansal, R.K. Bansal, S., 2009. "Comparison of JPEG and SPIHT image compression algorithms using objective quality measures", Multimedia, Signal Processing and Communication Technologies, Aligarh, pp. (90 – 93).
- [12] Sherman D. Riemenschneider and Zuwei Shen, 1997. "Construction of Compactly Supported Biorthogonal Wavelets: I", Adv. Comput. Math., pp. (81-104).
- [13] Richard E. Ewing, Jianguo Liu and Hong Wang, January 2004. "Adaptive biorthogonal spline schemes for advection-reaction equations", Journal of Computational Physics, Vol. 193, No. 1, pp. (21-39).
- [14] Chunlin Song, Rui Feng, Wei Jin, Aihuang GUO. Improved Multiresolution SPIHT Algorithm. Computer Engineering. Vol.34(4) 2008, pp.(241-243).