# Quality Assessment of Colour Image Compression using Haar Wavelet Transform

Sanjeev Kumar[1], Varun Sood[2]

[#]*Department of ECE, Amritsar College of Engg. & Technology*
*12 k.m Stone, Manawala, Amritsar, India*

*Abstract*— **Images require substantial storage and transmission resources, thus image compression is advantageous to reduce these requirements. This paper covers some background of wavelet analysis, data compression and how wavelets have been and can be used for image compression. The paper examines a set of wavelet functions (wavelets) for implementation in a still image compression system and discusses important features of wavelet transform in compression of still images, including the extent to which the quality of image is degraded by the process of wavelet compression and decompression. The effects of different wavelet functions, image contents and compression ratios are assessed.**

*Keywords*— **PSNR, MSE, CR, BPP.**

## I. Introduction

Image compression is an important field of research that has been studied for nearly three decades now. Image Compression addresses the problem of reducing the amount of data required to represent the digital image. Compression is achieved by the removal of one or more of three basic data redundancies: (a) Coding redundancy, which is present when less than optimal (i.e. the smallest length) code words are used; (b) Inter-pixel redundancy, which results from correlations between the pixels of an image & (c) psycho visual redundancy which is due to data that is ignored by the human visual system (i.e. visually nonessential information).Demand for communication of multimedia data through the telecommunications network and accessing the multimedia data through Internet is growing explosively. Compression of images has numerous applications in diverse areas such as high definition television, videophones, medical imaging, on-line product catalogs and other multimedia applications. Another important application is browsing, where the focus is on getting high compression.

There are two types of image compression: lossless and lossy. With lossless compression, the original image is recovered exactly after decompression. Unfortunately, with images of natural scenes it is rarely possible to obtain error-free compression at a rate beyond 2:1. Much higher compression ratios can be obtained if some error, which is usually difficult to perceive, is allowed between the decompressed image and the original image [1]. This is lossy compression. In many cases, it is not necessary or even desirable that there be error-free reproduction of the original image. For example, if some noise is present, then the error due to that noise will usually be significantly reduced via some de-noising method. In such a case, the small amount of error introduced by lossy compression may be acceptable. Another application where lossy compression is acceptable is in fast transmission of still images over the Internet. Unlike lossless compression, lossy compression reduces image quality. You can't get the original image back after using lossy compression methods. You will lose some information [2].

Lossless image compression is usually used in artificial images that contain sharp-edged lines such as technical drawings, textual graphics, comics, maps or logos. This is because lossy compression methods produce compression artefacts to images and sharp-edged lines become fuzzy especially when using strong compression. Instead, lossy compression is a good choice for natural images such as photos of landscapes where minor loss on sharpness is acceptable to achieve smaller file size. With the naked eye it is very hard to see any differences between uncompressed natural image and one with compressed by lossy methods if the compression is not too strong [3]. The most widely used methods of lossless compression in images are run-length encoding (RLE), entropy coding and dictionary coders. Lossy compression is usually based on techniques by removing details that the human eye typically doesn't notice. Digital images are composed of pixels that represent colour information. When a pixel differs only slightly from its neighbours, its value can be replaced theirs. This will lose some information but it is usually barely noticeable with human eye if the algorithm is good enough. After this e.g. RLE or Huffman coding can be used to compress data. Mostly used lossy compression method is transform coding such as discrete cosine transforms (DCT, used in JPEG) or wavelet transform (used in JPEG 2000).

Other popular methods are colour quantization (reducing the colour space) and chroma sub-sampling. These methods are based on a fact that the human eye is more sensitive to luminance than colour, so file size can be optimized by storing more luminance detail than colour detail. Also fractal compression is used but it's not so popular. For many years, the most popular image compression technique was based on the discrete cosine transform (DCT) [4]. In the recent past, wavelets have emerged as an important technique for image compression.

## II. Image Compression Using 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).

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 artefacts. Wavelet-based coding [5] 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.

The methods of lossy compression that we shall concentrate on are the following: EZW algorithm and SPIHT algorithm. These are relatively recent algorithms which achieve some of the lowest errors per compression rate and highest perceptual quality yet reported. Before we examine the algorithms listed above, we shall outline the basic steps that are common to all wavelet-based image compression algorithms. The five stages of compression and decompression are shown in Figs. 1 and 2.

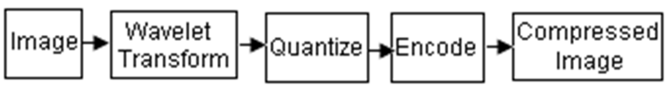


Figure 1: Compression of an image

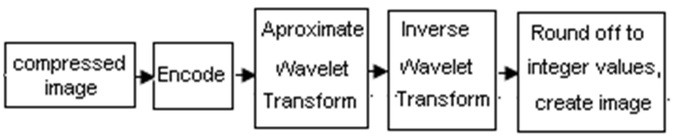


Figure 2: Decompression of an image

All of the steps shown in the compression diagram are invertable, hence lossless, except for the Quantize step. Quantizing refers to a reduction of the precision of the floating point values of the wavelet transform, which are typically either 32-bit or 64-bit floating point numbers. To use less bits in the compressed transform which is necessary if compression of 8 bpp or 12 bpp images is to be achieved these transform values must be expressed with less bits for each value. This leads to rounding error. These approximate, quantized, wavelet transforms will produce approximations to the images when an inverse transform is performed. Thus creating the error inherent in lossy compression. The relationship between the Quantize and the Encode steps, shown in Fig. 1, is the crucial aspect of wavelet transform compression. Each of the algorithms described below takes a different approach to this relationship. The purpose served by the Wavelet Transform is that it produces a large number of values having zeroed, or near zero, magnitudes.

Two commonly used measures for quantifying the error between images are Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR). The MSE between two images f and g is defined by

$$MSE = \frac{1}{N}\sum_{j,k}(f[j,k] - g[j,k])^2$$

where the sum over j; k denotes the sum over all pixels in the images, and N is the number of pixels in each image. The PSNR between two (8 bpp) images is, in decibels,

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right)$$

*1) EZW algorithm*

The EZW algorithm was one of the first algorithms to show the full power of wavelet based image compression. It was introduced in the groundbreaking paper of Shapiro [7]. Many algorithms build upon the fundamental concepts that were first introduced with EZW. EZW stands for Embedded Zerotree Wavelet. We shall explain the terms Embedded, and Zerotree, and how they relate to Wavelet-based compression. An embedded coding is a process of encoding the transform magnitudes that allows for progressive transmission of the compressed image. Zerotrees are a concept that allows for a concise encoding of the positions of significant values that result during the embedded coding process. The embedding process used by EZW is called bit-plane encoding.

*2) Set Partitioning in Hierarchical Trees (SPIHT encoding)*

The SPIHT [8-9] image coding algorithm was developed in 1996 by Said and Pearlman and is another more efficient implementation of the embedded zerotree wavelet algorithm by Shapiro. After the wavelet transform is applied to an image, the main algorithm works by partitioning the wavelet decomposed image into significant and insignificant partitions based on the following function:

$$S_n(T) = \begin{Bmatrix} 1, \max_{(i,j)\epsilon T}\{|c_{i,j}|\} \geq 2^n \\ 0, \text{otherwise} \end{Bmatrix} \ldots\ldots.1$$

Where Sn(T), is the significance of a set of coordinates T, and $c_{i,j}$ is the coefficient value at coordinate (i, j) . There are two passes in the algorithm - the sorting pass and the refinement pass. The sorting pass is performed on the list of insignificant sets (LIS), list of insignificant pixels (LIP) and the list of significant pixels (LSP). The LIP and LSP consist of nodes that contain single pixels, while the LIS contains nodes that have descendants. The maximum number of bits required to represent the largest coefficient in the spatial orientation tree is obtained and designated as $n_{max}$, which is

$$n_{max} = [\log_2(\max_{i,j}\{|c_{i,j}|\})] \quad ..........2$$

During the sorting pass, those co-ordinates of the pixels which remain in the LIP are tested for significance by using eqn. 2. The result, Sn(T), is sent to the output. Those that are significant will be transferred to the LSP as well as having their sign bit output. Sets in the LIS (which consists of nodes with descendants will also have their significance tested and, if found to be significant, will be removed and partitioned into subsets. Subsets with a single coefficient and found to be significant will be added to the LSP, or else they will be added to the LIP. During the refinement pass, the nth most significant bit of the coefficients in the LSP is output.

The value of n is decreased by 1 and the sorting and refinement passes are repeated. This continues until either the desired rate is reached or n =0, and all the nodes in the LSP have all their bits output. The latter case will result in almost perfect reconstruction as all the coefficients are processed completely. The bit rate can be controlled precisely in the SPIHT [8-9] algorithm because the output produced is in single bits and the algorithm can be terminated at any time. The decoding process follows the encoding exactly and is almost symmetrical in terms of processing time.

## III. RESULTS

Compressing colour images efficiently are one of the main problems in multimedia applications. So we have tested the efficiency of colour image compression using EZW and SPIHT algorithm. Reconstructed image is verified using human vision and PSNR. Table 1 shows the results of image compression.

TABLE 1. QUALITY ASSESSMENT

| | MSE | PSNR | CR | BPP |
|---|---|---|---|---|
| **EZW** | 10.1191 | 38.0794 | 18.0450 | 4.3308 |
| **SPIHT** | 12.5269 | 37.1524 | 12.2350 | 2.9364 |

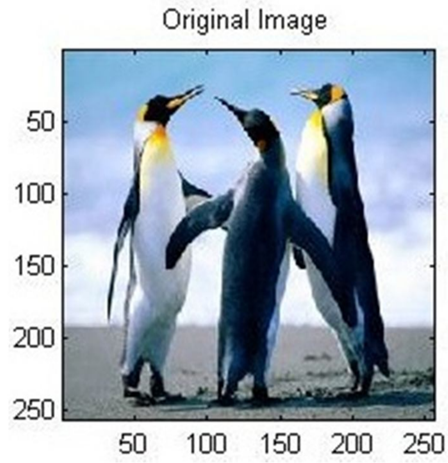


Fig. 1 Original 256 x 256 image

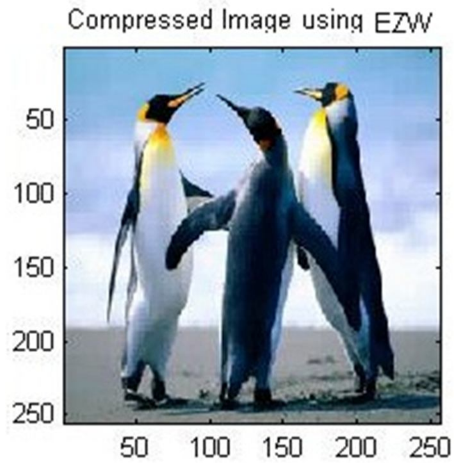


Fig. 2 Compressed image using EZW Haar Wavelet

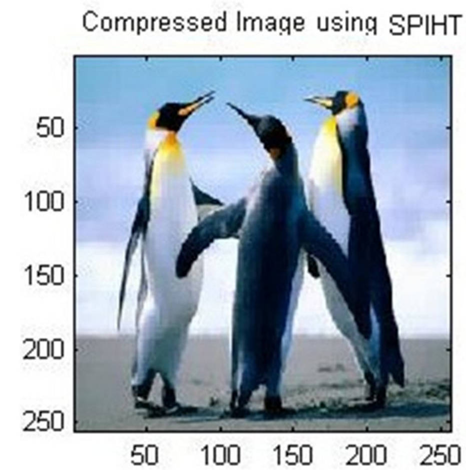


Fig. 3 Compressed image using SPIHT Haar Wavelet

## IV. CONCLUSIONS

We have reviewed and summarized the characteristics of image compression, need of compression and its principles and EZW and SPIHT image compression algorithms based on Wavelet. We use 256×256 color image for comparison. Any of the two 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. Also EZW gives better compression ratio and quality of images. However if 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] D.Donoho, M.Vetterli, R.DeVore and I.Daubechies, "Data Compression and Harmonic Analysis," IEEE Tr. On IT, Special Issue, Information Theory: 1948-1998 Commemorative Issue, Vol. 44, No. 6, Oct.1998, pp.2435-2476.

[2] A.Ortega and K.Ramchandran, "Rate-distortion methods for image and video compression," IEEE Signal Processing magazine, Oct. 1998.

[3] http://en.wikipedia.org/Image_compression.

[4] K. R. Rao and P. P. Yip, "Discrete Cosine Transform: algorithm, advantages, applications ", Academic Press, New York, 1990.

[5] M. J. Nadenau, J. Reichel, and M. Kunt, " Wavelet Based Color Image Compression: Exploiting the Contrast Sensitivity Function, " IEEE Transactions Image Processing, Vol. 12, no.1, PP. 58.

[6] K. H. Talukder, and K. Harada, " Haar Wavelet Based Approach for Image Compression and Quality Assessment of Compressed Image, " IAENG International Journal of Applied Mathematics, 2007.

[7] J.M. Shapiro. Embedded image coding using zerotrees of wavelet coefficients. IEEE Trans. Signal Proc., Vol. 41, No. 12, pp. 3445{3462, 1993.

[8] A. Said, W.A. Pearlman. Image compression using the spatial-orientation tree. IEEE Int. Symposium on Circuits and Systems, Chicago, IL, pp. 279-282, 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, No. 3, pp. 243-250, 1996.