 ISSN : 2230-7109(Online) | ISSN : 2230-9543(Print)

A Review of Image Compression and Comparison of its Algorithms

Sachin Dhawan
Deptt. of ECE, UIET, Kurukshetra University, Kurukshetra, Haryana, India

Abstract
Image compression is now essential for applications such as transmission and storage in data bases. In this paper we review and discuss about the image compression, need of compression, its principles, and classes of compression and various algorithm of image compression. This paper attempts to give a recipe for selecting one of the popular image compression algorithms based on Wavelet, JPEG/DCT, VQ, and Fractal approaches. We review and discuss the advantages and disadvantages of these algorithms for compressing grayscale images, give an experimental comparison on 256×256 commonly used image of Lenna and one 400×400 fingerprint image.

Keywords
Image compression; JPEG; DCT; VQ; Wavelet; Fractal.

I. Introduction
Image compression is the application of data compression on digital images. In effect, the objective is to reduce redundancy of the image data in order to be able to store or transmit data in an efficient form.
Uncompressed multimedia (graphics, audio and video) data requires considerable storage capacity and transmission bandwidth. Despite rapid progress in mass-storage density, processor speeds, and digital communication system performance, demand for data storage capacity and data-transmission bandwidth continues to outstrip the capabilities of available technologies. The recent growth of data intensive multimedia-based web applications have not only sustained the need for more efficient ways to encode signals and images but have made compression of such signals central to storage and communication technology.

II. What are the principles behind 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:

A. Coding Redundancy
A code is a system of symbols (letters, numbers, bits, and the like) used to represent a body of information or set of events. Each piece of information or events is assigned a sequence of code symbols, called a code word. The number of symbols in each code word is its length. The 8-bit codes that are used to represent the intensities in the most 2-D intensity arrays contain more bits than are needed to represent the intensities.

B. Spatial Redundancy and Temporal Redundancy
Because the pixels of most 2-D intensity arrays are correlated spatially, information is unnecessarily replicated in the representations of the correlated pixels. In video sequence, temporally correlated pixels also duplicate information.

C. Irrelevant Information
Most 2-D intensity arrays contain information that is ignored by the human visual system and extraneous to the intended use of the image. It is redundant in the sense that it is not used.
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.

III. Why do we need compression?
The Table 1 show the qualitative transition from simple text to full-motion video data and the disk space transmission bandwidth, and transmission time needed to store and transmit such uncompressed data.
Table 1 : Multimedia data types and uncompressed storage space, transmission bandwidth, and transmission time required. The prefix kilo- denotes a factor of 1000 rather than 1024.

Multimedia Data	Size/ Duration	Bits/Pixel or Bits/ Sample	Uncompressed Size (B for bytes)	Transmission Bandwidth (b for bits)	Transmission Time
A page of text	11"x8.5"	Varying resolution	4-8KB	32-64 Kb/ page	1.1 - 2.2 sec
Telephone quality speech	10 sec	8 bps	80 KB	64 Kb/sec	22.2 sec
Grayscale Image	512x512	8 bpp	262KB	2.1 Mb/ image	1 min 13 sec
Color Image	512x512	24 bpp	786KB	6.29 Mb/ image	3 min 39 sec
Medical Image	2048x 1680	12 bpp	5.16 MB	41.3 Mb/ image	23 min 54 sec
SHD Image	2048 x 2048	24 bpp	12.58 MB	100 Mb/ image	58 min 15 sec

The examples given in the Table I clearly illustrate the need for sufficient storage space, large transmission bandwidth, and long transmission time for image, audio, and video data.
At the present state of technology, the only solution is to compress multimedia data before its storage and transmission, and decompress it at the receiver for play back. For example, with a compression ratio of 32:1, the space, bandwidth, and transmission time requirements can be reduced by a factor of 32, with acceptable quality.

IV. What are the different classes of compression techniques?
Two ways of classifying compression techniques are mentioned here.

A. 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 achieve a modest amount of compression. An image reconstructed following lossy compression contains degradation relative to the original. Often this is because the compression scheme completely discards

redundant information. However, lossy schemes are capable of achieving much higher compression. Under normal viewing conditions, no visible loss is perceived (visually lossless).

B. 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.

V. What does a typical image coder look like?

A typical lossy image compression system which consists of three closely connected components namely (a) Source Encoder (b) Quantizer, and (c) Entropy Encoder. Compression is accomplished by applying a linear transform to decorrelate the image data, quantizing the resulting transform coefficients, and entropy coding the quantized values.

A. Source Encoder (or Linear Transformer)

Over the years, a variety of linear transforms have been developed which include Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT) [1], Discrete Wavelet Transform (DWT)[13] and many more, each with its own advantages and disadvantages.

B. Quantizer

A quantizer simply reduces the number of bits needed to store the transformed coefficients by reducing the precision of those values. Since this is a many-to-one mapping, it is a lossy process and is the main source of compression in an encoder. Quantization can be performed on each individual coefficient, which is known as Scalar Quantization (SQ). Quantization can also be performed on a group of coefficients together, and this is known as Vector Quantization (VQ). Both uniform and non-uniform quantizers can be used depending on the problem at hand.

C. Entropy Encoder

An entropy encoder further compresses the quantized values lossless to give better overall compression. It uses a model to accurately determine the probabilities for each quantized value and produces an appropriate code based on these probabilities so that the resultant output code stream will be smaller than the input stream. The most commonly used entropy encoders are the Huffman encoder and the arithmetic encoder, although for applications requiring fast execution, simple run-length encoding (RLE) has proven very effective.

VI. Various Compression Algorithms

A. JPEG : DCT-Based Image Coding Standard

The JPEG/DCT still image compression has become a standard recently. JPEG is designed for compressing full-color or gray-scale images of natural, real-world scenes.

To exploit this method, an image is first partitioned into non overlapped 8×8 blocks. A discrete Cosine transform (DCT) [10, 14] is applied to each block to convert the gray levels of pixels in the spatial domain into coefficients in the frequency domain. The coefficients are normalized by different scales according to the quantization table provided by the JPEG standard conducted by some psycho visual evidence. The quantized coefficients are rearranged in a zigzag scan order to be further compressed by an efficient lossless coding strategy such as run length coding, arithmetic coding, or Huffman coding. The decoding is simply the inverse process of encoding. So, the JPEG compression takes about the same time for both encoding and decoding. The encoding/ decoding algorithms provided by an independent JPEG group [14] are available for testing real-world images. The information loss occurs only in the process of coefficient quantization. The JPEG standard defines a standard 8×8 quantization table [14] for all images which may not be appropriate. To achieve a better decoding quality of various images with the same compression by using the DCT approach, an adaptive quantization table may be used instead of using the standard quantization table.

B. Image Compression by Wavelet Transform

1. What is a Wavelet Transform?

Wavelets are functions defined over a finite interval and having an average value of zero. The basic idea of the wavelet transform is to represent any arbitrary function (t) as a superposition of a set of such wavelets or basis functions. These basis functions or baby wavelets are obtained from a single prototype wavelet called the mother wavelet, by dilations or contractions (scaling) and translations (shifts). The Discrete Wavelet Transform of a finite length signal x(n) having N components, for example, is expressed by an N x N matrix. For a simple and excellent introduction to wavelets, see [3].

2. Why Wavelet-based Compression?

Despite all the advantages of JPEG compression schemes based on DCT 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. This results in noticeable and annoying ``blocking artifacts" particularly at low bit rates as shown in Fig.1. Lapped Orthogonal Transforms (LOT) [5] attempt to solve this problem by using smoothly overlapping blocks. Although blocking effects are reduced in LOT compressed images, increased computational complexity of such algorithms do not justify wide replacement of DCT by LOT.

(a) (b)

Fig.1:(a) Original Lena Image, and (b) Reconstructed Lena with DC component only, to show blocking artifacts.

 ISSN : 2230-7109(Online) | ISSN : 2230-9543(Print)

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 (also referred as sub band coding) outperform other coding schemes like the one based on DCT. Since there is no need to block the input image and its basis functions have variable length, wavelet coding schemes at higher compression avoid blocking artifacts. Wavelet-based coding [2] is more robust under transmission and decoding errors, and also facilitates progressive transmission of images. In addition, they are better matched to the HVS characteristics. Because of their inherent multi-resolution nature [6], wavelet coding schemes are especially suitable for applications where scalability and tolerable degradation are important.

C. VQ Compression

A vector quantizer is composed of two operations. The first is the encoder, and the second is the decoder. The encoder takes an input vector and outputs the index of the codeword that offers the lowest distortion. In this case the lowest distortion is found by evaluating the Euclidean distance between the input vector and each codeword in the codebook. Once the closest codeword is found, the index of that codeword is sent through a channel (the channel could be computer storage, communications channel, and so on). When the encoder receives the index of the codeword, it replaces the index with the associated codeword.

The fundamental idea of VQ[4] for image compression is to establish a codebook consisting of code vectors such that each code vector can represent a group of image blocks of size m × m, (m=4 is always used). An image or a set of images is first partitioned into m × m non overlapping blocks which are represented as m2-tuple vectors, called training vectors. The size of training vectors can be very large. For example, a 512 × 512 image contributes 16,384 training vectors.

The goal of codebook design is to establish a few representative vectors, called code vectors of size 256 or 512, from a set of training vectors. The encoding procedure is to look for a closest code vector in the codebook for each non overlapped 4 × 4 block of an image to be encoded. The most important work is to design a versatile codebook. Nasrabadi and King [11] give a good review of VQ. Chen's comparison [16] indicates that a codebook developed based on LBG [12] algorithm generally has higher PSNR values over some other schemes despite its slow off-line training. In this paper, we adopt LBG algorithm for training a codebook of size 256×256 to meet a desired 0.5 bpp compression ratio.

D. Fractal Compression

Fractal image coding was introduced in the late 1980s and early 1990s [20]. It is used for encoding/ decoding images in Encarta/Encyclopedia [15]. Fractal coding is based on the Collage theorem and the fixed point theorem [15, 19] for a local iterated function system consisting of a set of contraction affine transformations [15]. A fractal compression algorithm first partitions an image into non overlapping 8×8 blocks, called range blocks and forms a domain pool containing all of possibly overlapped 16×16 blocks, associated with 8 isometries from reflections and rotations, called domain blocks. For each range block, it exhaustively searches, in a domain pool, for a best matched domain block with the minimum square error after a contractive affine transform is applied to the domain=block.

A fractal compressed code for a range block consists of quantized contractively coefficients in the affine transform, an offset which is the mean of pixel gray levels in the range block, the position of the best matched domain block and its type of isometry. The decoding is to find the fixed point, the decoded image, by starting with any initial image. The procedure applies a compressed local affine transform on the domain block corresponding to the position of a range block until all of the decoded range blocks are obtained. The procedure is repeated iteratively until it converges (usually in no more than 8 iterations).

Two serious problems that occur in fractal encoding are the computational demands and the existence problem of best range-domain matches [19]. The most attractive property is the resolution-independent decoding property. One can enlarge an image by decoding an encoded image of smaller size so that the compression ratio may increase exponentially [15,18]. An algorithm based on [20] using range and domain block matches of fixed sizes is written and is used for a comparison in this paper [17].

VII. Advantages And Disadvantages Of Various Compression Algorithm

There are some advantages and disadvantages of various algorithms which are shown in table 2.

TABLE 2 : Experimental Comparision

Method	Advantages	Disadvantages
Wavelet	High Compression Ratio State-Of-The-Art	Coefficient quantization Bit allocation
JPEG	Current Standard	Coefficient(dct) quantization Bit allocation
VQ	Simple decoder No-coefficient quantization	Slow codebook generation Small bpp
Fractal	Good mathematical Encoding-frame	Slow Encoding

Image compression algorithms based on Wavelet Transform [9], JPEG/DCT [7], Vector Quantization [16], and Fractal [15] methods were tested for 256×256 real image of Lenna and 400×400 fingerprint image. The results of performance are shown in Table 3, 4 and 5.

In Table III, IV and V the performance of different algorithms is shown in which there is PSNR value and CPU Time (Encoding and Decoding) is shown. And we summarize the comparison of Compression ratio of different algorithm in Table 6 given below.

TABLE 3 : performance of coding algorithms on 256×256 images

Algorithm	PSNR values OF Leena's image (in dB)	CPU time	
		Encoding	Decoding
Wavelet	34.66	0.35 sec	0.27 sec
JPEG	31.73	0.12 sec	0.12 sec
VQ	29.28	2.45 sec	0.18 sec
Fractal	29.04	5.65 hrs	1.35 sec

TABLE 4 : Performance Of Coding Algorithms On A 400×400 Fingerprint Image Of 0.5bpp

Algorithm	0.5bpp		
	PSNR values	Encoding Time	Decoding Time
Wavelet	36.71	0.8 sec	0.7 sec
JPEG	34.27	0.2 sec	0.2 sec
VQ	28.26	6.0 sec	0.7 sec
Fractal	27.21	6.3 hrs	3.5 sec

TABLE 5 : Performance Of Coding Algorithms On A 400×400 Fingerprint Image Of 0.25bpp

Algorithm	0.25bpp		
	PSNR values	Encoding Time	Decoding Time
Wavelet	32.47	0.7 sec	0.5 sec
JPEG	29.64	0.2 sec	0.2 sec
VQ	N/A	N/A	N/A
Fractal	N/A	N/A	N/A

TABLE 6 : Performance On The Basis Of Compression Ratio Of Different Coding Algorithms

Method	Compression ratio
Wavelet	>>32
JPEG	<=50
VQ	<32
Fractal	>=16

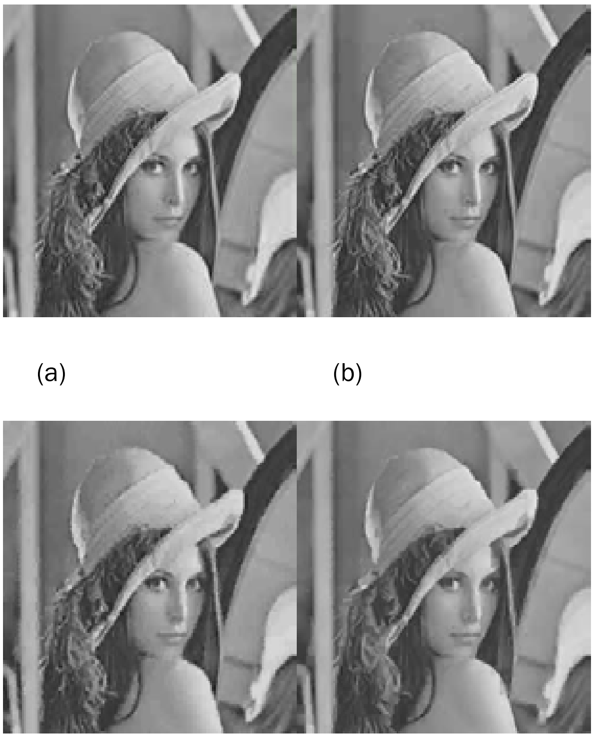
The associated PSNR values and encoding/decoding times shown in Table III ,IV and V for the images shown in Fig.2 indicate that all the four approaches are satisfactory at 0.5 bpp request (CR=16). However, the EZW [11, 8] has significantly larger PSNR values and a better visual quality of decoded images compared with the other approaches.

At a desired compression of 0.25 bpp (CR=32) for the fingerprint image, the commonly used VQ cannot be tested, and the fractal coding cannot be achieved unless resolution-free decoding property is utilized which is not useful for the current purpose; both EZW [9] and JPEG [7] approaches perform well, and the results of EZW have significant larger PSNR values than that of JPEG. The original images of Lenna and fingerprint are shown in Fig.2.

(a) (b)

Fig.2: Original images of (a) Lenna and (b) fingerprint

The decoded images of Leena based on the four approaches (a) Wavelet Transform, (b) JPEG, (c) Vector Quantization, (d) Fractal are shown in Fig. 3.



(a) (b)

(c) (d)

Fig.3: Decoded image of Lena by (a) Wavelet, (b) JPEG, (c) VQ, and (d) Fractal algorithms

The decoded images of fingerprints based on the four approaches (a) Wavelet Transform, (b) JPEG, (c) Vector Quantization, (d) Fractal are shown in Fig. 4.

ISSN : 2230-7109(Online) | ISSN : 2230-9543(Print)

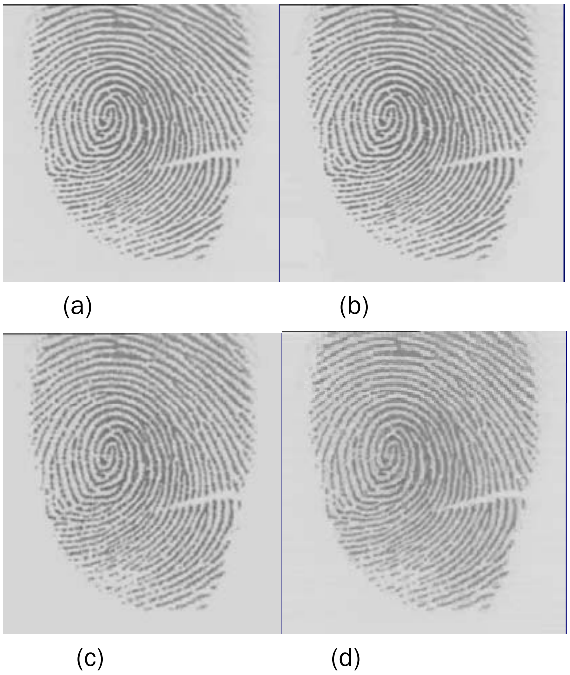


Fig.4 : Decoded fingerprints by (a) Wavelet, (b) JPEG, (c) VQ, (d) Fractal algorithms.

IX. Conclusion

We have reviewed and summarized the characteristics of image compression, need of compression, principles behind compression, different classes of compression techniques and various image compression algorithms based on Wavelet, JPEG/DCT, VQ, and Fractal approaches. Experimental comparisons on 256×256 commonly used image of Lenna and one 400×400 fingerprint image suggest a recipe described as follows. Any of the four approaches is satisfactory when the 0.5 bits per pixel (bpp) is requested. However, for a very low bit rate, for example 0.25 bpp or lower, the embedded zero tree wavelet (EZW) approach is superior to other approaches. For practical applications, we conclude that (1) Wavelet based compression algorithms are strongly recommended, (2) DCT based approach might use an adaptive quantization table, (3) VQ approach is not appropriate for a low bit rate compression although it is simple, (4) Fractal approach should utilize its resolution-free decoding property for a low bit rate compression.

References

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

[2] Buccigrossi, R., Simoncelli, E. P., "EPWIC: Embedded Predictive Wavelet Image Coder".

[3] Chan, Y. T., "Wavelet Basics", Kluwer Academic Publishers, Norwell, MA, 1995.

[4] Gersho, A., Gray, R.M., "Vector Quantization and Signal Compression", Kluwer Academic Publishers, 1991.

[5] Malavar, H. S., "Signal Processing with Lapped Transforms", Norwood, MA, Artech House, 1992.

[6] Mallat, S.G., "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation", IEEE Trans. PAMI, vol. 11, no. 7, July 1989, pp. 674-693.

[7] [Online] Available : ftp.uu.net:/graphics/jpeg/jpegsrc.v6a.tar.gz

[8] J.M. Shapiro, "Embedded image coding using zero tree of wavelet coefficients", IEEE Trans. on Signal Processing, vol. 41, 3445-3462, 1993.

[9] A. Said, W.A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees", IEEE Trans. on Circuits and Systems for Video Technology, vol. 6, 243-250, 1996.

[10] Rao, K. R., Yip, P., "Discrete Cosine Transforms - Algorithms, Advantages, Applications", Academic Press, 1990.

[11] N.M. Nasrabadi, R.A. King, "Image coding using vector quantization: a review", IEEE Trans. On Communications,vol. 36, 957-571, 1988.

[12] Y. Linde, A. Buzo, R. M Gray, "An algorithm for vector quantizer design", IEEE Trans. on Communications, vol. 36, 84-95, 1980.

[13] Vetterli, M., Kovacevic, J. Wavelets, "Subband Coding", Englewood Cliffs, NJ, Prentice Hall, 1995, [Online] Available : http://cm.bell-labs.com/who/jelena/Book/home.html.

[14] Wallace, G. K. "The JPEG Still Picture Compression Standard", Comm. ACM, vol. 34, no. 4, April 1991, pp. 30- 44.

[15] M.F. Barnsley, L.P. Hurd, "Fractal Image Compression", AK Peters, Ltd. Wellesley, Massachusetts, 1993.

[16] Y.W. Chen, "Vector Quantization by principal component analysis", M.S. Thesis, National Tsing Hua University, June, 1998.

[17] H.S.Chu, "A very fast fractal compression algorithm", M.S. Thesis, National Tsing Hua University, June, 1997.

[18] Y. Fisher, "Fractal Image Compression", SIGGRAPH Course Notes, 1992.

[19] Y. Fisher, Editor, "Fractal Image Compression: Theory and Applications", Springer-Verlag, 1994.

[20] A.E. Jacquin, "Image coding based on a fractal theory of iterated contractive image transformations". IEEE Trans. on Image Processing, vol. 1, pp.18-30, 1992.

Sachin Dhawan received the B.Tech. Degree (with first class honors) in electronics & Communication engineering, in 2008 from Doon Valley institute of Engineering and Technology affiliated from Kurukshetra University. And presently pursuing MTech in Electronics and Communication from University institute of engineering and Technology, Kurukshetra University. He is also a teaching associate in Electronics and Communication Department, University Institute of Engineering and Technology. His interest area is Image processing and Digital Signal Processing. Currently he is working on improvement of one of the compression technique that is Fractal image compression technique. He has published four technical papers in national and international conferences and one technical paper in international journal.