Volume 4, Issue 6, June 2014 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Image Compression Technique under JPEG by Wavelets Transformation

Bhawna Gupta
M-Tech Student,
IEC college, Greater Noida,
UP, India

Manju Gupta
M-Tech Student,
IEC college, Greater Noida,
UP, India

Banita Chadha
Assistant Professor
IEC college, Greater Noida,
UP, India

Abstract— It is used specially for the compression of images where tolerable degradation is required. With the wide use of computers and consequently need for large scale storage and transmission of data, efficient ways of storing of data have become necessary. With the growth of technology and entrance into the Digital Age, the world has found itself amid a vast amount of information. Dealing with such enormous information can often present difficulties. Image compression is minimizing the size in bytes of a graphics file without degrading the quality of the image to an unacceptable level. The reduction in file size allows more images to be stored in a given amount of disk or memory space. It also reduces the time required for images to be sent over the Internet or downloaded from Web pages.JPEG and JPEG 2000 are two important techniques used for image compression.JPEG image compression standard use dct (discrete cosine transform). Now there wavelets Transform is using with JPEG 2000 standard. It is a widely used and robust method for image compression. It has excellent compaction for highly correlated data. Wavelets transform divided the image into high frequency components. Which gives good compromise between information packing ability and computational complexity?

Keywords — JPEG, DCT, compression.

I. INTRODUCTION

Image compression is an application of data compression that encodes the original image with few bits. The objective of image compression is to reduce the redundancy of the image and to store or transmit data in an efficient form. The main goal of such system is to reduce the storage quantity as much as possible, and the decoded image displayed in the monitor can be similar to the original image as much as can be compression with acceptable visual quality for decoded images. There has always been intense interest in development of efficient image compression algorithms. A lot of work in image compression has focused on transform coding. The transform coders are designed to remove the redundancy in images for purposes of bit rate reduction, based upon signal processing and information theory.

The JPEG standard is one of the most widely known standards for lossy image compression. The approach recommended by the JPEG is a transform coding approach using the Discrete Cosine Transform (DCT) of 8´8 sub-blocks4. Nevertheless, for the DCT, the blocks do not fit the boundaries of the real objects in the image scene, leading to visually annoying blocking artifacts, especially for compression at low bit rates. A new class of transformation called the wavelets which is hybrid with JPEG to give the better result in aspect of space as well as time execution. A fast computable approximation to the JPEG with wavelets has been designed to save the computation cost. Therefore, the hybrid algorithms are a good candidate when picking a transformation for image compression.

II. METHODOLOGY

We will implement said technique using MATLAB; MATLAB (Matrix laboratory) is an interactive software system for numerical computations and graphics. As the name suggests, Mat lab is especially designed for matrix computations: solving systems of linear equations, computing eigenvalues and eigenvectors, factoring matrices, and so forth. In addition, it has a variety of graphical capabilities, and can be extended through programs written in its own programming language. Many such programs come with the system; a number of these extend Mat lab's capabilities to nonlinear problems, such as the solution of initial value problems for ordinary differential equations. When working with images in Mat lab, there are many things to keep in mind such as loading an image, using the right format, saving the data as different data types, how to display an image, conversion between different image formats, etc. This worksheet presents some of the commands designed for these operations. Most of these commands require you to have the Image processing tool box installed with Mat lab. To find out if it is installed, type "ver." at the MATLAB prompt. This gives you a list of what tool boxes that are installed on your system. But our focus is on only image processing toolbox for implement the lapped orthogonal transform and discrete cosine transformation to perform the compression operation with image.

III. OBJECTIVES:

A new compression scheme will developed for images. The proposed compression scheme combines wavelet transform with JEPG. The wavelets are calculated for each elemental image and the elemental images are stacked. The image quality obtained with the presented technique is compared with other hybrid technique. And using the inverse transformation image will decompress. And quality of decompressed image can be judge using quality parameter. There are a variety of ways in which different compression algorithms can be evaluated and compared. For quantifying the error between images, like PSNR, SNR, and CRR. Compression is used in a wide-ranging variety of applications, from the transmission of science data collected on board NASA space probes, to the storage of digital music on personal computers. Almost as long as there has been digital data, there has been compression of that data. The basic aim of compression is to sacrifice time and/or processing power in exchange for a reduction in storage requirements.

Despite all the advantages of JPEG compression schemes based on wavelets namely simplicity, satisfactory performance, and availability of special purpose hardware for implementation; these are not without their shortcomings. Since the input image needs to be ``blocked," correlation across the block boundaries is not eliminated.
The main aims of thesis are to achieve image compression and decompression using transformation, and also study the various parameters SNR, Entropy etc.

In image compression, transform is indented to de-correlate the input pixels. Selection of proper transform is one of the important issues in image compression schemes. The transform should be selected in such a way that it reduces the size of the resultant data set as compared to the source data set. Few transformations reduce the numerical size of the data items that allow them to represent by fewer binary bits. The technical name given to these methods of transformation is mapping [28]. Some mathematical transformations have been invented for the sole purpose of data compression; others have been borrowed from various applications and applied to data compression. These include the Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT) [30], Walsh-Hadamard Transform (WHT), Hadamard-Haar Transform (HHT), Karhune-Loeve Transforms (KLT), Slant-Haar Transform (SHT), Short Fourier Transforms (SFT), and Wavelet Transforms (WT) [10]. Transform selection process still remains an active field of research.

IV. TRANSFORMATION SELECTION

JPEG stands for the Joint Photographic Experts Group, a standards committee that had its origins within the International Standard Organization (ISO).JPEG provides a compression method that is capable of compressing continuous-tone image data with a pixel depth of 6 to 24 bits with reasonable speed and efficiency.JPEG may be adjusted to produce very small, compressed images that are of relatively poor quality in appearance but still suitable for many applications. Conversely, JPEG is capable of producing very high-quality compressed images that are still far smaller than the original uncompressed data. JPEG is primarily a lossy method of compression.JPEG was designed specifically to discard information that the human eye cannot easily see. Slight changes in color are not perceived well by the human eye, while slight changes in intensity (light and dark) are. Therefore JPEG's lossy encoding tends to be more frugal with the gray-scale part of an image and to be more frivolous with the color [21].DCT separates images into parts of different frequencies where less important frequencies are discarded through quantization and important frequencies are used to retrieve the image during decompression. Compared to other input dependent transforms, DCT has many advantages: (1) It has been implemented in single integrated circuit; (2) It has the ability to pack most information in fewest coefficients; (3) It minimizes the block like appearance called blocking artifact that results when boundaries between sub-images become visible [11].

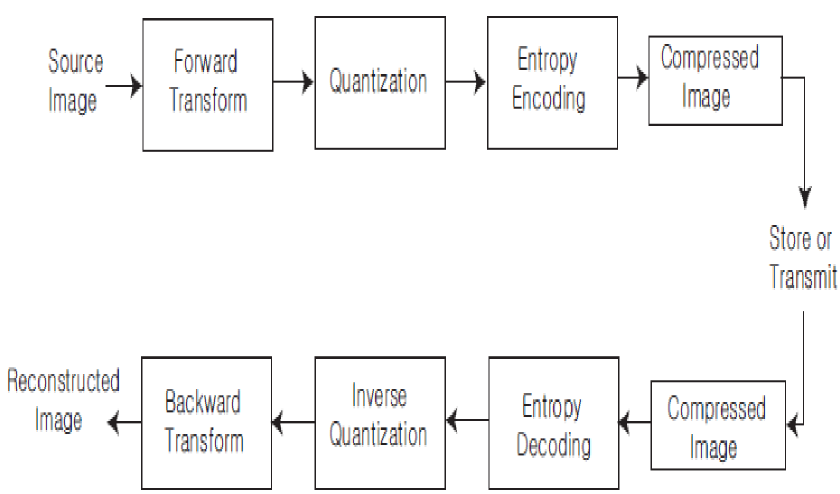


Figure 1: Image Reconstructed Model

In transform coding systems, the input signal is typically divided into blocks, which are then subjected to an energy preserving unitary transformation. The aim of the transformation is to convert statistically dependent pixels into a set of essentially independent transform coefficients, preferably packing most of the signal energy into a minimum number of coefficients, preferably packing most of the signal energy into a minimum number of coefficients. The resulting transform coefficients are quantized, coded, and transmitted. At the receiver, the signal is recovered by computing the inverse transformation after decoding and de-quantizing the transmitted data.

The DCT transform coding is widely used such as in the JPEG standard. However, the basic functions of the DCT have abrupt changes at the endpoints of their supports, which cause one of the main problems of the DCT, blocking effects especially at low bit rates. These effects are perceived in reconstructed images as visible discontinuities, or artifacts, at the cross-block boundaries. In order to avoid this, we should choose basis functions without abrupt changes. Some approaches have been introduced to reduce blocking effects, such as overlapping and filtering10, 11. However, the overlapping method increases bit rates for coding, and the filtering method blurs images at the cross block regions. A more successful method is a lapped transform for block signal coding.

V. WAVELETS TRANSFORM

The objective of the wavelet transform is to decompose the input signal into components that are easier to deal with, have special interpretations, or have some components that can be threshold away, for compression purposes. We want to be able to at least approximately reconstruct the original signal given these components.

The basic functions of the wavelet transform are localized in both time and frequency. There are two types of wavelet transforms: the continuous wavelet transforms (CWT) and the discrete wavelet transforms (DWT).

VI. 2D WAVELET TRANSFORMATION:

For an N by N input image, the two-dimensional DWT proceeds as follows:

A) Convolve each row of the image with h0 [n] and h1 [n], discard the odd numbered columns of the resulting arrays, and concatenate them to form a transformed row.

B) After all rows have been transformed, convolve each column of the result with h0 [n] and h1 [n]. Again discard the odd numbered rows and concatenate the result.

After the above two steps, one stage of the DWT is complete.
The transformed image now contains four sub band LL, HL, LH, and HH, standing for low-low, high-low, etc.
The LL sub band can be further decomposed to yield yet another level of decomposition. This process can be continued until the desired number of decomposition levels is reached.

VII. ALGORITHM:

COMPRESSED ALGORITHMS:

Step 1: Read the input image and check the size and dimension of image before compression (1.bmp)
Step 2: Apply the wavelets transformation (db4, db5, db6, db7) and divided the image into four sections.
Step 3 Extract the frequency components form the transformed image.
Step 4: set the quality parameter value for JPEG compression write it in JPEG.
Step 5: Find out the difference between original image and JPEG image.
Step 6: Apply the quantization with extracted image, and translate the matrix into array.
Step 7: Find out the Key Generation and apply the Encryption concept.
Step 8: Apply the arithmetic (arithenco).
Step 9: Save the header data as well as compressed image. (1.bha)

DECOMPRESSED ALGORITHMS:

Step 1: Read The compressed image (1.bha)
Step 2: Extract the header information from the saved data.
Step 3: Apply the arithmetic decoding using (arithdeco).
Step 4: Apply the decryption decoding for offset the effect of Step 8 of compression algorithms.
Step 5: Convert array to matrix and apply inverse wavelets to retrieve the original image.
Step 6: Save as well as Display the decompressed image

VIII. COMPRESSION TOOL:

Using above proposed algorithms designed a tool in MATLAB and run on MATLAN command, interface is shown in below figure.

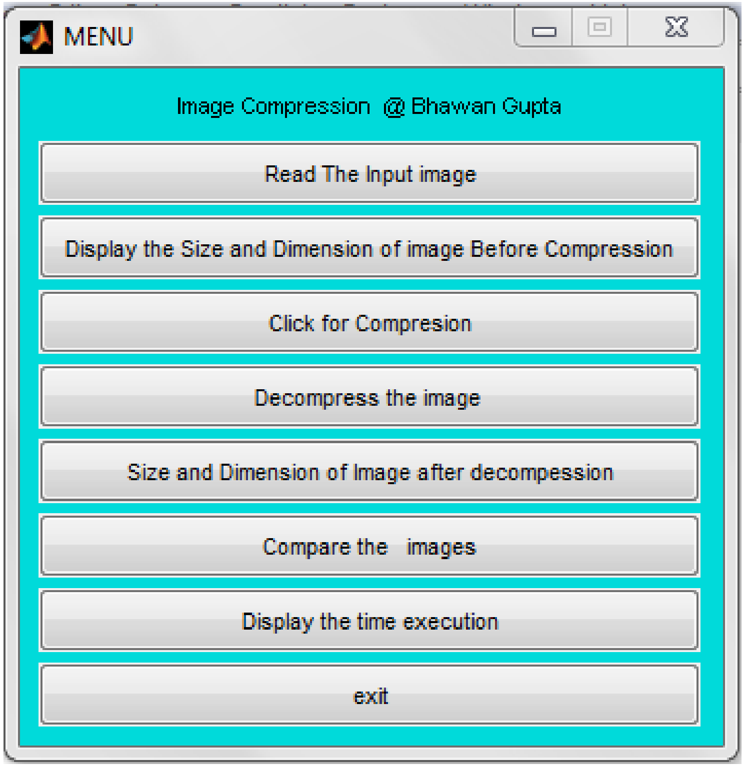


Figure 2: GUI of Compression Model

5.6 Compression with Wavelets:

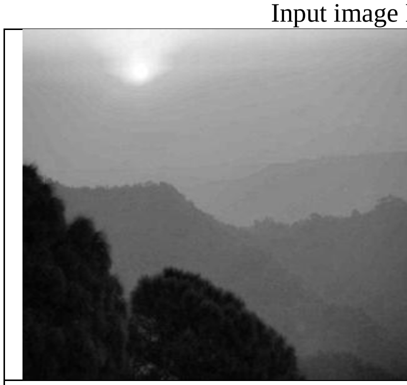
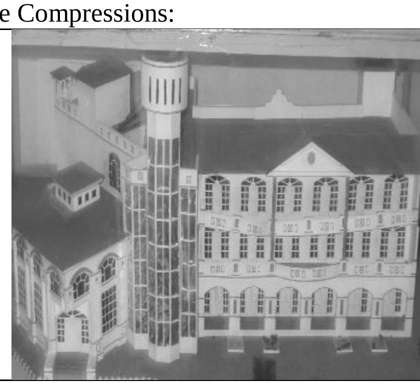
Wavelets "JPEG_db4":

Run the above algorithm with the said tool and record the size before and after the compression and save the image with extension ".bha". Calculate the compression ratio with input image 1.bmp, 2.bmp, 3.bmp and 4.bmp. Results are showing by the Table 1.

Table 1: Compression Details with 'db4'

S.No	Input image	Dimension	Size before compression	After compression (*.bha)
1	1.bmp	644X480	301 KB	5.96 KB
2	2.bmp	604X287	507 KB	6.08 KB
3	3.bmp	756X567	1.22 MB	16.4 KB
4	4.bmp	512X512	257 KB	5.08 KB

Input image Before Compressions:

	
1.bmp	2.bmp

	
3.bmp	4.bmp

Figure 3: Input Image

COMPRESSION RATIO WITH 'JPEG_DB4:

An image 1.bmp is 301 KB before compression and 5.96 KB after compression what are the values of compression ratio and saving parentage.

Compression ratio= 301/5.96
=50.5033

The interpretation is that 50.5033 pixels of input image are expressed as 1 pixel in the output image.

Saving Percentage = 1-1/50.5033
=98.0199%

An image 2.bmp is 507 KB before compression and 6.08KB after compression what are the values of compression ratio and saving parentage.

Compression ratio= 507/6.08
=83.3881

The interpretation is that 83.381 pixels of input image are expressed as 1 pixel in the output image.

Saving Percentage =1-1/83.3881
=98.8079 %

An image 3.bmp is 1.22 MB before compression and 16.04 KB after compression what are the values of compression ratio and saving parentage.

Compression ratio= (1.22×1024)/16.04
=77.88528

The interpretation is that 77.88528 pixels of input image are expressed as 1 pixel in the output image.

Saving Percentage =1-1/77.88528
=98.7160%

An image 4.bmp is 257KB before compression and 5.08 Kb after compression what are the values of compression ratio and saving parentage.

Compression ratio= 257/5.08
=50.5905

The interpretation is that 50.5905 pixels of input image are expressed as 1 pixel in the output image.

Saving Percentage =1-1/50.5905
=98.0233 %

Table 2: Result of compression ratio with db4

Input Image	Compression ratio
1.bmp	98.0199
2.bmp	98.8079
3.bmp	98.7160
4.bmp	98.0233

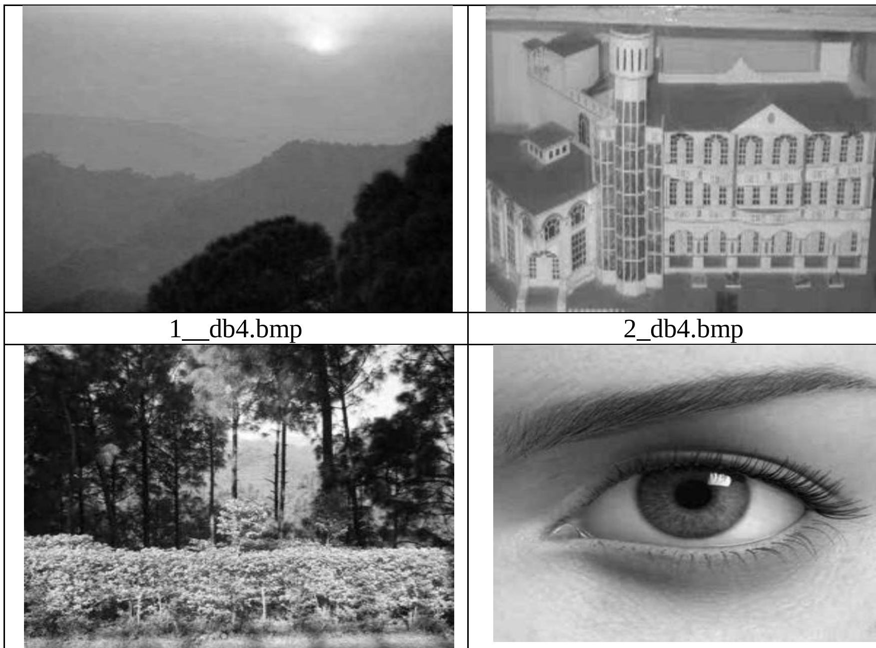
AVERAGE COMPRESSION RATIO: 98.39177 %

DECOMPRESSION WITH 'JPEG_DB4':

Read the compressed image (*.bha) and run the decompression algorithms and maintain the size of input image and decompressed. Compressed images are 1_db4.bha, 2_db4.bha, 3_db4.bha and 4_db4.bha,

Table 3: Decompression details with 'JEG_db4'

S. no	input image	Size	Size after decompression	Dimension	Name of Decompressed image
1	1_db4.bha	5.96 KB	301 KB	640X480	1_D_db4.bmp
2	2_db4.bha	6.08 KB	170 KB	604X288	2_D_db4.bmp
3	3_db4.bha	16.4 KB	420 KB	756X568	3_D_db4.bmp
4	4_db4.bha	5.08 KB	257 KB	512X512	4_D_db4.bmp



1__db4.bmp | 2_db4.bmp | 3_db4.bmp | 4_db4.bmp

Figure 4: Decompression Image with db4

Compression with 'JPEG_db5':

Run the above algorithm with the said tool and with wavelets "db5"and record the size before and after the compression and save the image with extension ".bha". And find out the how much time is required to compress the image in second. Calculate the compression ratio with input image 1.bmp, 2.bmp, 3.bmp and 4.bmp. All results shown in Table 4

Table 4: compression details with 'JEPG_db5'

Sino	Input image	Dimension	Size before compression	After compression
1	1.bmp	644X480	301 KB	6.04 KB
2	2.bmp	604X287	507 KB	6.17 KB
3	3.bmp	756X567	1.22 MB	17.0 KB
4	4.bmp	512X512	257 KB	5.10 KB

Result for wavelets 'JPEG_db5':

Table 5: Result of Compression Ratio with db5

Input Image	Compression ratio
1.bmp	97.9933
2.bmp	98.7822
3.bmp	98.6447
4.bmp	98.0155

Average compression ratio: 98.3589%

Decompression with 'JPEG_db5':

Read the compressed image (*.bha) and run the decompression algorithms with wavelets "db5" and maintain the size of input image and decompressed. Compressed images are 1_db5.bha, 2_db5.bha, 3_db5.bha and 4_db5.bha,

Table 6: Decompression Details with 'db5'

S.no	input image	Size	Size after decompression	Dimension	Decompressed image
1	1_db5.bha	6.04 KB	301 KB	640X480	1_d_db5.bmp
2	2_db5.bha	6.17 KB	507 KB	604X288	2_d_db5.bmp
3	3_db5.bha	17.0 KB	420 KB	756X568	3_4_db5.bmp
4	4_db5.bha	5.10 KB	257 KB	512X512	4_d_db5.bmp

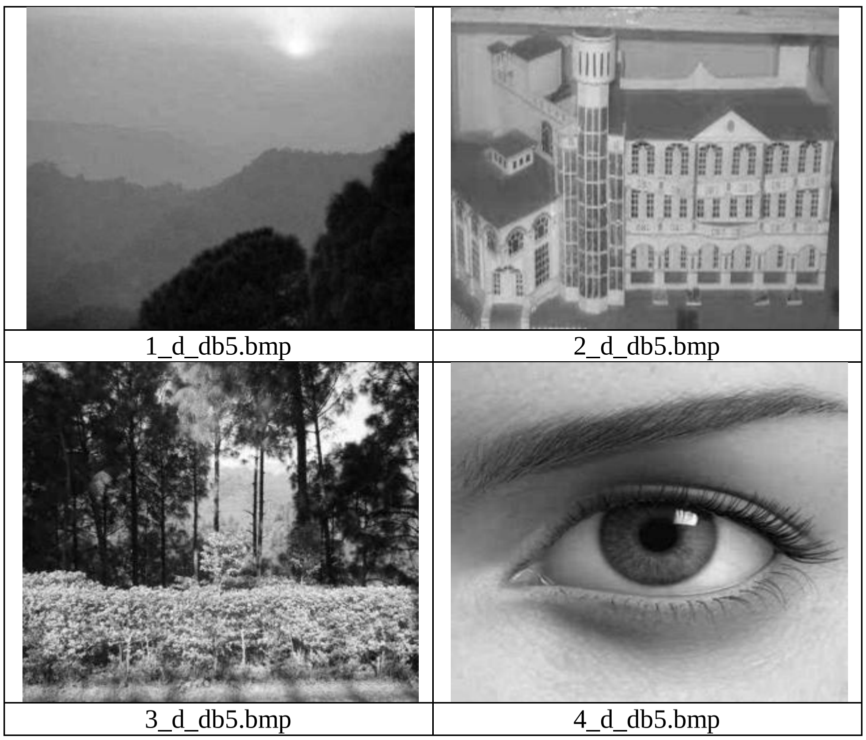


Figure 1: Decompression image with db5

Compression with 'JPEG_db6':

Run the above algorithm with the said tool with wavelets "db6" and record the size before and after the compression and save the image with extension ".bha". And find out the how much time is required to compress the image in second. Calculate the compression ratio with input image 1.bmp, 2.bmp, 3.bmp and 4.bmp. All results shown in Table 7.

Table 7: Compression Details with 'db6'

S.no	Input age	Dimension	Size before compression	After compression
1	1.bmp	644X480	301 KB	4.81 KB
2	2.bmp	604X287	507 KB	6.00 KB
3	3.bmp	756X567	1.22 MB	16.9 KB
4	4.bmp	512X512	257 KB	4.81 KB

Compression ratio with 'JPEG_db6':

An image 1.bmp is 301 KB before compression and 4.81 Kb after compression what are the values of compression ratio and saving parentage.

Compression ratio= 301/4.81
=64.4490

The interpretation is that 64.4490 pixels of input image are expressed as 1 pixel in the output image.

Saving Percentage =1-1/64.4490
=98.4483%

An image 2.bmp is 507KB before compression and 6.00 Kb after compression what are the values of compression ratio and saving parentage.

Compression ratio= 507/6.00
=84.5

The interpretation is that 84.5 pixels of input image are expressed as 1 pixel in the output image.

Saving Percentage =1-1/84.5
=98.8165 %

An image 3.bmp is 1.22 MB before compression and 16.9 Kb after compression what are the values of compression ratio and saving parentage.

Compression ratio= (1.22×1024)/16.9
=73.9218

The interpretation is that 73.9218 pixels of input image are expressed as 1 pixel in the output image.

Saving Percentage =1-1/73.9218
=98.6472 %

An image 4.bmp is 257 KB before compression and 4.81 Kb after compression what are the values of compression ratio and saving parentage.

Compression ratio= 257/4.81
=53.4303

The interpretation is that 53.4303 pixels of input image are expressed as 1 pixel in the output image.

Saving Percentage =1-1/53.4303
=98.128 %

Result for wavelets 'JPEG_db6":

Table 8: Result of compression ratio with db6

Input Image	Compression ratio
1.bmp	98.4483
2.bmp	98.8165
3.bmp	98.6472
4.bmp	98.128

Average compression ratio: 98.51%

Decompression with 'JPEG_db6':

Read the compressed image (*.bha) and run the decompression algorithms with wavelets "db6" and maintain the size of input image and decompressed. Compressed images are 1_db6.bha, 2_db6.bha, 3_db6.bha and 4_db6.bha.

Table 9: Decompression Details with 'db6'

S. no	input image	Size	Size after decompression	Dimension	Decompressed image
1	1_db6.bha	4.81 KB	301 kb	640X480	1_D_db6.bmp
2	2_db6.bha	6.00 KB	170 kb	604X288	2_D_db6.bmp
3	3_db6.bha	16.9 KB	420 kb	756X568	3_D_db6.bmp
4	4_db6.bha	4.81 KB	257 kb	512X512	4_D_db6.bmp

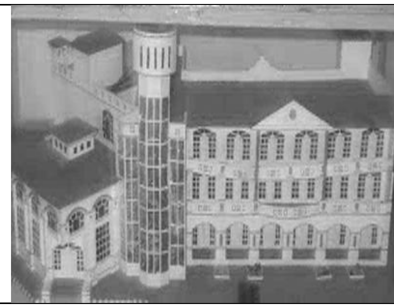
	
1_d_db6.bmp	2_d_db6.bmp
	
3_d_db6.bmp	4_d_db6.bmp

Figure 2: decompression image with db6

Comparison between wavelets and compression & Decompression ratio:

Table 10: Comparison with Different Wavelets

Wavelets	Compression ratio
Db4	98.39177
Db5	98.3589
Db6	98.51

From the table 10, we conclude that if space is more critical then select the "db6".

IX. QUALITY PARAMETER

There are different approaches to image quality evaluation and they are based on objective and subjective parameters. The quality of a compressed image is evaluated by analyzing the difference between the original and the compressed one.
One of the most widely used parameters for the evaluation of image quality is the MSE. The list of Image Quality measures implemented in this package include,

1. Structural Content (SC)

$$SC = \sum_{j=1}^{M} \sum_{k=1}^{N} {x_{j,k}}^2 \Big/ \sum_{j=1}^{M} \sum_{k=1}^{N} {x'_{j,k}}^2$$

2. Mean Square Error (MSE)

$$MSE = \frac{1}{MN} \sum_{j=1}^{M} \sum_{k=1}^{N} \left(x_{j,k} - x'_{j,k} \right)^2$$

3. Peak Signal to Noise Ratio (PSNR in dB)

$$PSNR = 10 \log \frac{(2^n - 1)^2}{MSE} = 10 \log \frac{255^2}{MSE}$$

4. Normalized Cross-Correlation (NCC)

$$NK = \sum_{j=1}^{M} \sum_{k=1}^{N} x_{j,k} \cdot x'_{j,k} \Big/ \sum_{j=1}^{M} \sum_{k=1}^{N} x_{j,k}^2$$

5. Average Difference (AD)

$$AD = \sum_{j=1}^{M} \sum_{k=1}^{N} \left(x_{j,k} - x'_{j,k} \right) \Big/ MN$$

6. Maximum Difference (MD)

$$MD = Max\left(\left|x_{j,k} - x'_{j,k}\right|\right)$$

7. Normalized Absolute Error (NAE)

$$NAE = \sum_{j=1}^{M}\sum_{k=1}^{N}\left|x_{j,k} - x'_{j,k}\right| \Big/ \sum_{j=1}^{M}\sum_{k=1}^{N}\left|x_{j,k}\right|$$

X. Result of Quality parameter:

Above said parameters have been implemented with MATLAB and find out the value of the said parameter with image 4.bmp and decompressed image with wavelets db4, db5, and db6.

S.No	Parameter	Db4	Db5	Db6
1	Mean Square Error	1222	1201	1178
2	Peak Signal to Noise Ratio	17.2484	17.3339	17.4194
3	Normalized Cross-Correlation	0.8592	0.8621	.8650
4	Average Difference	4	3.800	3.6000
5	Structural Content	1.1414	1.1379	1.1343
6	Maximum Difference	60	59	58
7	Normalized Absolute Error	0.2926	0.2902	.2878

On the basic of above table and ideal value of above said parameters, we can categorize the wavelets means which one is more beneficial with respect to the quality parameter.

S.No	Parameter	Wavelets
1	Mean Square Error	Db6
2	Peak Signal to Noise Ratio	Db6
3	Normalized Cross-Correlation	Db6
4	Average Difference	Db6
5	Structural Content	Db6
6	Maximum Difference	Db6
7	Normalized Absolute Error	Db6

Now from the above table db6 is more powerful wavelets for quality parameter point of view.

XI. Conclusion and future scope

We have studied three JPEG wavelets algorithms successfully JPEG_db4, JPEG_db5, JPEG_db6 and concluded that JPEG_db6 is better in context to compression ratio in comparison to other algorithms that we have considered as well for quality parameter .This proposed algorithm can save storage space in some online applications where picture and signature are stored in the databases for future reference.

For the future work, the above algorithms can be studied for video compression also. Parallel model may be enhanced and modeled in some combination of mathematical transformation. As we have calculated compression time, we can also calculate decompression time. Like fast Fourier, wavelets, sparse orthonormal transform, Sparse Lapped Transforms and other transformation.

Reference

[1] Ahmed, N., Natarajan, T., Rao, K. R., “Discrete Cosine Transform”, IEEE Trans. Computers, vol. C-23, Jan. 1974, pp. 90-93.

[2] Wallace, G.. The JPEG still picture compression standard. Communications of the ACM 34(4): 30-44. 1991.

[3] Watson, A. B. DCT quantization matrices visually optimized for individual images. Proceedings of the SPIE 1913: 202-216 (Human Vision, Visual Processing, and Digital Display IV. Rogowitz ed. SPIE. Bellingham, WA). 1993

[4] Matthias Kramm, "Compression of Image Clusters using Karhunen Loeve Transformations", TU-M¨unchen, Institute for Computer Science, Boltzmannstr. 3, 85748 Garching, Germany.

[5] N. Ahmed, T. Natarajan, and K. R. Rao, “Discrete cosine transform,” IEEE Trans. Comput., vol. C-23, pp. 90–93, Jan. 1974.

[6] J. M. Shapiro, “Embedded image coding using zero trees of wavelet coefficients,” IEEE Trans. Signal Processing, vol. 41, pp. 3445–3462, Dec.1993.

[7] A. Said andW. A. Pearlman, “New, fast, and efficient image codec based on set partitioning in hierarchical trees,” IEEE Trans. Circuits, Syst., Video Technol., vol. 6, pp. 243–249, June 1996.

[8] C. Chrysafis and A. Ortega, “Efficient context-based entropy coding for lossy wavelet image compression,” in Proc. 1997 Data Compression Conf., Mar. 1997, pp. 241–250.

[9] J. Liang and T. D. Tran, “Fast multiplier less approximations of the DCT with the lifting scheme,” IEEE Trans. Signal Processing, vol. 49, pp. 3032–3044, Dec. 2001.

[10] G. K. Wallace, “The JPEG still picture compression standard,” IEEE Trans. Consumer Electronics, vol. 38, pp. 18–34, Feb. 1992.

[11] M. Atonini, M. Barlaud, P. Mathieu, and I. Daubechies, “Image coding using wavelet transform,” IEEE Trans. Image Processing, vol. 1, pp. 205–220, Apr. 1992.

[12] H. S. Malvar, Signal Processing With Lapped Transforms. Norwood, MA: Artech House, 1992.

[13] R. C. Gonzalea and R. E. Woods, "Digital Image Processing", 2nd Ed., Prentice Hall, 2004.

[14] Liu Chien-Chih, Hang Hsueh-Ming, "Acceleration and Implementation of JPEG 2000 Encoder on TI DSP platform" Image Processing, 2007. ICIP 2007. IEEE International Conference on, Vo1. 3, pp. III-329-339, 2005.

[15] Ahmed, N., T. Natarajan, and K. R. Rao. 1974. On image processing and a discrete cosine transform. IEEE Transactions on Computers C-23(1): 90-93.

[16] Chen, W. H., and W. K. Pratt. 1984. Scene adaptive coder. IEEE Transactions on Communications COM-32: 225-232.

[17] Jain, A. K. 1989. Fundamentals of digital image processing. Prentice Hall: Englewood Cliffs, NJ.

[18] Puri, A. 1992. Video coding using the MPEG-1 compression standard. Society for Information Display Digest of Technical Papers 23: 123-126.

[19] Shannon, C. E., and W. Weaver. 1949. The mathematical theory of communication. Urbana: University of Illinois Press.

[20] Wallace, G. 1991. The JPEG still picture compression standard. Communications of the ACM 34(4): 30-44.

[21] Watson, A. B. 1993. DCT quantization matrices visually optimized for individual images.
Proceedings of the SPIE 1913: 202-216 (Human Vision, Visual Processing, and Digital Display IV. Rogowitz ed. SPIE. Bellingham, WA).

[22] W. A. Pearlman. “High Performance, Low Complexity Image Compression,” Applications of Digital Image Processing X, Proc. SPIE 3164, July 1997, pp. 234 – 246.

[23] H. S. Malvar and D. H. Staelin. “The LOT: Transform coding without blocking effects,” IEEE Trans. Acoustics, Speech, Signal Processing, vol. 37, April 1989, pp. 553 – 559.

[24] G. K. Wallace. “The JPEG Still Picture Compression Standard,” Communications of the ACM, vol. 34, April 1991, pp. 31 – 44.

