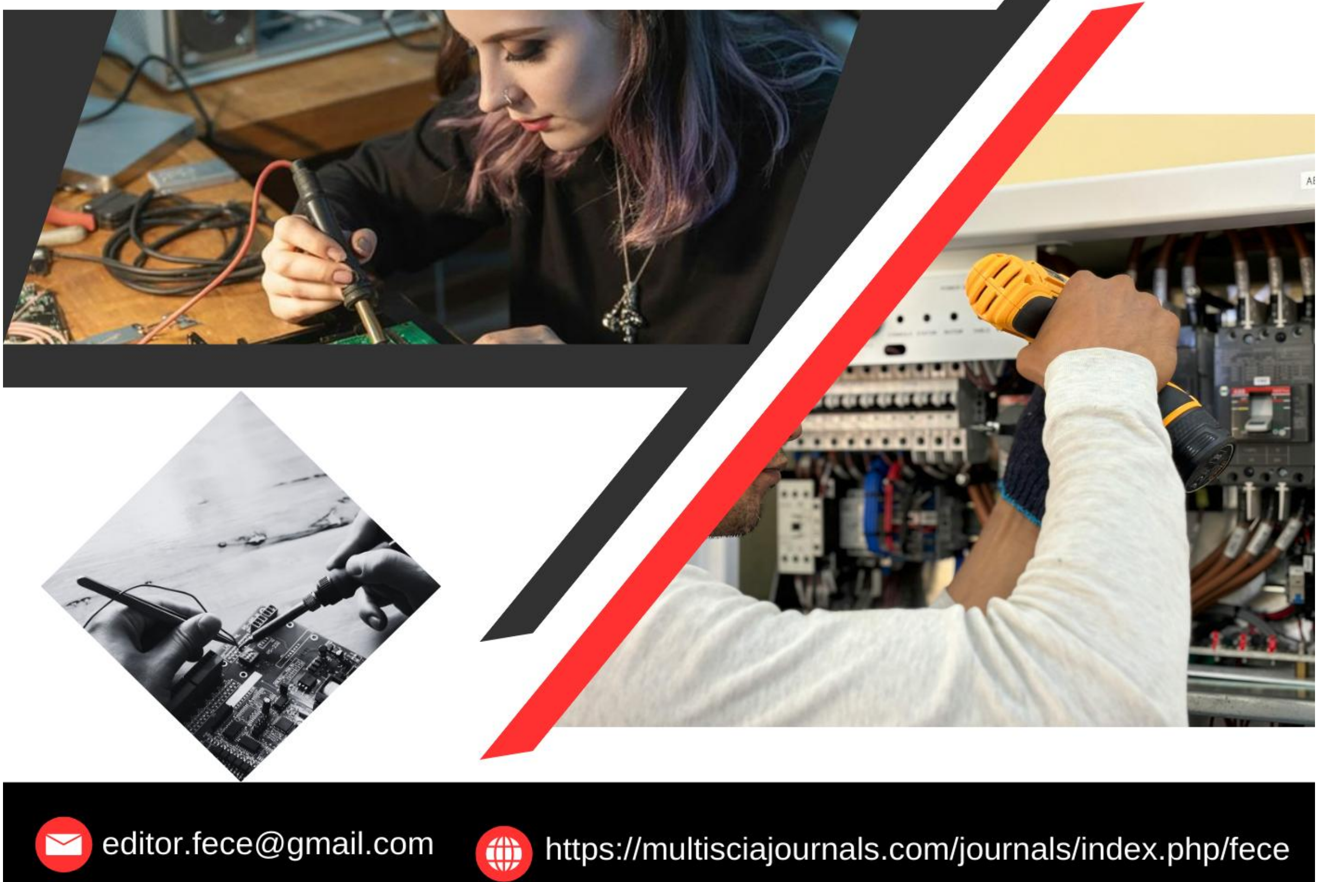
MULTISCIA
JOURNALS PUBLISHERS

FRONTIERS IN ELECTRONICS AND COMMUNICATION ENGINEERING

ISSN: (3065- 4106)



editor.fece@gmail.com https://multisciajournals.com/journals/index.php/fece

Frontiers in Electronics and Communication Engineering
Volume 1 Issue 4 2025

A Comparative Analysis of Image Compression Algorithms

Cristiano Giuffrida*, Giovanni Abramo
Department of ECE

Article Info

Received: 25-10-2025 ***Revised:02-12-2025*** ***Accepted:11-12-2025*** ***Published:22-12-2025***

Abstract

These days, image compression is crucial for uses like data base storage and communication. This study examines and discusses picture compression, including its necessity, concepts, classes, and different algorithms on the subject. The goal of this study is to provide a formula for choosing one of the widely used image compression algorithms based on the Wavelet, JPEG/DCT, VQ, and fractal techniques. We examine and analyze the benefits and drawbacks of different grayscale picture compression algorithms, and we provide an experimental comparison using one 400x400 fingerprint image and a regularly used 256x256 image of Lenna.

Keywords

I. image compression, wavelet, fractal, DCT, VQ, and JPEG.

II. Introduction

III. The process of applying data compression to digital photos is known as image compression. In actuality, the goal is to minimize picture data redundancy so that data may be efficiently stored or transmitted.

Large amounts of storage space and transmission bandwidth are needed for uncompressed multimedia (graphics, audio, and video) data. The need for data storage capacity and data transmission bandwidth continues to exceed the capabilities of current technology, despite the rapid advancements in mass storage density, processor speeds, and digital communication system performance. In addition to maintaining the need for more effective methods of encoding signals and images, the recent expansion of data-intensive multimedia-based web applications has made signal compression essential to communication and storage technologies.

IV. What are the principles behind compression?

V. The majority of photos share the trait of having connected nearby pixels, which results in duplicate information. Finding a less correlated representation of the image is therefore the most important challenge. Redundancy and irrelevancy reduction are two essential elements of compression. Eliminating duplication from the signal source (image/video) is the goal of redundancy reduction. Irrelevancy reduction leaves out portions of the signal that the Human Visual System (HVS), which is the signal receiver, will not detect. Three categories of redundancy can often be distinguished:

A. Coding Redundancy

VI. A system of symbols, such as letters, numbers, bits, and the like, that are used to represent a collection of events or a body of information is called a code. Each piece of information or events is assigned a sequence of code symbols, called a code word. Each code word's length is determined by how many symbols it contains. The intensities in the majority of 2-D intensity arrays are represented by 8-bit codes, which have more bits than are required to do so.

Spatial Redundancy and Temporal Redundancy Most 2-D intensity arrays have spatially correlated pixels, therefore information is needlessly duplicated in the

A.

VII. depictions of the associated pixels. Temporally connected pixels also duplicate information in a video clip.

A. Irrelevant Information

VIII. The majority of 2-D intensity arrays include information that is irrelevant to the intended purpose of the image and disregarded by the human visual system. In the sense that it is not utilized, it is redundant. By eliminating as much of the spatial and spectral redundancy as feasible, image compression research seeks to lower the number of bits required to represent an image.

VIII. Why is compression necessary?
The qualitative shift from plain text to full-motion video data, as well as the disk space transmission bandwidth and transmission time required to store and send such uncompressed data, are displayed in Table 1.
Table 1 lists the various types of multimedia data along with the amount of uncompressed storage space, bandwidth, and transmission time needed. Instead of 1024, the prefix kilo- indicates a multiplier of 1000. For image, audio, and video data, the examples in Table I make it abundantly evident that adequate storage capacity, a high bandwidth, and a lengthy transmission duration are required. The only option available with current technology is to compress multimedia data before storing and sending it, then decompress it at the recipient to play it again. For instance, the space, bandwidth, and transmission time requirements can be lowered by a factor of 32 while maintaining acceptable quality by using a compression ratio of 32:1.

IX. Which sorts of compression techniques exist?
Here, two classification schemes for compression approaches are discussed.

A. Lossless versus lossy compression
The reconstructed image in lossless compression algorithms is quantitatively identical to the original image following compression. However, only a little amount of compression can be accomplished using lossless compression. After lossy compression, an image's quality deteriorates in comparison to the original. This is frequently the result of the compression strategy entirely discarding duplicate data. Lossy techniques, on the other hand, can achieve substantially higher compression. Under typical viewing circumstances, there is no discernible loss (visually lossless).

A. Predictive vs. Transform coding

IX. Predictive coding uses data that has already been communicated or made available to forecast future values; the discrepancy is then coded. This is reasonably easy to implement and easily adaptable to local picture features because it is done in the image or spatial domain. One specific instance of predictive coding is Differential Pulse Code Modulation (DPCM). On the other hand, transform coding codes the changed values (coefficients) after first converting the image from its spatial domain representation to an other kind of representation using a well-known transform. Compared to predictive methods, this approach offers more data compression, but at the cost of more processing.

X. What does a typical image coder look like?

XI. Three closely related parts make up a standard lossy image compression system: the source encoder, quantizer, and entropy encoder. To achieve compression, the image data is decorrelated using a linear transform, the transform coefficients are quantized, and the quantized values are entropy coded.

A. Source Encoder (or Linear Transformer)

XII. Numerous linear transforms, each with unique benefits and drawbacks, have been created over time, such as the Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT) [1], Discrete Wavelet Transform (DWT) [13], and many more.

A. Quantizer

XIII. By decreasing the precision of the modified coefficients, a quantizer merely lowers the number of bits required to store them. This is the primary source of compression in an encoder and is a lossy operation because it is a many-to-one mapping. Scalar quantization, or SQ, is the process of quantizing each individual coefficient. Vector Quantization (VQ) is another type of quantization that can be applied on a collection of coefficients collectively. Depending on the issue, either uniform or non-uniform quantizers can be applied.

A. Entropy Encoder

XIV. To improve total compression, the quantized values are further compressed losslessly using an entropy encoder. In order to ensure that the output code stream is smaller than the input stream, it employs a model to precisely calculate the probability for each quantized value and generates a suitable code based on these probabilities. The Huffman encoder and the arithmetic encoder are the most widely used entropy encoders; nevertheless, basic run-length encoding (RLE) has shown great success for applications that need to execute quickly.

XV. Various Compression Algorithms

A. JPEG: Image Coding Standard Based on DCT
Recently, the JPEG/DCT still picture compression standard was established. Images of natural, real-world scenes in full color or grayscale can be compressed using JPEG.
In order to use this technique, an image is first divided into non overlapping blocks of 8 by 8. The gray levels of pixels in the spatial domain are transformed into coefficients in the frequency domain by applying a discrete Cosine transform (DCT) [10, 14] to each block. According to the quantization table supplied by the JPEG standard, the coefficients are normalized using various scales based on certain psychovisual data. To be further compressed by an effective lossless coding technique like run length coding, arithmetic coding, or Huffman coding, the quantized coefficients are reorganized in a zigzag scan order. Simply put, decoding is the opposite of encoding. Therefore, encoding and decoding JPEG compression require roughly the same amount of time. Real-world photos can be tested using the encoding/decoding algorithms supplied by an independent JPEG organization [14]. Only during the coefficient quantization procedure does information become lost. For all photos, the JPEG standard specifies a common 8x8 quantization table [14], which might not be suitable. An adaptive quantization table may be used in place of the regular quantization table to improve the decoding quality of different images with the same compression using the DCT technique.

A. Image Compression by Wavelet Transform

1. What is a Wavelet Transform?

XVI. Wavelets are functions with an average value of zero that are defined over a finite interval. Representing any arbitrary function (t) as a superposition of a collection of these wavelets or basis functions is the fundamental principle behind the wavelet transform. Through translations (shifts) and dilations or contractions (scaling), these basis functions, also known as baby wavelets, are derived from a single prototype wavelet known as the mother wavelet. For instance, a N x N matrix represents the Discrete Wavelet Transform of a finite length signal x(n) with N components. See [3] for an excellent and straightforward introduction to wavelets.

1. Why Wavelet-based Compression?

XVII. Even though JPEG compression systems based on DCT have many benefits, such as ease of use, adequate performance, and the availability of specialized hardware for implementation, they are not without drawbacks. Correlation across the block boundaries is not eliminated because the input image must be ``blocked. As seen in Fig. 1, this leads to observable and bothersome "blocking artifacts," especially at low bit rates. Lapped Orthogonal Transforms (LOT) [5] use smoothly overlapping blocks to try to alleviate this issue. Even though LOT compressed images have fewer blocking effects, the higher computational complexity of these techniques does not support LOT's widespread replacement of DCT.

(a) (b)

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

The wavelet transform has become widely accepted in signal processing in general and picture compression research in particular over the last few years. Wavelet-based schemes, often known as sub-band coding, perform better than other coding schemes, such as the DCT-based system, in many situations. Wavelet coding techniques with higher compression prevent blocking artifacts since there is no need to block the input image and its basis functions have variable length. Wavelet-based coding [2] allows for progressive image transmission and is more resilient to transmission and decoding problems. Furthermore, they fit the HVS features better. Wavelet coding methods are particularly well-suited for applications where scalability and acceptable degradation are crucial due to their intrinsic multi-resolution nature [6].

A. VQ Compression

v Two operations make up a vector quantizer. The encoder comes first, followed by the decoder. The index of the codeword with the least amount of distortion is produced by the encoder from an input vector. In this instance, calculating the Euclidean distance between each codeword in the codebook and the input vector yields the least amount of distortion. The index of the closest codeword is transmitted over a channel (which could be a communications channel, computer storage, etc.) after it has been located. The encoder substitutes the codeword for the index when it receives the codeword's index. Creating a codebook of code vectors that may individually represent a collection of image blocks of size m × m (m=4 is always used) is the basic concept behind VQ[4] for image compression. Initially, an image or collection of images is divided into m × m non-overlapping blocks, which are represented as m2-tuple vectors, also known as training vectors. Training vectors can have a very large size. For instance, a 512

16,384 training vectors are contributed by the 512 image. Establishing a small number of representative vectors from a collection of training vectors—known as code vectors of size 256 or 512—is the aim of codebook design. For each nonoverlapping 4 × 4 block of an image that needs to be encoded, the encoding process finds the closest code vector in the codebook. Creating a flexible codebook is the most crucial task. VQ is well reviewed by Nasrabadi and King [11], and Chen's study [16] shows that, despite its long off-line training, a codebook built using the LBG [12] technique typically has higher PSNR values than certain other schemes. In this work, we use the LBG technique to train a 256x256 codebook with a target compression ratio of 0.5 bpp.

B. Fractal Compression

XVIII. The late 1980s and early 1990s saw the introduction of fractal image coding [20]. Encarta/Encyclopedia uses it to encode and decode images [15]. The Collage theorem and the fixed point theorem [15, 19] for a local iterated function system made up of a collection of contraction affine transformations [15] serve as the foundation for fractal coding. A fractal compression algorithm first divides a picture into 8x8 non-overlapping blocks, known as range blocks, and then creates a domain pool with all of the 16x16 pieces that could overlap and are connected by 8 isometries from rotations and reflections, known as domain blocks. After applying a contractive affine transform on the

domain=block, it does an exhaustive search within a domain pool for the best matched domain block with the smallest square error for each range block.Quantized contractively coefficients in the affine transform, an offset that is the average of the range block's pixel gray levels, the location of the best-matched domain block, and its kind of isometry make up a fractal compressed code for a range block. The goal of decoding is to start with any initial image and determine the fixed point, or the decoded image. Until all of the decoded range blocks are retrieved, the process performs a compressed local affine transform to the domain block that corresponds to a range block's position. Iteratively, the process is repeated until it converges, which typically takes no more than eight iterations.
The processing requirement and the existence problem of optimum range-domain matching are two significant issues that arise in fractal encoding [19]. The resolution-independent decoding property is the most appealing feature. To enhance the compression ratio exponentially, one can enlarge an image by decoding a smaller encoded image [15,18]. This research [17] compares a method based on [20] that uses range and domain block matches of defined sizes.

XIX. 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

For the images in Fig. 2, the corresponding PSNR values and encoding/decoding times displayed in Tables III, IV, and V show that all four methods are adequate at a request of 0.5 bpp (CR=16). But when compared to the other methods, the EZW [11, 8] has noticeably higher PSNR values and better-looking decoded images.

Both the EZW [9] and JPEG [7] approaches work well, and the results of EZW have significantly larger PSNR values than those of JPEG. However, 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 the resolution-free decoding property is utilized, which is not useful for the current purpose. Fig. 2 displays the original Lenna and fingerprint photos.

(a) (b)

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

The decoded images of Leena based on the four approaches

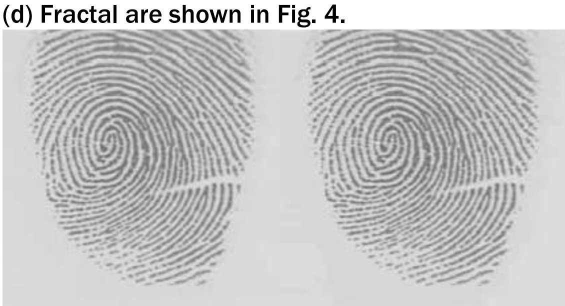
(a) Wavelet Transform, (b) JPEG, (c) Vector Quantization, (d) Fractal are shown in Fig. 3.

(a) (b)

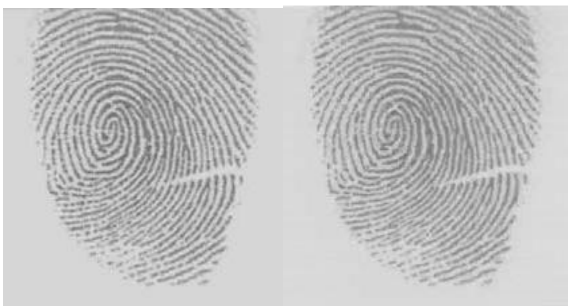
(c) (d)

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.



(a) (b)



(c) (d)

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

IX. Conclusion

The features of image compression, its necessity, its underlying concepts, the numerous classes of compression techniques, and the various image compression algorithms based on Wavelet, JPEG/DCT, VQ, and Fractal approaches have all been studied and summarized. The following recipe is suggested by experimental comparisons between a 400x400 fingerprint image and a 256x256 image of Lenna that is frequently utilized. When 0.5 bits per pixel (bpp) is needed, any of the four methods works well. However, the embedded zero tree wavelet (EZW) method outperforms other methods for extremely low bit rates, such as 0.25 bpp or less. We come to the following conclusions for real-world applications: (1) Wavelet-based compression algorithms are highly advised; (2) DCT-based approaches may employ an adaptive quantization table; (3) VQ-based approaches, despite their simplicity, are not suitable for low bit rate compression; and (4) Fractal-based approaches should make use of their resolution-free decoding property.

References

[1] "Discrete Cosine Transform," IEEE Trans. Computers, vol. C-23, Jan. 1974, pp. 90-93, Ahmed, N., Natarajan, T., and Rao, K. R.
[2] Embedded Predictive Wavelet Image Coder (EPWIC) by Buccigrossi, R. and Simoncelli, E. P.
[3] "Wavelet Basics," by Y. T. Chan, Kluwer Academic Publishers, Norwell, MA, 1995.
[4] "Vector Quantization and Signal Compression," Gersho, A., and Gray, R.M., Kluwer Academic Publishers, 1991.
[5] Malavar, H. S., "Signal Processing with Lapped Transforms," Artech House, Norwood, MA, 1992.
In July 1989, IEEE Trans. PAMI, vol. 11, no. 7, pp. 674-693, Mallat, S.G., "A Theory for Multiresolution Signal Decomposition: The Wavelet Representation."
[7] [Online] ftp.uu.net:/graphics/jpeg/jpegsrc.v6a.tar.gz is accessible.
[8] J.M. Shapiro, "Zero tree of wavelet coefficients for embedded image coding," IEEE Transactions on Signal Processing,
vol. 41, 1993, 3445–3462.
[9] A. Said and W.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, 243-250, 1996.
[10] Rao, K. R., and Yip, P. "Discrete Cosine Transforms-Algorithms, Advantages, Applications," Academic Press, 1990.
[11] "Image coding using vector quantization: areview," IEEETrans. On Communications, vol. 36, 957-571, 1988, N.M. Nasrabadi, R.A. King.
[12] "A vector quantizer design algorithm," IEEE Trans. on Communications, vol. 36, 84-95, 1980, by Y. Linde, A. Buzo, and R. M. Gray.
[13] Wavelets, "Subband Coding," Englewood Cliffs, NJ, Prentice Hall, 1995, Vetterli, M., and Kovacevic, J.
[Online] http://cm.bell-labs.com/who/jelena/Book/home.html is accessible.
In April 1991, Wallace, G. K., "The JPEG Still Picture Compression Standard," Comm. ACM, vol. 34, no. 4, pp. [15] 30–44.
[16] "Fractal Image Compression," by M.F. Barnsley and L.P. Hurd, AK Peters, Ltd., Wellesley, Mass., 1993.
[17] M.S. Thesis, June 1998, National Tsing Hua University, "Vector Quantization by Principal Component Analysis," Y.W. Chen.
[18] H.S. Chu, "A extremely quick fractal compression algorithm," National Tsing Hua University, June 1997, M.S. Thesis.
[19] "Fractal Image Compression," SIGGRAPH Course Notes, 1992, by Y. Fisher.
[20] "Fractal Image Compression: Theory and Applications," edited by Y. Fisher, Springer-Verlag, 1994.
[21] "Image coding using a fractal theory of iterated contractive image transformations," by A.E. Jacquin. Vol. 1, pp. 18-30, IEEE Trans. on Image Processing, 1992.