ISSN (Online): 2583-2468
Volume-2 Issue-5 || September 2023 || PP. 50-57 DOI: 10.54741/asejar.2.5.8

Multiple Description Coding for Efficient, Low-Complexity Image Processing

NKEMDILIM Olusegun P.
Assistant Professor, Department of Chemical Engineering, Lagos State University, Nigeria

Corresponding Author: nolusegunp98@gmail.com

Received: 01-09-2023	Revised: 15-09-2023	Accepted: 30-09-2023

ABSTRACT
This study uses Set Partitioned Embedded bloCK based coding, which is quick, effective, straightforward, and often used, to code several descriptions of changed images. Using images' discrete wavelet transform (DWT), this type of coding can be used to its fullest capacity. To enable accurate transmission of the image over noisy wireless channels, two associated descriptions are created from a wavelet processed image. These associated descriptions are broadcast across wireless channels using the set partitioning technique and SPECK coders. The quantity of descriptions received affects the quality of the image reconstruction at the decoder side. The quality of the reconstructed image improves as the number of descriptions received at the output side increases. When a description is lost from the various descriptions, the receiver can still guess it by using the correlation between the descriptions. Even when half of the descriptions are lost in transmission, the simulations run on an image in MATLAB still produce respectable performance and outcomes.

Keywords: *coding effect, low complexity, image processing*

I. INTRODUCTION

Sending, receiving and sharing of images has become the part of our daily lives. Social media fuelled this trend of sharing images with more and more people sharing images casually. To keep up with this ever increasing demand of image communication, compression of images is done so that the sharing and transfer of images is not hindered by the limitations of the bandwidth, battery power, storage and processing power of the handheld devices.

Uncompressed images have large amount of data, some of which can be redundant for our application and use. This uncompressed image data have high correlation. Therefore, image compression is done in almost every image after acquisition. The storage and bandwidth requirements of a compressed image data is only a fraction of the requirements of the original contents. But this is not the only reason for widespread use of image compression. The reduced computational complexity and memory and embedded feature of the compression algorithms being more important for several applications.

In our work on compression, we have processed the image using three main techniques. Firstly, we have transformed the image using Discrete Wavelet Transform (DWT)[9,11]. Multiple Descriptive Coding is done on the transformed image. Then, these descriptions are encoded using SPECK[5,6,7,8] coder. The paper is organized in such a way that the processes involved are described in the sequence of their application. Section 2 is the description of DWT. Sections 3 and 4 explains the application of MDC and SPECK respectively. Finally, we will have simulation results and conclusion.

II. DISCRETE WAVELET TRANSFORM

The Discrete Wavelet Transform is based on sub-band coding. By using digital filtering method, time-scale representation of the digital signal is obtained in DWT. These digital filters are mainly used to suppress either the high frequencies in the image (smoothing the image), or the low frequencies, (enhancing or detecting edges in the image).

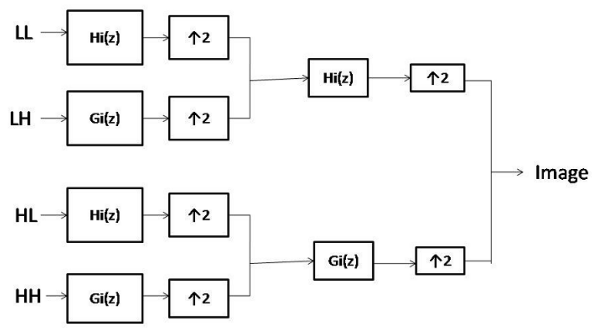

Figure 1(a): DWT filter implementation

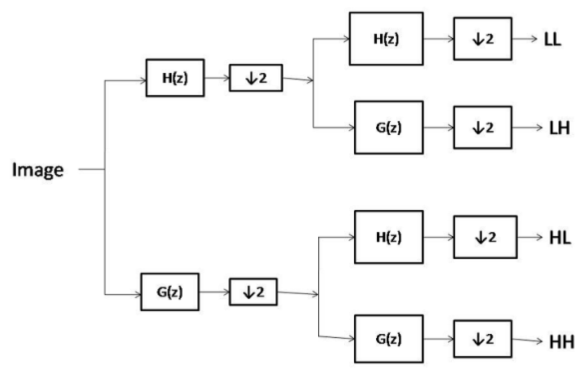

Figure 1(b): IDWT filter implementation

In the discrete wavelet transform, the image signal is processed by passing through an analysis filter bank. At each decomposition stage of the process, the analysis filter banks consisting of a low-pass and high-pass filter is used. When the signal passes through these low and high pass filters, it split through two bands. The low-pass filter of this analysis bank, which is responsible for the averaging operation of the image sample, extracts the coarse information of the digital image. The high-pass filter operation corresponds to a differencing operation, and it extracts the detailed information of the signal or image. After that, the output of the filtering operation is decimated by two. The two-dimensional transformation of time and frequency is accomplished by performing two separate one-dimensional transforms. The image is filtered along the row and the the outcome is decimated by two. Then it is followed by filtering the sub bands of the image along the column and decimation by two. This DWT[10,11] operation splits the image into four bands, which are LL, LH, HL, and HH respectively.

Inverse of DWT is done at the decoder side to get the compressed image. The process of filtering is inverted where decimation is inverted by interpolation.

The 2- level decomposition in DWT can also be represented as

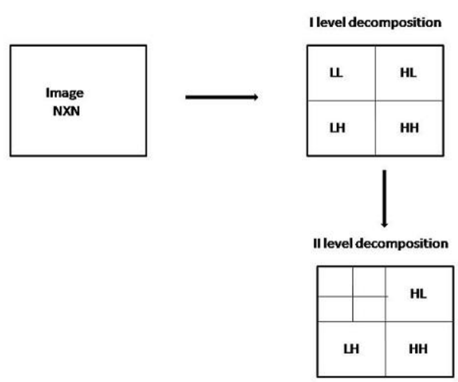

Figure 1(c): Subband Decomposition of Image

The Wavelets transform captures most image information in the highly sub sampled low frequency band (LL) also called as the approximation signal. The additional information at localized edges in the form of coefficients will be in the high frequency bands (HL, LH, and HH). Another attractive aspect of the coarse to fine nature of the wavelet representation naturally facilitates a transmission feature that enables progressive transmission as an embedded bit stream.

Multiple Descriptive Coding

Multiple Descriptive Coding[2,3,4] increases the reliability of transmission through wireless channels. Two-descriptions of the original image data are transmitted through two channels here but a higher number of descriptions are possible. In this figure 2, an image is coded such that two complementary and correlated descriptions that are individually decodable are generated and transmitted separately, through two different network paths. The descriptions may get lost due to noise or congestion in channels. So one or both descriptions can reach the receiving side. At the receiver side, if only one description is reached , it is decoded by the *side decoder* and the resulting quality(distortion) is called *side quality (distortion)*. When both descriptions are reached, they are decoded by the *central decoder* and the resulting quality (distortion) is called *central quality (distortion)*. In central decoder, the two descriptions are merged and hence an image with higher quality is achieved. In other words, two types of decoding is done at the receiver, when all descriptions are received, the central decoding is used, and if one or more descriptions are lost in the transmission, the side decoder is used for the description(s) received. As we shall see in our result that the PSNR value do not decrease significantly on the loss of information of descriptions, this is because the decoder exploit the correlation among descriptions, and approximate the lost data for reconstruction of image.

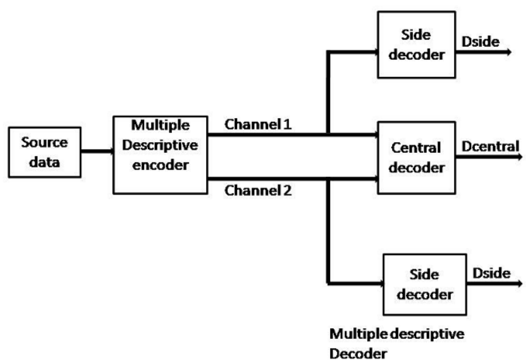

Figure 2: Block diagram of Multiple Descriptive Coding

III. SPECK CODING TECHNIQUE

The basic logic behind the coding method of SPECK algorithm is given here. An image has been adequately transformed using discrete wavelet transformation. An image after wavelet transform exhibits a hierarchical pyramidal structure defined by the decomposition levels, with the topmost level being the root. The finest pixels of the transformed image lie at the bottom level of the structure while the coarsest pixels lie at the root level. The SPECK[7,8] algorithm exploits the rectangular regions of the image defined as sets. In the algorithm, sets of varying sizes are formed, depending on the characteristics of pixels in the original set. A set of size 1 will have just one pixel. These sets are formed by chopping off a small square part from the top left of a larger region. Following figure 3 shows the formation of sets.

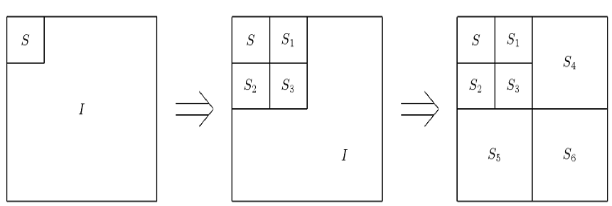

Figure 3: Partitioning of an image into sets S & I,

There are two linked lists: LIS - List of Insignificant Sets, and LSP - List of Significant Pixels. These two lists are maintained in SPECK algorithm. The LIS contains sets of type S of varying sizes which is not found significant against a threshold n while LSP contains those pixels which have been tested significant against n. Two types of set partitioning are used in SPECK: quad tree partitioning and octave band partition as shown in figure below.

ISSN (Online): 2583-2468
Volume-2 Issue-5 || September 2023 || PP. 50-57 DOI: 10.54741/asejar.2.5.8

IV. SPECK CODING PROCEDURE

The SPECK coding procedure[7] is explained here with an example of encoding data of the type resulting from an 8x8 two-level wavelet transform. In this type of coding, partitions are generated recursively. Here, partition of square blocks of contiguous data elements is presented. Since these elements are arranged as two-dimensional array, we shall call them *pixels* and suppose we have a square $2^3 \times 2^3$ array of pixels. First, the square array of source data is split into four $2^{3-1} \times 2^{3-1}$ quadrants, pictured in figure. At least one of those quadrants contains an element greater than the threshold (2^{nmax}), i.e., $p_j \geq 2^{nmax}$.

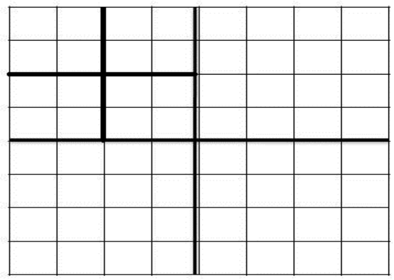

Figure 4(a): Partitioning data set into an 8x8 array of pixels

Those quadrants containing elements such that $p_j \geq 2^{nmax}$ are labeled as "1", and those having no such elements as "0". The data elements in the quadrants labeled with "0" require at most n_{max} bits for lossless representation. Now, we split the "1" labeled quadrants into four $2^{3-2} \times 2^{3-2}$ element quadrants and test each of these four new quarter-size quadrants, whether or not all of its elements are smaller than 2^{nmax}.

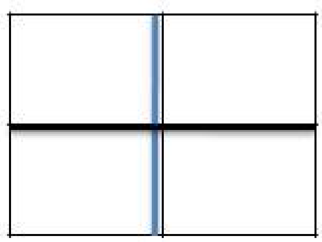

Figure 4(b): Portioning 4x4 array further into 2x2 according to significance

Again, we label these new 2x2 quadrants with "1" or "0", depending whether any element in the quandrant is significant, i.e., $p_j \geq 2^{nmax}$ or not, respectively. Again any "0" labeled quadrant requires n_{max} bits for lossless representation of its elements. Quadrant labeled "1" is again split into four equal parts (quadrisected), with each part tested again whether its elements exceed the threshold 2^{nmax}. This procedure of quadrisection and testing is continued until the '1'-labeled quadrants are split into 4 single elements, whereupon all the individual elements greater than or equal to 2^{nmax} are located. These elements are known to be one of the 2^{nmax} integers from 2^{nmax} to $2^{nmax+1} - 1$, so their differences from 2^{nmax} are coded with n_{max} bits and inserted into the bitstream to be transmitted. The single elements less than 2^{nmax} can be coded now with n_{max} bits. What also remains are sets of sizes 2×2 to $2^{3-1} \times 2^{3-1}$ labeled with "0" to indicate that every element within these sets is less than 2^{nmax}. Figures are shown above to understand level and types of decomposition. Three levels of splitting and labeling is done. But the algorithm would not be efficient by finding sets requiring just one less bit for representation of its elements. So, the threshold is lowered by a factor of 2 to 2^{nmax-1} and above procedure of quadrisection is repeated and labeling is done on the "0"-labeled sets already found.

In this way, SPECK coding exploits the clustering or accumulation of energy in frequency and space in the hierarchical structures of wavelet transformed images. For reconstructing the compressed image, inverse of SPECK coding is done at the decoder.

ISSN (Online): 2583-2468
Volume-2 Issue-5 || September 2023 || PP. 50-57 DOI: 10.54741/asejar.2.5.8

V. SIMULATION RESULT

To evaluate the performance of the presented method through MATLAB, the standard Lena image, Barbara image, Goldgate and House images (with 512*512 pixels and 8 Bpp bitrate) are used as the test images in the simulation. A 5-level wavelet decomposition quantized to nearest integers using filers is simulated through MATLAB. The test images are encoded upto the last bit plane. The encoding of images is done at 1bpp. After processing, these images are decoded at different bit rates as shown in the figure. Variation of PSNR with bits per pixel (bpp) is given in the chart below. We can see in the chart that the degradation in image quality (PSNR) is almost negligible when data is lost in transmission. This result shows that the quality of image remains almost same even if the 20% or 50% data is lost. So this scheme of compression is very reliable when image is transmitted on a noise prone channel. We get decent quality of image even after half of the data is lost.

Image	**bpp**	**PSNR (in dB)**		
		All data received	**50% packet lost**	**25% data lost**
Lena	0.5	28.1504	27.4272	27.8091
	0.25	23.9964	23.9009	23.8903
	0.125	21.1391	21.1354	21.1421
Barbara	0.5	23.9682	22.9702	23.709
	0.25	20.9343	20.8579	20.9181
	0.125	19.0213	18.7791	18.7501
Goldgate	0.5	33.1238	31.045	32.3067
	0.25	29.3595	28.482	29.0553
	0.125	27.6008	27.2018	27.5061
House	0.5	23.9899	21.6488	23.3081
	0.25	21.3282	20.3638	21.0944
	0.125	19.8207	19.2443	19.6856

Table 1: Variations in PSNR with change in bit rate for test images

5.1 Comparative Analysis of Coding Efficiencies with Data Lost during Transmission

Using the data from the simulation results, the degradation in quality of images is compared when no data is lost during transmission, when 50% data is lost during transmission and when 25% data is lost during transmission.

5.1.1 LENA Image

Through simulation we have represented a coding efficiency, when no data is lost during transmission (represented by red line), when 50% data is lost during transmission (represented by black line), and when 25% data is lost(represented by blue line).

Figure (a): Image when 25% data is lost at 0.5 bpp

Figure (b): Image when 50% data is lost at 0.5 bpp

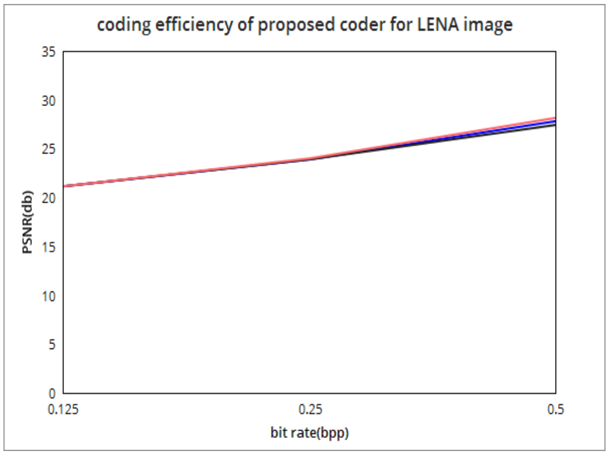

Figure 5: Coding efficiency for LENA image

5.1.2 BARBARA Image

Again we have used the test image of Barbara and simulated the coding to get data given in figure 4.1. The comparisons in image quality with subsequent loss in data is plotted separately as shown in figure 4.3. Red line represents the efficiency when no data is lost, black line represents when 50% data is lost, and blue line represents when 25% data is lost.

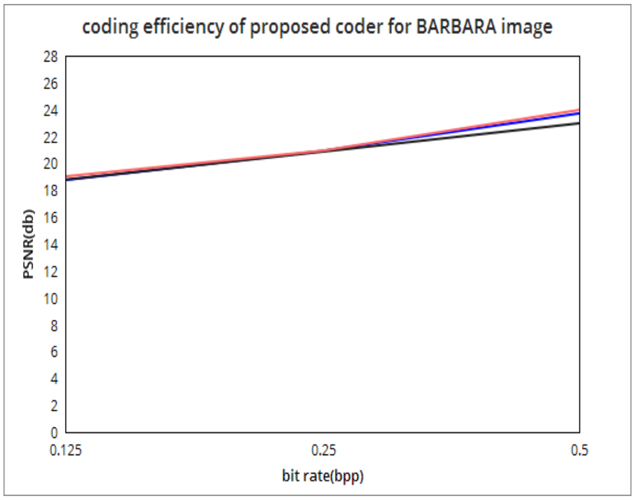

Figure 6: Coding efficiency for BARBARA image

5.1.3 GOLDGATE Image

The coding efficiency of this standard test image is found out by simulation and results obtained are shown in figure 4.2. Now for comparative analysis, PSNR vs bitrates graphs are plotted for no loss in data during transmission (represented by red line), when 50% data is lost during transmission (represented by black line), and when 25% data is lost(represented by blue line).

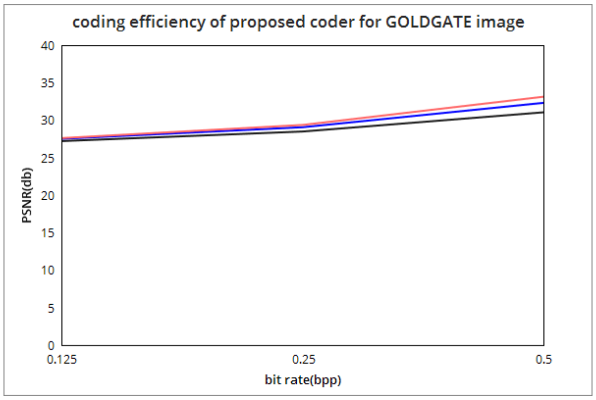

Figure 7: Coding efficiency for GOLDGATE image

Figure (a): Image when 25% data is lost

Figure (b): Image when 50% data is lost
Figure 8: Final output images of GOLDGATE at 0.5 bpp

5.1.4 HOUSE Image

This is the last image tested on our coding. The comparative graph plotted for percentage of data lost is shown in figure 4.5, with red line representing the efficiency when no data is lost, black line representing when 50% data is lost, and blue line when 25% data is lost.

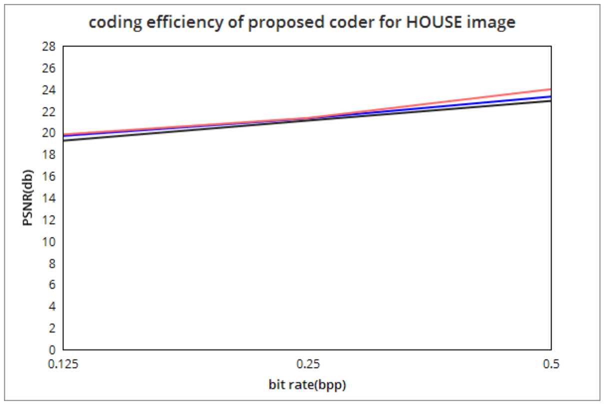

Figure 9: Coding efficiency for HOUSE image

VI. CONCLUSION

This study presents an MDC-based compression strategy that is straightforward, quick, and effective. This system enables dependable transmission over congested channels. Both the SPECK coder and the bandwidth-efficient MDC use less processing power. When MDC is used, reliability is still present even if a sizable portion of data is lost. Data transmission using two channels greatly lowers overhead costs by lowering wireless transmission network congestion. Additionally, the SPECK coding employed here is incredibly quick and easy. The efficiency of the compression method is increased because the SPECK coder utilised in this study completely takes use of the clustering of energy in lower frequency bands by DWT. Even while the SPECK utilised in this case uses more memory than the listless SPECK, the gains in efficiency and dependability more than make up for this limitation. The hierarchy of sorting in the SPECK used in our research begins with high magnitude critical data and moves down to lower magnitude data. The amount of data already encoded can therefore still be used to produce a compressed image at the decoder side, although one of slightly lower quality, in the event that the coder abruptly stops in the middle of its operation. The compression strategy is more effective and less complex when DWT, MDC, and SPECK are used.

REFERENCES

1. Y. Wang, M.T. Orchard, V.A. Vaishampayan, & A.R. Reibman. (2001). Multiple description coding using pairwise correlating transforms. *IEEE Transactions on Image Processing*, *10*(3), 351-366.
2. S. D. Servetto, K. Ramchandran, & V.A Vaishampayan. (2000). Multiple description wavelet based image coding. *IEEE Transactions on Image Processing*, *9*(5), 813-826.
3. V. K. Goyal & J. Kovacevic. (2001). Generalized multiple description coding with correlating transforms. *IEEE Transactions on Information Theory*, *47*(6), 2199-2224.
4. V.K. Goyal. (2001). Multiple description coding: Compression meets the network. *IEEE Signal Processing Magazine*, *18*(5), 74-93.
5. A. Said, & W.A. Pearlman. (1996). A new fast and efficient image codec based on set partitioning in hierarchical trees. *IEEE Transactions on Circuits and Systems for Video Technology*, *6*(3), 243–250.
6. Islam, & W.A. Pearlman. (1999). *An embedded and efficient low-complexity hierarchical image coder*. Available at: https://www.ecse.rpi.edu/~pearlman/papers/vcip99_ip.pdf.
7. W.A. Pearlman, A. Islam, N. Nagaraj, & A. Said. (2004). Efficient, low-complexity image coding with a set-partitioning embedded block coder. *IEEE Transactions on Circuits and Systems for Video Technology, 14*(11), 1219–1235.
8. K. Sayood. (2000). *Introduction to data compression*. (2^{nd} ed.). Academic Press: Morgan Kaufmann Publishers.
9. G. Strang, & T. Nguyen. (1996). *Wavelets and filter banks*. Cambridge: Wellesley-Cambridge Press.
10. Li Ze-Nian, & Drew Mark S. (2004). *Image compression standards in fundamentals of multimedia*. Prentice-Hall.
11. Lewis A. S., & Knowles G. (1992). Image compression using the 2-D wavelet transform. *IEEE Transaction on Image Processing*, 244-250.